

Oracle® Retail Merchandising Cloud Services

Inbound and Outbound Integration Guide



G58997-02
August 2026



Copyright © 2026, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

Send Us Your Comments

Preface

Audience	i
Documentation Accessibility	i
Customer Support	i
Improved Process for Oracle Retail Documentation Corrections	ii
Oracle Retail Documentation on the Oracle Help Center (docs.oracle.com)	ii
Conventions	ii

1 Introduction

2 RIB Publication Designs

Allocations Publication API	1
Functional Area	1
Business Overview	1
New Allocations	2
New Allocation Details	4
Updated Allocations	4
Deleted Allocations	5
Error Handling	6
Message XSD	6
ASN Outbound Publication API	6
Functional Area	6
Business Overview	6
On-line Shipping/Receiving	7
Franchise Order Shipment and Return	7
Advance Shipment Notification	8
Error Handling	11
Message XSD	11
Available Inventory for Store and Warehouse Publication API	12

Functional Area	12
Business Overview	12
Item-Location Level Available to Sell Message	12
Error Handling	13
Message XSD	13
Available Inventory for Store Publication API	13
Functional Area	13
Business Overview	13
Package Impact	14
Function Level Description - ADDTOQ	14
Function Level Description - GETNXT	14
Function Level Description - PUB_RETRY	14
Function Level Description - PROCESS_QUEUE_RECORD (local)	15
Function Level Description - HANDLE_ERRORS (local)	15
Message XSD	15
Table Impact	15
Design Assumptions	15
Banner and Channel Publication API	15
Functional Area	15
Business Overview	16
New Banners	16
Updated Banners	16
Deleted Banners	17
New Channels	17
Updated Channels	17
Deleted Channels	17
Error Handling	18
Message XSD	18
Code and Diff Type Publication API	18
Functional Area	18
Integration Type	18
Business Overview	18
New Code Types	19
Updated Code Types	19
Deleted Code Types	19
New Codes	20
Updated Codes	20
Deleted Codes	20
New Diff Types	21
Updated Diff Types	21
Deleted Diff Types	22
Error Handling	22

Message XSD	22
Company Closed Publication API	23
Functional Area	23
Business Overview	23
New Closings	23
Updated Closings	23
Deleted Closings	23
New Closing Exceptions	24
Updated Closing Exceptions	24
Deleted Closing Exceptions	24
Error Handling	25
Message XSD	25
Country Publication API	25
Functional Area	25
Business Overview	25
New Country	25
Updated Country	26
Deleted Country	26
Error Handling	26
Message XSD	27
Customer Order Fulfillment Confirmation Publication API	27
Functional Area	27
Business Overview	27
Error Handling	28
Message XSD	28
Delivery Slot Publication API	28
Functional Area	28
Business Overview	29
Create Delivery Slot	29
Update Delivery Slot	29
Delete Delivery Slot	29
Error Handling	29
Message XSD	30
Differentiator Group Publication API	30
Functional Area	30
Business Overview	30
New Diff Group Header	30
New Diff Group Detail	30
Updated Diff Group Header	30
Updated Diff Group Detail	30
Deleted Diff Group	31
Deleted Diff Group Detail	31

Error Handling	32
Message XSD	32
Differentiator Publication API	32
Functional Area	32
Business Overview	32
New Differentiator IDs	32
Updated Differentiator IDs	32
Deleted Differentiator IDs	33
Error Handling	33
Message XSD	33
Item Location Publication API	33
Functional Area	34
Business Overview	34
New Item Location Relationship	34
Modified Item Location Relationship	39
Deleted Item Location Relationship	40
Item Replenishment Attributes	40
Custom Flex Attributes	41
Error Handling	41
Message XSD	42
Item Publication API	42
Functional Area	42
Business Overview	42
Deposit items	42
Catch-Weight Items	43
Item Transformation	44
Item and Item Component Descriptions	45
New Item Message Processes	45
Basic Item Message	46
New Item Message Publication	46
Subordinate Data and XML Tags	46
Modify and Delete Messages	63
Modify Messages	64
Delete Messages	64
Flex Attributes	66
Full Message Updates	67
Error Handling	67
Message XSD	67
Merchandise Hierarchy Publication API	68
Functional Area	68
Business Overview	68
New Division	68

Updated Division	69
Deleted Division	69
New Group	69
Updated Group	69
Deleted Group	70
New Department	70
Updated Department	70
Deleted Department	72
New Class	72
Updated Class	72
Deleted Class	72
New Subclass	73
Updated Subclass	73
Deleted Subclass	74
Custom Flex Attributes	74
Error Handling	74
Message XSD	74
Order Publication API	75
Functional Area	75
Business Overview	75
New Purchase Orders	75
Updated Purchase Orders	82
Full Message Updates	82
Deleted Purchase Orders	82
Flex Attributes	83
Error Handling	84
Message XSD	84
Organizational Hierarchy Publication API	84
Functional Area	85
Business Overview	85
New Chains/Areas/Regions/Districts	85
Updated Chains/Areas/Regions/Districts	86
Deleted Chains/Areas/Regions/Districts	86
Error Handling	86
Message XSD	86
Partner Publication API	87
Functional Area	87
Business Overview	87
Partner Creation	87
Partner Header Modification	90
Partner Detail Creation	90
Partner Detail Modification	90

Deleting Partners	90
Deleting Partner Details	90
Flex Attributes	91
Error Handling	91
Message XSD	91
Receiver Unit Adjustment Publication API	92
Functional Area	92
Business Overview	92
Creating Receiver Unit Adjustments	92
Error Handling	93
Message XSD	93
RTV Request Publication API	93
Functional Area	93
Business Overview	93
Create RTV Request	94
Update RTV Request	95
Delete RTV Request	95
Flex Attributes	96
RTV Request Full Publish	96
Error Handling	96
Message XSD	97
Season and Phase Publication API	97
Functional Area	97
Business Overview	97
New Seasons	97
Updated Seasons	98
New Phases	98
Updated Phases	98
Deleted Seasons	98
Deleted Phases	98
Error Handling	99
Message XSD	99
Store Publication API	99
Functional Area	99
Business Overview	99
Create Stores	99
Modify Stores	100
Create Store Addresses	100
Modify Store Addresses	100
Create Store Hours	100
Modify Store Hours	100
Flex Attributes	104

Delete Stores	105
Error Handling	105
Message XSD	105
Transfers Publication API	106
Functional Area	106
Business Overview	106
New Transfer	106
Updated Transfer	107
Full Message Updates	107
Create or Update Transfer Header	107
Create or Update Transfer Detail	113
Pack Component Detail	114
Flex Attributes	114
Deleted Transfer	114
Transfer Unapprove	115
Error Handling	116
Message XSD	116
UDA Publication API	116
Functional Area	116
Business Overview	116
New UDAs	116
Updated UDAs	117
Deleted UDAs	117
New UDA Details	117
Updated UDAs	118
Deleted UDAs	118
Error Handling	118
Message XSD	118
Vendor Publication API	119
Functional Area	119
Business Overview	119
Vendor Creation	119
Vendor Header Modification	119
Vendor Address Creation	124
Vendor Address Modification	124
Vendor Org Unit Creation and Modification	125
Vendor Detail Deletion	125
Flex Attributes	126
Full Message Updates	126
Error Handling	127
Message XSD	127
Warehouse Publication API	127

Functional Area	127
Business Overview	127
Warehouse Creation	128
Warehouse Modification	128
Warehouse Addresses	131
Deleting Warehouses	132
Flex Attributes	133
Error Handling	133
Message XSD	133
Work Orders In Publication API	133
Functional Area	134
Business Overview	134
New Work Order	134
Updated Work Order	135
Deleted Work Order	135
Error Handling	136
Message XSD	136
Work Orders Out Publication API	136
Functional Area	136
Business Overview	136
New Work Order	137
Deleted Work Order	139
Error Handling	140
Message XSD	140

3 RIB Subscription Designs

Allocation Subscription API	1
Functional Area	1
Business Overview	1
Allocation Creation	1
Allocation Modification	2
Allocation Details	3
Deleting Allocations	3
Error Handling	4
Message XSD	4
Appointment Subscription API	4
Functional Area	4
Business Overview	4
New Appointments	4
New Appointment Details	5
Updated Appointments	5

Updated Appointment Details	5
Appointment Header	5
Appointment Detail Message	5
Delete Appointments	6
Delete Detail Level Appointments	6
Appointment Header	6
Appointment Detail Message	6
Error Handling	6
Message XSD	7
ASN Outbound Subscription API	7
Functional Area	7
Business Overview	7
Other Notes	8
Shipment Message Details	8
Error Handling	12
Message XSD	12
Cost Change Subscription API	12
Functional Area	12
Business Overview	12
Custom Flex Attributes	14
Error Handling	15
Message XSD	15
Currency Exchange Rate Subscription API	15
Functional Area	15
Business Overview	15
Error Handling	16
Message XSD	16
Customer Order Fulfillment Subscription API	16
Functional Area	16
Business Overview	17
Other Notes	17
Error Handling	24
Message XSD	25
Differentiator Group Subscription API	25
Functional Area	25
Business Overview	25
Creating Diff Groups	25
Updating Diff Groups	26
Deleting Differentiator Groups	26
Error Handling	27
Message XSD	27
Differentiator Subscription API	27

Functional Area	27
Business Overview	28
Creating Differentiators	28
Updating Differentiators	28
Deleting Differentiators	29
Error Handling	29
Message XSD	29
DSD Deals Subscription API	29
Functional Area	29
Business Overview	29
Error Handling	31
Message XSD	31
DSD Receipt Subscription API	31
Functional Area	31
Business Overview	32
Error Handling	33
Message XSD	34
Freight Term Subscription API	34
Functional Area	34
Business Overview	34
Creating/Updating Freight Terms	34
Error Handling	35
Message XSD	35
General Ledger Chart of Accounts Subscription API	35
Functional Area	35
Business Overview	35
Create and Update Chart of Accounts	35
Error Handling	37
Message XSD	37
GL Chart of Account Validation Service	37
Functional Area	37
Overview	38
Inbound ASN Subscription API	38
Functional Area	38
Business Overview	38
Creating/Updating ASN	39
Delete ASN	42
Error Handling	42
Message XSD	42
Inventory Adjustment Subscription API	43
Functional Area	43
Business Overview	43

Inventory Adjustment Transaction Codes	43
Other Notes	44
Error Handling	46
Message XSD	46
Inventory Request Subscription API	46
Functional Area	46
Business Overview	47
Creating Inventory Requests	47
Error Handling	48
Message XSD	48
Item Subscription API	48
Functional Area	48
Business Overview	48
Creating Items	49
Updating Items	54
Deleting Items	54
Creating an Item Supplier	55
Updating an Item Supplier	57
Deleting an Item Supplier	57
Creating an Item Supplier Country	57
Updating an Item Supplier Country	60
Deleting an Item Supplier Country	60
Creating Item Supplier Manufacturing Country	60
Updating Item Supplier Manufacturing Country	61
Deleting Item Supplier Manufacturing Country	61
Creating Item Supplier Country Location	61
Updating Item Supplier Country Locations	63
Deleting Item Supplier Country Locations	63
Creating Item Supplier Country Dimensions	63
Updating Item Supplier Country Dimensions	64
Deleting Item Supplier Country Dimensions	64
Creating Item VAT	65
Deleting Item VAT	65
Creating Item UDA	65
Updating Item UDA	66
Deleting Item UDA	66
Creating Item Season	67
Deleting Item Season	67
Creating Item Images	68
Updating Item Images	68
Deleting Item Images	68
Creating HTS and Assessments	69

Updating HTS and Assessments	70
Deleting HTS and Assesements	70
Creating Expenses	71
Updating Expenses	71
Deleting Expenses	72
Creating Tickets	72
Updating Tickets	72
Deleting Tickets	73
Creating Up-charges	73
Deleting Up-charges	74
Updating Up-charges Detail	74
Creating Conditional Tariff treatments	74
Creating Item Country	75
Deleting Item Country	75
Creating Related Items	76
Deleting Related Items	76
Updating Related Items	77
Creating Required Documents Items	77
Deleting Required Documents Items	77
Flex Attributes	77
Translated Language	77
Error Handling	80
Message XSD	80
Item Location Subscription API	82
Functional Area	82
Business Overview	82
Creating Locations	82
Updating Item Location	82
Flex Attributes	87
Error Handling	88
Message XSD	88
Item Reclassification Subscription API	88
Functional Area	88
Business Overview	88
New/Updated Reclassifications	89
Deleting Reclassifications	90
Error Handling	91
Message XSD	92
Location Trait Subscription API	92
Functional Area	92
Business Overview	92
Error Handling	92

Message XSD	93
Merchandise Hierarchy Reclassification Subscription API	93
Functional Area	93
Business Overview	93
Creating Merchandise Hierarchy Reclassifications	93
Updating Merchandise Hierarchy Reclassifications	95
Deleting Merchandise Hierarchy Reclassifications	95
Error Handling	96
Message XSD	96
Merchandise Hierarchy Subscription API	96
Functional Area	96
Business Overview	96
Creating a Company	97
Updating a Company	97
Creating Divisions	98
Updating Divisions	98
Deleting Divisions	99
Creating Groups	99
Updating Groups	99
Deleting Groups	100
Creating Departments	100
Updating Departments	100
Deleting Departments	102
Creating VAT Information for Departments	102
Updating VAT Information for Departments	103
Creating Up-charges for Departments	103
Updating Up-charges for Departments	103
Deleting Up-charges for Departments	105
Creating Classes	106
Updating Classes	106
Deleting Classes	106
Creating Subclasses	107
Updating Subclasses	107
Deleting Subclasses	108
Flex Attributes	108
Error Handling	109
Message XSD	109
Order Subscription API	110
Functional Area	110
Business Overview	110
Creating Purchase Orders	110
Updating Purchase Orders	116

Deleting Purchase Orders	117
Creating a Purchase Order Letter of Credit	118
Updating a Purchase Order Letter of Credit	119
Deleting a Purchase Order Letter of Credit	119
Creating Expenses	119
Updating Expenses	120
Deleting Expenses	121
Creating HTS and Assessments	121
Updating HTS and Assessments	122
Deleting HTS and Assessments	122
Publishing Updates	123
Flex Attributes	123
Error Handling	123
Message XSD	124
Organizational Hierarchy Subscription API	124
Functional Area	124
Business Overview	125
Other Notes:	125
Error Handling	125
Message XSD	125
Payment Terms Subscription API	126
Functional Area	126
Business Overview	126
Creating Payment Terms	126
Updating Payment Terms	129
Error Handling	129
Message XSD	129
Receiving Subscription API	129
Functional Area	129
Business Overview	129
Receiving Exceptions	130
Receipt	131
Receipt Detail	132
Receipt Carton Detail	134
Error Handling	135
Message XSD	136
RTV Subscription API	136
Functional Area	136
Business Overview	136
New RTVs	136
Updated RTVs	136
RTV Header	137

RTV Details	138
Flex Attributes	140
Error Handling	140
Message XSD	141
Stock Count Schedule Subscription API	141
Functional Area	141
Business Overview	141
Creating/Updating Stock Count Requests	141
Deleting Stock Count Requests	142
Error Handling	143
Message XSD	143
Stock Order Status Subscription API	143
Functional Area	143
Business Overview	144
Stock Order Statuses	144
Stock Order Status Message Details	146
Error Handling	148
Message XSD	148
Store Subscription API	149
Functional Area	149
Business Overview	149
Create or Update Stores	149
Delete Stores	149
Other Store Attributes	149
Store Message Details	150
Flex Attributes	158
Error Handling	158
Message XSD	159
Transfer Subscription API	159
Functional Area	159
Business Overview	159
Creating Transfers	159
Transfers with Finishing	165
Updating Transfers	165
Deleting Transfers	166
Publishing Updates	166
Flex Attributes	167
Error Handling	167
Message XSD	167
Vendor Subscription API	167
Functional Area	167
Business Overview	168

Vendor Create and Update	168
Flex Attributes	174
Error Handling	174
Message XSD	174
Work Order Status Subscription API	174
Functional Area	174
Business Overview	175
Work Order Example	175
Work Order Status Creation	175
Error Handling	176
Message XSD	177

4 SOAP Web Services

Using SOAP Services During Batch Window	2
Common Characteristics of Merchandising SOAP Services	2
Security	2
Standard Success Response	2
Standard Error Response	3
URL Path	3
Web Service APIs Process Flow	4
Provider Services	4
Consumer Services	4
Provider Services	5
Allocation Service	5
Functional Area	5
RSB Proxy WSDL	5
Merchandising Service WSDL	5
Overview	5
Operation XSD	5
Average Cost Service	6
Functional Area	6
RSB Proxy WSDL	6
Merchandising Service WSDL	6
Overview	6
Operation XSD	6
Cost Change Service	6
Functional Area	7
RSB Proxy WSDL	7
Merchandising Service WSDL	7
Overview	7
Operation XSD	7

Customer Credit Check Service	7
Functional Area	7
RSB Proxy WSDL	7
Merchandising Service WSDL	7
Overview	7
Operation XSD	8
Customer Order Fulfillment Service	8
Functional Area	8
RSB Proxy WSDL	8
Merchandising Service WSDL	8
Overview	8
Operation XSD	9
Customer Order Item Substitution Service	9
Functional Area	9
RSB Proxy WSDL	9
Merchandising Service WSDL	9
Business Overview	9
Assumptions	10
Operation XSD	10
Diff Management Service	10
Functional Area	10
RSB Proxy WSDL	10
Merchandising Service WSDL	10
Overview	10
Operation XSD	11
Inventory Back Order Service	11
Functional Area	11
RSB Proxy WSDL	11
Merchandising Service WSDL	11
Overview	11
Assumptions	12
Operation XSD	12
Inventory Lookup Service	12
Functional Area	12
RSB Proxy WSDL	12
Merchandising Service WSDL	12
Overview	12
Assumptions	13
Operation XSD	13
Item Management Service	13
Functional Area	13
RSB Proxy WSDL	13

Merchandising Service WSDL	13
Overview	13
Operation XSD	14
Pay Term Service	15
Functional Area	15
RSB Proxy WSDL	15
Merchandising Service WSDL	15
Overview	16
Operation XSD	16
Pricing Cost Service	16
Functional Area	16
RSB Proxy WSDL	16
Merchandising Service WSDL	16
Overview	17
Assumptions	17
Operation XSD	17
Purchase Order Management Service	17
Functional Area	17
RSB Proxy WSDL	17
Merchandising Service WSDL	18
Overview	18
Operation XSD	18
Report Locator Service	19
Functional Area	19
RSB Proxy WSDL	19
Merchandising Service WSDL	19
Overview	19
Operation XSD	20
Store Order Service	20
Functional Area	20
RSB Proxy WSDL	20
Merchandising Service WSDL	20
Overview	20
Assumptions	21
Operation XSD	21
Supplier Service	22
Functional Area	22
RSB Proxy WSDL	22
Merchandising Service WSDL	22
Overview	22
Operation XSD	23
Transfer Service	23

Functional Area	23
RSB Proxy WSDL	23
Merchandising Service WSDL	23
Overview	23
Operation XSD	24
Consumer Services	24
Customer Address Service	24
Functional Area	24
Overview	24
Customer Order Address Service	25
Functional Area	25
Overview	25
Get Drill Back Forward URL Service	26
Functional Area	27
Overview	27
GL Chart of Accounts Validation Service	27
Functional Area	27
Overview	27

5 ReSTful Web Services

Introduction	1
Common Characteristics	2
Security	2
Standard Request and Response Headers	2
Standard Error Response	3
Merchandising URL Paths	3
Sales Audit URL Paths	4
Fiscal Document Generation URL Paths	4
OAuth for REST Service Authentication	5
Date Format	6
Dynamic Response Field Filtering	6
'Include' Parameter	6
'Exclude' Parameter	6
Limited Availability During Nightly Batch Cycle	6
Custom HTTP Headers	7
Paging	8
Web Service APIs Process Flow	9
Configuring ReSTful Web Services for Integration with External Systems	9
File Transfer Services (FTS)	11
Business Overview	11
List Storage Prefixes	12

Business Overview	12
Service Type	12
ReST URL	12
Headers	12
Output	12
Retrieve PAR for Uploading Files	13
Business Overview	13
Service Type	13
ReST URL	13
Headers	13
Output	14
Uploading the Batch Incoming File	14
Retrieve PAR for Downloading Files	15
Business Overview	15
Service Type	15
ReST URL	15
Headers	15
Output	15
Downloading the File	16
Delete Files	16
Business Overview	16
Service Type	16
ReST URL	16
Headers	17
Output	17
List Files	17
Business Overview	17
Service Type	18
ReST URL	18
Headers	18
Input Parameters	18
Output	18
FTS Health Check	19
Business Overview	19
Service Type	19
ReST URL	19
Headers	19
Output	19
Merchandising to Omnichannel Services	19
REST Endpoint for Omni channel/Diff Group	20
Endpoint	20
Functional Area	20

Business Overview	20
Service Type	21
ReST URL	21
Input Parameters	22
Output	22
Elements in JSON Output	24
Table Impact	25
REST Endpoint for Omni channel/diff Ids	25
Endpoint	25
Functional Area	25
Business Overview	25
Service Type	26
ReST URL	26
Input Parameters	26
Output	27
Elements in JSON Output	29
Table Impact	30
REST Endpoint for Omni channel/Inventory/Store details	30
Endpoint	30
Functional Area	30
Business Overview	30
Service Type	30
ReST URL	31
Input Parameters	31
Output	31
Elements in JSON Output	34
Table Impact	35
REST Endpoint for Omni channel/Inventory/Warehouse details	35
Endpoint	35
Functional Area	35
Business Overview	35
Service Type	35
ReST URL	35
Input Parameters	36
Output	36
Elements in JSON Output	39
Table Impact	40
REST Endpoint for Omni channel/ Item details	40
Endpoint	40
Functional Area	40
Business Overview	40
Service Type	43

ReST URL	43
Input Parameters	43
Output	43
Elements in JSON Output	50
Table Impact	52
REST Endpoint for Omni channel/ Item UPC details	53
Endpoint	53
Functional Area	53
Business Overview	53
Service Type	53
ReST URL	53
Input Parameters	53
Output	54
Elements in JSON Output	61
Table Impact	62
REST Endpoint for Omni channel/ Merchandise Hierarchy details	63
Endpoint	63
Functional Area	63
Business Overview	63
Service Type	64
ReST URL	64
Input Parameters	64
Output	65
Elements in JSON Output	67
Table Impact	68
REST Endpoint for Omni channel/Org hier descendant's details	68
Endpoint	68
Functional Area	68
Business Overview	69
Service Type	70
ReST URL	70
Input Parameters	70
Output	70
Elements in JSON Output	72
Table Impact	73
REST Endpoint for Omni channel/Org hierarchy details	74
Endpoint	74
Functional Area	74
Business Overview	74
Service Type	75
ReST URL	75
Input Parameters	75

Output	76
Elements in JSON Output	78
Table Impact	79
REST Endpoint for Omni channel/refresh Date	80
Endpoint	80
Functional Area	80
Business Overview	80
Service Type	80
ReST URL	80
Input Parameters	81
Output	81
Elements in JSON Output	83
Table Impact	84
REST Endpoint for Omni channel/Warehouse details	84
Endpoint	84
Functional Area	84
Business Overview	85
Service Type	86
ReST URL	86
Input Parameters	86
Output	86
Elements in JSON Output	90
Table Impact	91
REST Endpoint for Omnichannel/related Item	92
Endpoint	92
Functional Area	92
Business Overview	92
Service Types	93
ReST URL	93
Input Parameters	93
Output	94
Elements in JSON Output	96
Table Impact	97
REST Endpoint for Omnichannel/Retail Store details	97
Endpoint	97
Functional Area	97
Business Overview	97
Service Type	99
ReST URL	99
Input Parameters	99
Output	99
Elements in JSON Output	103

Table Impact	104
REST Endpoint for Omnichannel/Dimension Type details	104
Endpoint	104
Functional Area	104
Business Overview	105
Service Type	105
ReST URL	105
Input Parameters	106
Output	106
Elements in JSON Output	108
Table Impact	109
REST Endpoint for Omnichannel/Dimension Value details	110
Endpoint	110
Functional Area	110
Business Overview	110
Service Type	111
ReST URL	111
Input Parameters	111
Output	111
Elements in JSON Output	113
Table Impact	114
REST Endpoint for Omnichannel/future Inventory	115
Endpoint	115
Functional Area	115
Business Overview	115
Service Type	116
ReST URL	116
Input Parameters	116
Output	117
Elements in JSON Output	119
Table Impact	120
REST Endpoint for Omnichannel/item Image	120
Endpoint	120
Functional Area	120
Business Overview	120
Service Type	122
ReST URL	122
Input Parameters	122
Output	123
Elements in JSON Output	125
Table Impact	126
REST Endpoint for Omnichannel/Item Initial Price details	126

Endpoint	126
Functional Area	126
Business Overview	127
Service Type	127
ReST URL	127
Input Parameters	127
Output	128
Elements in JSON Output	130
Table Impact	131
REST Endpoint for Omnichannel/Item Location details	132
Endpoint	132
Functional Area	132
Business Overview	132
Service Type	132
ReST URL	132
Input Parameters	132
Output	133
vatcodes JSON Output	139
Elements in JSON Output	139
Table Impact	140
REST Endpoint for Omnichannel/vat	141
Endpoint	141
Functional Area	141
Business Overview	141
Service Type	142
ReST URL	142
Input Parameters	143
Output	143
Elements in JSON Output	145
Table Impact	146
Merchandising Services	146
Administration	159
Data Privacy	160
Operations	176
System Options	210
Cost	216
Average Cost	216
Cost Changes	219
Pricing Cost	231
Customer Orders	234
Customer Order Available Inventory Download Service	234
Customer Order Inventory Backorder Upload Service	237

Customer Order Item Substitution Upload Service	240
Customer Order Upload Service	242
Deals	255
Manage Complex Deals	255
Manage Fixed Deals	256
Deals Download Service	260
Financials	263
Budgets	263
Currency	266
General Ledger	270
Terms	294
Transaction Data	306
Foundation	309
Banners and Channels Download Service	310
Location List Upload Service	312
Location Trait Upload Service	317
Suppliers and Partners	321
Tax Rules	399
Franchise Management	432
Customer Credit Status Upload Service	433
Franchise Order Upload Service	435
Import Management	451
Actual Landed Cost	452
Obligations	456
Inventory	488
Appointments	490
Inventory Download Service	501
Inventory Adjustments	512
Returns To Vendor	522
Shipments and Receipts	543
Stock Counts	589
Transfers and Allocations	600
Items	758
Item AI Services	762
Item Definition	772
Item Foundation	1223
Merchandise Hierarchy	1288
Copy Department Up Charges	1289
Class Download Service	1292
Class Upload Service	1323
Company Upload Service	1329
Department Download Service	1331

Department Upload Service	1382
Division Download Service	1411
Division Upload Service	1459
Group Download Service	1464
Group Upload Service	1512
Merchandise Hierarchy Download Service	1517
Subclass Download Service	1519
Subclass Upload Service	1567
Reclassification	1575
Organizational Hierarchy	1592
Get Company Closings	1593
Get Location Closings	1594
Get Locations for AIF Applications	1596
Manage Company Closings	1598
Manage Location Closings	1599
Manage Warehouse	1601
Organizational Hierarchy Download Service	1606
Organizational Hierarchy Upload Service	1626
Stores	1631
Warehouse	1747
Purchase Orders	1785
Mobile Workflow Purchase Order Download Service	1786
Mobile Workflow Purchase Order Upload Service	1820
Purchase Order Download Service	1830
Purchase Order Upload Service	1874
Replenishment	2004
Inventory Request Upload Service	2004
Replenishment Scheduled Update Upload Service	2007
Sales	2023
Sales Download Service	2023
Store Day User Upload Service	2034
Fiscal Document Generation Services	2044
Fiscal Pre Document Request Service	2044
Functional Area	2045
Business Overview	2045
Service Type	2045
ReST URL	2045
Input	2045
Output	2050
Table Impact	2051
Fiscal Document Request Service	2052
Functional Area	2052

Business Overview	2052
Service Type	2052
ReST URL	2052
Input	2053
Output	2060
Table Impact	2062
Fiscal Document Status Detail Request Service	2062
Functional Area	2063
Business Overview	2063
Service Type	2063
ReST URL	2063
Input Parameters	2063
Output	2064
Table Impact	2065
Next Fiscal Document Request Service	2065
Functional Area	2065
Business Overview	2065
Service Type	2065
ReST URL	2065
Input Parameters	2066
Output	2066
Table Impact	2078
Fiscal Document Process Status Request Service	2078
Functional Area	2078
Business Overview	2078
Service Type	2078
ReST URL	2078
Input	2079
Output	2081
Table Impact	2081
Fiscal Document Cancel Request Service	2082
Functional Area	2082
Business Overview	2082
Service Type	2082
ReST URL	2082
Input	2082
Output	2084
Table Impact	2084
Fiscal Document Detail Request Service	2084
Functional Area	2084
Business Overview	2084
Service Type	2085

ReST URL	2085
Input Parameters	2085
Output	2085
Table Impact	2097
Sales Audit Services	2097
Data Privacy	2098
Sales Audit Data Privacy Download Service	2098
Sales Audit Data Privacy Upload Service	2098
Store Day Summary	2098
Mobile Workflow Store Day Download Service	2099
Transactions	2109
Consignment Sales Transaction Update	2109
Transaction Download Service	2113
Transaction Upload Service	2125

6 Scheduled Integration

File-based Integration Overview	1
Standard File Layouts	1
Detail-Only Files	1
Master and Detail Files	1
Integration Batch Wrapper FAQ	3
For input files, are there any specific file format names?	3
How and where will files be moved?	3
When will files be unzipped?	4
What will happen to the files once they are processed?	4
Will processed file be archived?	4
What will happen if the file is rejected?	4
How will rejected files be reprocessed?	4
File Movement Flow	5
Input/Output File Naming Convention	6
Bulk Data Integration Overview	7
Outbound Scheduled Integration	7
Foundation Data	7
Brand Publication API (BDI_Brand_Fnd_PF_From_RMS_EOW_JOB)	8
Calendar Publication API (BDI_Calendar_Fnd_PF_From_RMS_EOW_JOB)	10
Code Detail Publication API (BDI_CodeDetail_Fnd_PF_From_RMS_JOB)	11
Code Head Publication API (BDI_CodeHead_Fnd_PF_From_RMS_JOB)	12
Company-wide Closings and Company Closed Exceptions (BDI_CompanyClosed_Fnd_PF_From_RMS_JOB)	13
Currency Conversion Rates Publication API (BDI_CurrConvRates_Fnd_PF_From_RMS_EOW_JOB)	14
Delivery Slot Publication API (BDI_DeliverySlot_Fnd_PF_From_RMS_JOB)	15

Diff Group Export (export_diffgrp.ksh)	16
Diff Group Publication API (BDI_DiffGrp_Fnd_PF_From_RMS_JOB)	17
Diff ID and Type Export (export_diffs.ksh)	18
Diff ID Publication API (BDI_Diff_Fnd_PF_From_RMS_JOB)	19
Finisher Address Publication API (BDI_FinisherAddr_Fnd_PF_From_RMS_JOB)	20
Location Closed Publication API (BDI_LocClosed_Fnd_PF_From_RMS_JOB)	21
Merch Hierarchy Publication API (BDI_MerchHier_Fnd_PF_From_RMS_JOB)	22
Merchandise Hierarchy Export (export_merchhier.ksh)	23
Organization Hierarchy Publication API (BDI_OrgHier_Fnd_PF_From_RMS_JOB)	24
Organizational Hierarchy Export (export_orghier.ksh)	25
Partner Address Publication API (BDI_PartnerAddr_Fnd_PF_From_RMS_JOB)	26
Partner Org Unit Publication API (BDI_PartOrgUnit_Fnd_PF_From_RMS_JOB)	27
Partner Publication API (BDI_Partner_Fnd_PF_From_RMS_JOB)	28
Store Address Publication API (BDI_StoreAddr_Fnd_PF_From_RMS_JOB)	29
Store Hours Publication API (BDI_StoreHours_Fnd_PF_From_RMS_JOB)	30
Store Publication API (BDI_Store_Fnd_PF_From_RMS_JOB)	31
Stores Export (export_stores.ksh)	32
Supplier Address Publication API (BDI_SupplierAddr_Fnd_PF_From_RMS_JOB)	33
Sups Publication API (BDI_Supplier_Fnd_PF_From_RMS_JOB)	34
Tax Download - Brazil (taxdnld)	35
Ticket Download (tcktdnld)	37
UDA Publication API (BDI_Uda_Fnd_PF_From_RMS_JOB)	39
UDA Values Publication API (BDI_UdaValues_Fnd_PF_From_RMS_JOB)	40
UOM Class Publication API (BDI_UomClass_Fnd_PF_From_RMS_JOB)	41
UOM Conversion Publication API (BDI_UomConversion_Fnd_PF_From_RMS_JOB)	42
VAT Codes, Regions, and Rates Export (export_vat.ksh)	43
VAT Publication API (BDI_Vat_Fnd_PF_From_RMS_JOB)	44
Warehouse (BDI_Wh_Fnd_PF_From_RMS_JOB)	45
Warehouse Address Publication API (BDI_WhAddr_Fnd_PF_From_RMS_JOB)	46
Items	47
Item Image Publication API (BDI_ItemImage_Fnd_PF_From_RMS_JOB)	48
Item Location Export (export_itemloc.ksh)	49
Item Location Publication API (BDI_ItemLoc_Fnd_PF_From_RMS_JOB)	50
Item Master Export (export_itemmaster.ksh)	51
Item Master Publication API (BDI_ItemHdr_Fnd_PF_From_RMS_JOB)	52
Item Supplier Country Dimensions Publication API (BDI_ItSupCtryDim_Fnd_PF_From_RMS_JOB)	53
Item Supplier Country Publication API (BDI_ItSupCtry_Fnd_PF_From_RMS_JOB)	54
Item Supplier Manufacturing Country Publication API (BDI_ItSupManCtry_Fnd_PF_From_RMS_JOB)	55
Item Supplier Publication API (BDI_ItemSupp_Fnd_PF_From_RMS_JOB)	56
Item Supplier UOM Publication API (BDI_ItemSuppUom_Fnd_PF_From_RMS_JOB)	57

Item VAT Rates Export (export_itemvat.ksh)	58
Pack Item Publication API (BDI_PckitemBrkout_Fnd_PF_From_RMS_JOB)	59
Populate ITEM_LOC_SOH_EOD table for RDE Extract (maintain_ilseed_hist)	60
Populate ITEM_SUPP_COUNTRY_HIST and ITEM_SUPP_COUNTRY_LOC_HIST table (maintain_isc_hist)	61
POS Configuration Data to 3rd Party POS (poscdnld)	62
Price History Publication API (BDI_PriceHist_Fnd_PF_From_RMS_JOB)	67
Related Items Export (export_relitem.ksh)	68
Related Item Publication API (BDI_RelatedItem_Fnd_PF_From_RMS_JOB)	69
UDA Item Date Publication API (BDI_UdaltemDate_Fnd_PF_From_RMS_JOB)	70
UDA Item FF Publication API (BDI_UdaltemFF_Fnd_PF_From_RMS_JOB)	71
UDA Item LOV Publication API (BDI_UdaltemLov_Fnd_From_RMS_JOB)	72
Financials	73
Daily or Weekly Download of Stock Ledger Data (stlgdnld)	74
Finance General Ledger to RFI (BDI_RFI_FinGenLdgr_Tx_PF_From_RMS_JOB)	78
Financial Data Publish Extract (fif_gl_publish_extract)	79
Fixed Deal Income (dealfinc)	80
Franchise Billing Extract (wfbillex.ksh)	81
General Ledger Cross Reference Dynamic Mapping (fif_gl_dynamic_map)	87
Invoice Matching Accounts Payable Detail Outbound Publish Extract (fif_ap_detail_publish_extract)	88
Invoice Matching Accounts Payable Header Publish Extract (fif_ap_head_publish_extract)	89
Item/Location Daily Stock Ledger Transactions (fifglnd1)	90
Monthly Stock Ledger Transactions (fifglnd3)	92
Open to Buy Download Stock Ledger (otbdlsal)	93
Populate FLASHBACK_SNAPSHOT_INFO table (stockledger_switchtime)	102
Populate IF_TRAN_DATA_HIST and IF_FUTURE_TRAN_DATA_HIST tables for RDE Extract (maintain_iftd_hist)	103
RMS Financial Data Publish (fif_gl_publish)	103
Rolled Up Daily Stock Ledger Transactions (fifglnd2)	104
Stage G/L Extracts (gl_extract.ksh)	106
Tran Data Publication (BDI_TranData_Tx_PF_From_RMS_EOW_JOB)	108
Ordering and Inventory	109
Purchasing	109
Import Management	131
Invoices	161
Inventory	171
Planning and Forecasting	180
Brand Extract to Planning (BDI_RPAS_Brand_Fnd_PF_From_RMS_JOB)	182
Calendar Extract to Planning and Forecasting (BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB)	184

Currency Rates Extract to Planning and Forecasting (BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB)	187
Differentiator Extract to Planning (BDI_RPAS_Diff_Fnd_PF_From_RMS_JOB)	190
Inventory Extract to Planning (BDI_MFP_Inventory_Tx_PF_From_RMS_JOB)	192
Merchandise Hierarchy and Item Extract to Planning and Forecasting (BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB)	195
On Order Extract to Planning (BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB)	200
Organization Hierarchy Extract to Planning and Forecasting (BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB)	202
Out of Stock Extract to Forecasting (BDI_RDF_StockOut_Tx_PF_From_RMS_JOB)	206
Store Extract to Planning and Forecasting (BDI_RPAS_Store_Fnd_PF_From_RMS_JOB)	209
Supplier Extract to Planning (BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB)	212
Transaction Data Extract to Planning (BDI_MFP_TranData_Tx_PF_From_RMS_JOB)	214
UDA Extract to Planning (BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB)	220
UDA Item Extract to Planning and Forecasting (BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB)	222
Weekly Sales Extract to Forecasting (BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB)	225
Sales Audit	227
Download from Sales Audit to Account Clearing House (ACH) System (saexpach)	228
Download of Escheated Vouchers from Sales Audit for Payment (saescheat)	233
Export DSD and Escheatment from Sales Audit to Invoice Matching (saexpim)	234
Export from Sales Audit to Oracle Retail Analytics (saexport_sales_to_dw)	236
Export from Sales Audit to Oracle Retail Insights (saexpdw)	238
Export Inventory Reservation/Release for In Store Customer Order & Layaway Transactions (saordinvexp)	267
Export of POS transactions from Sales Audit to Merchandising (saexprms)	270
Export of POS Transactions from Sales Audit to Merchandising Based on Direct Table Load (saexpprocsales)	276
Export of Revised Sale/Return Transactions from ReSA to SIM/SIOCS (saexpsim)	279
Export of Revised Sale/Return Transactions from Sales Audit to Store Inventory Management (saexport_sales_to_sim)	283
Export to Universal Account Reconciliation System from Sales Audit (saexpuar)	284
Extract of POS Transactions by Store/Date from Sales Audit for Web Search (ang_saplgen)	285
Post User Defined Totals from Sales Audit to General Ledger (saexpgl)	287
Inbound Scheduled Integration	288
Item, Cost, and Price	288
Upload Competitor's Prices (cmpupld)	288
Upload Items and Cost Changes (iindbatch.ksh)	292
Ordering and Inventory	293
Purchasing	293

Import Management	328
Stock Counts	370
Franchise	375
Other Inventory	394
Fiscal Document Generation (FDG)	403
Fiscal Document Upload into FDG (fdg_reim_job)	403
Sales Processing	404
Sales Audit	405
Sales Posting	473
Forecasting	482
Daily Demand Item Forecast Subscription API	482
Weekly Demand Item Forecast Subscription API	483
Weekly/Daily Item Forecast Upload (load_item_forecast)	483

Send Us Your Comments

Oracle® Retail Merchandising Inbound and Outbound Integration Guide

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note

Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Technology Network Web site. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at <http://www.oracle.com>.

Preface

The *Oracle Retail Merchandising Foundation Cloud Service Inbound and Outbound Integration Designs* provides critical information about Integration with Oracle Retail Merchandising System, including the following:

- Publication designs which describe, on a technical level, how Merchandising publishes messages.
- Subscription designs which describe, on a technical level, how Merchandising subscribes to messages.

Audience

This guide is for:

- Systems administration and operations personnel
- Systems analysts
- Integrators and implementers
- Business analysts who need information about Merchandising System processes and interfaces

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

<https://support.oracle.com>

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create

- Exact error message received
- Screen shots of each step you take

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times not be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Technology Network Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Technology Network at the following URL:

<http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html>

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-02 is an updated version of a document with part number E123456-01.

If a more recent version of a document is available, that version supersedes all previous versions.

Oracle Retail Documentation on the Oracle Help Center (docs.oracle.com)

Oracle Retail product documentation is also available on the following Web site:

<https://docs.oracle.com/en/industries/retail/index.html>

(Data Model documents can be obtained through My Oracle Support.)

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introduction

This volume contains details about Merchandising and Sales Audit integrations. These integrations fall into four main categories:

- **Message-based Integration** - These are covered in two sections: publication and subscription. Publication covers RIB messages published from Merchandising to other solutions. Subscription covers RIB message that are subscribed to by Merchandising from other solutions.
- **SOAP Web Services** - This chapter provides a summary of the provider and consumer SOAP services supported by Merchandising and Sales Audit, including details on security, URLs, and payload information.
- **ReSTful Web Services** - This chapter provides a summary of the ReST services supported by Merchandising and Sales Audit, including details on security, URLs, and payload information.
- **Scheduled Integration** - This chapter provides a summary of integrations that are scheduled either to be run once per day or periodically throughout the day. There are generally two types of integrations - those that expect or produce files and those that move data between integration tables, also referred to as Bulk Data Integration (BDI).

2

RIB Publication Designs

This chapter provides an overview of the RIB publication APIs used in by Merchandising.

Allocations Publication API

This section describes the allocations publications API.

Functional Area

Allocations

Business Overview

Merchandising is responsible for communicating allocation information with external systems such as a store inventory system (SIOCS, for example) and a warehouse management system, like Oracle WMS Cloud.

There are several ways in which allocation information can be created in Merchandising:

- Through integration with the Allocation Cloud Service
- Through Merchandising replenishment, where cross dock orders generate allocations
- Through the Allocation Subscription API, where a third-party system can create allocations and send to Merchandising for execution.

Allocations can be created from a virtual warehouse to any type of stockholding location in Merchandising, including other virtual warehouses, and to both company and franchise stores. Allocations include a store type and stockholding indicator at the detail level when allocating to stores, to allow the store and warehouse inventory management system to filter out the data irrelevant to their respective systems. When allocating to a franchise store, the linked franchise orders are not published; only the allocation itself is published. When allocating to another warehouse, the allocation quantities are summed up to the physical warehouse level and the physical warehouse is what is communicated in the integration.

An allocation and its details are not published from Merchandising until it is approved. Modified and deleted allocation information is also sent. Allocation header modification messages will be sent if the status of the allocation is changed to approved (A) or closed (C) or if the allocation release date is changed. Allocation detail modification messages will be sent if the allocated quantity is changed. A header delete message signifies that the completed allocation has been deleted.

Note

Allocations, when published to external systems, are combined in the RIB with transfers (published in the Transfer Publication API) into a combined Stock Order Publication message.

New Allocations

When an allocation is created, an Allocation Create message request is queued. The Allocation Create message is a flat message containing a full snapshot of the allocation at the time the message is published. The message will not be sent until the allocation has been approved.

The allocation create message contains the following:

Allocation Header

Allocation Header

Message Element	Included?	Notes
Allocation Number	Always	Contains the number that uniquely identifies the allocation within the system.
Document Type	Always	This is the type of stock order. This value will be 'A' to signify an allocation. This will differentiate allocations and transfers in the Stock Order Publication API subscribed to by stores and warehouses.
Physical WH	Always	The physical warehouse location from which the allocation will originate.
Warehouse	Always	Contains the number of the virtual warehouse where the allocation will originate.
Item	Always	Specifies the item on the allocation.
Pick Not Before Date	Optional	Contains the earliest date on which the allocation should start being picked for the allocation. This will contain the release date on the allocation or the not before date on the associated purchase order.
Pick Not After Date	Optional	This value is calculated by adding x days to the release date on the allocation. The number of days to add is defined in the description column on code_detail for code DATE and code type DEFT.
Order Type	Optional	This field contains the type of order. Allocations created against Purchase Orders will be marked as PREDIST order types. Allocations created against Warehouse stock will be populated with the DEFAULT_ORDER_TYPE from the SYSTEM_OPTIONS table which can be AUTOMATIC, MANUAL or WAVE.;
Order No	Optional	Contains the order number to which the allocation applies. This is only populated for order-based allocations.
Order Document Type	Always	Always 'P'.
Event	Optional	This field contains the event to which the promotion belongs to. This is an optional field that provides a method to group promotions together for tracking and reporting purposes.
Event Description	Optional	This field contains the description of the promotional event.
Priority	Always	A value which indicates the priority of an allocation. This value will always be 1.

Message Element	Included?	Notes
Ticket Type ID	Optional	This field contains a character string which uniquely identifies the ticket type which is associated with the item. This is only populated if the ticket associated with the item on the allocation is configured to be printed upon the receipt of the purchase order for order-based allocations.
Context Type	Optional	This field holds the code for the reason for the allocation. For example, it could indicate the allocation was created for an upcoming promotion. Valid context codes are defined in Codes and Descriptions under the code type CNTX.
Context Value	Optional	Contains a character string relating to the context type.
Allocation Status	Always	Contains the code for the allocation status.
Allocation Details	Always	Child node

Allocation Detail

Message Element	Required?	Notes
Physical To Location	Always	This field contains the physical location to which the allocation is being sent. If the location type is a store, this will be the same as the To Location.
To Location	Always	This field contains the location to which the allocation is being sent. The location type field determines if the location is a store or a virtual warehouse.
Location Type	Always	This field contains the type of location in the To Location field. Valid values are: • S - Store • W - Warehouse
Store Type	Optional	If the To Location is a store, this field contains the type of store. Valid values are: • C - company store • F - franchise store
Stockholding	Optional	If the To Location is a store, this field indicates if the store is a stockholding location or not. Valid values are: yes (Y) or no (N).
Quantity Allocated	Always	Contains the total quantity of the item allocated to the to location.
Price	Optional	This field holds the unit retail in the selling unit of measure for the item/to location combination. This field is stored in the local currency of the to location.
Selling UOM	Optional	This field holds the selling unit of measure for the item's single-unit retail at the to location.
Priority	Always	A value which indicates the priority of an allocation detail. This value will always be 1.
Store Order Multiple	Always	This column contains the multiple in which the item needs to be shipped from a warehouse to the store.

Message Element	Required?	Notes
In-Store Date	Optional	Indicates the date that the item needs to be at the store.
Rush Flag	Optional	Indicates if there is a rush on shipping this item to the destination location.
Allocation Detail Component Items	Optional	Child node

Allocation Detail Component Items

This node is only included if the item being allocated is a pack item.

Message Element	Required?	Notes
Component Item	Always	This field contains the alphanumeric identifier of an item within the pack.
Component Price	Always	This field holds the unit retail in local currency and in the component selling unit of measure for the component item at the destination location.
Component Selling UOM	Always	This field holds the selling unit of measure for the item's single-unit retail at the to location.

New Allocation Details

Creating new allocations details for an existing header triggers a message to be sent to notify external systems of the changes. The message for new allocation details contains the same information as the new allocation create message.

Updated Allocations

Allocation updates trigger a message to be sent to notify external systems based on updates made at the allocation header and/or detail level. The message for updated allocations contain the same information as the new allocation create message.

Header Only Updates

Allocations are published when updates at the header level are made to the release date or for certain allocation status updates. Only changed statuses where the final status is Approved (A) or Closed (C) and the old status is not Externally Closed (X) will trigger a header update to be published. When updates are made at the allocation header level, only header information is published.

Detail Only Updates

When updates are made at the allocation detail level without changes to the header information, the full header and detail information are published. A detail update will be published when the quantity allocated has changed.

Full Message Updates

In cases where the system receiving allocations cannot support just receiving the changes, another option is provided that can resend the full allocation details whenever there is a

change. This will be published when the system option Publish Full Objects (PUB_FULL_OBJECTS_IND) is set to Deltas and Full (Y). This message will contain the allocation header and its detail level information. This is used for Oracle WMS Cloud integration.

Deleted Allocations

Allocations can be deleted when they are in Approved or Closed status. When an allocation delete message is triggered, a message is sent to external systems to notify them of the changes via an allocation header delete message.

Detail Only Deletes

When allocation details are deleted, a message is sent to external systems to notify them of the change. Both header and detail information are included in the published message.

The allocation delete message contains the following:

Table 2-1 Allocation Header

Message Element	Included?	Notes
Allocation Number	Always	Contains the numeric identifier that uniquely identifies the allocation within the system.
Document Type	Always	This is the type of stock order. This value will be 'A' to signify an allocation. This will differentiate allocations and transfers in the Stock Order Publication API subscribed to by stores and warehouses.
Physical Warehouse	Always	Contains the numeric identifier of the physical warehouse location where the allocation will originate.
Warehouse	Always	Contains the numeric identifier of the virtual warehouse location where the allocation will originate.
Item	Always	Unique alphanumeric value that identifies the item.
Allocation Detail	Optional	Child node

Table 2-2 Allocation Detail

Message Element	Required?	Notes
To Location	Always	This field contains the location to which the allocation is being sent. The location type field determines if the location is a store or a virtual warehouse.
Location Type	Optional	This field contains the type of location in the To Location field. Valid values are S - store and W - warehouse.
Store Type	Optional	If the To Location is a store, this field contains the type of store. Valid values are: C - company store, F - franchise store.
Stockholding Indicator	Optional	If the To Location is a store, this field indicates if the location is a stockholding store or not. Valid values are: yes (Y) or no (N).

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the particular message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish these messages is available. In case the error is a fatal error, for example, when changes to data have already been made, a status of Error (E) is sent to the RIB and the message status in the queue will be in Error status. The error message as well as the object containing the allocation number and details is returned to Merchandising.

Message XSD

Here are the filenames that correspond with each message type. Please consult the mapping documents for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
AllocCre	Allocation Create Message	AllocDesc.xsd
AllocDtlCre	Allocation Detail Create Message	AllocDesc.xsd
AllocHdrMod	Allocation Header Modify Message	AllocDesc.xsd
AllocDtlMod	Allocation Detail Modify Message	AllocDesc.xsd
AllocFullRep	Allocation Full Replacement Message	AllocDesc.xsd
AllocDel	Allocation Delete Message	AllocRef.xsd
AllocDtlDel	Allocation Detail Delete Message	AllocRef.xsd

ASN Outbound Publication API

This section describes the ASNOUT Publication API.

Functional Area

Inventory

Business Overview

The Advanced Shipping Notification (ASN) outbound message is used to communicate the shipment of merchandise against transfers or allocations. This message is published by Merchandising to stores or warehouses.

Merchandising supports the following shipping functionality:

- On-line Shipping/Receiving.
- Franchise Order Shipment and Return.

On-line Shipping/Receiving

Two system options (ship_rcv_store and ship_rcv_wh) are used to control whether Merchandising on-line shipment/receiving functionality is enabled.

- Ship_rcv_store = 'Y' means a store inventory management application, such as Oracle Retail Store Inventory Operations Cloud Service (SIOCS), is NOT installed and shipping/receiving for **stores** will be done in RMS.
- Ship_rcv_wh = 'Y' means a warehouse management system, such as Oracle Cloud WMS, is NOT installed and shipping/receiving for **warehouses** will be done in Merchandising.

If either (but not both) of these indicators is set to 'Y', shipments created in Merchandising should be published to the RIB to allow the integration subsystem application to have visibility to the corporately created shipment.

The possible scenarios for on-line shipping/receiving:

Store Inventory System Installed (Yes/No)	Warehouse Management Installed (Yes/No)	System Options Settings	RMS Publishes Shipments (Yes/No)	Apps to subscribe to the message
Yes	Yes	Ship_rcv_store = N Ship_rcv_wh = N	No	No
No	No	Ship_rcv_store = Y Ship_rcv_wh = Y	No	No
Yes	No	Ship_rcv_store = N Ship_rcv_wh = Y	Yes – for warehouse-to-store shipments	Store Inventory System
No	Yes	Ship_rcv_store = Y Ship_rcv_wh = N	Yes – for store-to-warehouse shipments	Warehouse Management System

Merchandising on-line shipping can involve a customer order transfer . For a customer order transfer, customer order number and fulfillment order number are included in the published information.

Franchise Order Shipment and Return

Franchise stores are a special kind of stores that are not 'owned' by the company; therefore, any shipment to a franchise store is considered a sale. From Merchandising, franchise stores can order goods from company stores or warehouses; they can also return goods back to company stores or warehouses. These orders and returns are created as transfers in Merchandising.

Merchandising supports two kinds of franchise stores – stockholding franchise stores (in which Merchandising manages inventory and financials like regular stores) and non-stockholding franchise stores (in which Merchandising does NOT manage inventory and financials).

The Store Inventory Operations Cloud Service (SIOCS) manages transactions for stockholding franchise stores, but not for non-stockholding franchise stores. The Shipping and Receiving of non-stockholding franchise orders and returns are handled within Merchandising from the Store perspective even if SIOCS is installed.

For warehouses, if a franchise return from a non-stockholding franchise store is to be processed, the warehouse management system (WMS) will require an ASN against which to receive. Because Merchandising automatically creates the shipment for non-stockholding stores upon the approval of a franchise return, Merchandising needs to publish those shipments for the WMS. Similar to on-line Shipping/Receiving, Merchandising publishes shipments of non-stockholding Franchise Returns to warehouses as ASNOut messages.

Advance Shipment Notification

Shipments created in Merchandising cannot be modified. Upon saving a shipment, the entire shipment is published from Merchandising as one ASNOut message. As a result, Merchandising only needs to support the ASNOut create message type for shipment publishing.

Table 2-3 Advance Shipment Notification Header

Message Element	Included?	Notes
Schedule Number	No	Contains the Schedule identifier number. Not used by RMS.
Auto Receive	Always	Contains the flag to enable auto-receiving in the store inventory system (e.g. SIOCS). Defaulted to "No".
To Location	Always	Contains the location that the shipment will be delivered to.
To Location Type	Always	This field contains the location type of the to_location field. Valid values are 'S' = store, 'E' = finisher, and 'W' = warehouse.
To Store Type	Always	Indicates the store type of the to_location. This will only be populated if to_loc_type is 'S'. Valid values are: 'C' = company store, 'F' = franchise store.
To Stockholding Indicator	Optional	Indicates if the to_location is stockholding or not. Only populated if to_loc_type is 'S'. Valid values are: 'Y' = yes, 'N' = no.
From Location	Always	For transfer and allocation shipments, this field will hold the location from which the shipment was sourced.
From Location Type	Always	This field contains the location type of the from_location field. Valid values are 'S' = store and 'W' = warehouse.
From Store Type	Optional	Indicates the store type of the from_location. This will only be populated if from_location is 'S'. Valid values are: 'C' = company store, 'F' = franchise store. Publications from RWMS will always be NULL for this field.
From Stockholding Indicator	Optional	Indicates if the from_location is stockholding or not. Only populated if from_location is 'S'. Valid values are: 'Y' = yes, 'N' = no. Publications from RWMS will always be NULL for this field.
ASN Number	Always	Holds the bill of lading number associated with a shipment.
ASN Type	Always	Not used by RMS. Set to C for container when ship to location is a store, T for tare when ship to location is a warehouse.
Container Quantity	Always	Contains the number of boxes associated with the shipment.

Table 2-3 (Cont.) Advance Shipment Notification Header

Message Element	Included?	Notes
BOL Number	Always	Holds the transaction sequence number on the message from the transfer shipment confirmation process.
Shipment Date	Always	This field contains the date the transfer or PO was shipped.
Estimated Arrival Date	Always	This field contains the estimated arrival date of a vendor PO shipment. It is updated by EDIUP856. It is used for vendor/lead time analysis.
Ship Address 1	Optional	Value of addr.add_1 for store/warehouse number in addr.key_value_1 and addr.module 'ST' (if to_loc_type is store) or 'WH' (if to_loc_type is warehouse).
Ship Address 2	Optional	Value of addr.add_2 for store/warehouse number in addr.key_value_2 and addr.module 'ST' (if to_loc_type is store) or 'WH' (if to_loc_type is warehouse).
Ship Address 3	No	Not used by RMS.
Ship Address 4	No	Not used by RMS.
Ship Address 5	No	Not used by RMS.
Ship City	Optional	Value of addr.city for store/warehouse number in addr.key_value_1 and addr.module 'ST' (if to_loc_type is store) or 'WH' (if to_loc_type is warehouse).
Ship State	Optional	Value of addr.state for store/warehouse number in addr.key_value_1 and addr.module 'ST' (if to_loc_type is store) or 'WH' (if to_loc_type is warehouse).
Ship ZIP	No	Not used by RMS.
Ship Country Id	Optional	Value of addr.country_id for store/warehouse number in addr.key_value_1 and addr.module 'ST' (if to_loc_type is store) or 'WH' (if to_loc_type is warehouse).
Trailer Number	Always	Not used by RMS. Defaulted to 1.
Seal Number	No	Not used by RMS.
Trans Shipment Number	No	Not used by RMS.
Comments	Optional	Contains any miscellaneous comments about the shipment.
Carrier Code	Always	Contains the courier that will deliver the shipment. Defaulted to "DC".
Carrier Service Code	No	Contains the service level code for the courier that will deliver the shipment.
System Code	No	The code identifying the system associated with the location.
From Location Warehouse	No	For transfer and allocation shipments, this field will hold the virtual warehouse from which the shipment was sourced.

Table 2-4 Advance Shipment Notification Distro

Message Element	Included?	Notes
Distro Number	Always	Specifies the transfer or allocation number associated with the shipment/item/carton.

Table 2-4 (Cont.) Advance Shipment Notification Distro

Message Element	Included?	Notes
Distro Doc Type	Always	Specifies what the distro_nbr field corresponds to in RMS: V, D, and T specify transfer, A specifies Allocation.
Customer Order Number	Optional	This is the customer order number that was generated by OMS and contains multiple fulfillment numbers.
Fulfill Order Number	Optional	Based on a customer order - OMS will generate fulfillments to specific locations based on availability. RMS will generate Transfers based on the fulfillment request.
Consumer Direct	No	Not used by RMS.
Comments	No	Not used by RMS.

Table 2-5 Advance Shipment Notification Carton

Message Element	Included?	Notes
Final Location	No	Not used by RMS.
Container Id	Always	Identifies the UCC-128 carton number for shipments originating from the Advance Shipment Notification process as carton shipments. This field will be zero for all shipments that are not at a carton level.
Container Weight	No	Not used by RMS.
Container Length	No	Not used by RMS.
Container Width	No	Not used by RMS.
Container Height	No	Not used by RMS.
Container Cube	No	Not used by RMS.
Expedite Flag	Always	Indicates if the shipment should be expedited. For transfers, the value will be 'Y' if the transfer's freight_code is 'E'; otherwise 'N'. For allocations, the field is mapped to alloc_detail.rush_flag.
In Store Date	Always	Contains the date to be included in the RMS publication to the RIB for communication to the warehouse. For transfers, the value is always NULL. For allocations, the value is alloc_detail.in_store_date.
Tracking Number	No	This is a unique tracking number that is used to track containers through carrier's system.
Freight Charge	No	Not used by RMS.
Master Container Id	No	Not used by RMS.
Comment	No	Contains any comments about the shipment container.
Weight	No	Actual weight shipped for the container.
Weight Unit of Measure	No	Unit of measurement for weight (e.g. pounds, kilograms) that was shipped.
Carrier Shipment Number	No	This field represents the shipment number that manifest systems use to group multi-container shipments for a carrier. Typically a 1 of 4 type message.
Original Item Id	No	The ID of the item being replaced. Populated only when this record is for a substitute item on a customer order.

Table 2-6 Advance Shipment Notification Item

Message Element	Included?	Notes
Item Id	Always	Unique identifier for the item.
Unit Quantity	Always	Contains the number of items expected to be received based on the supplier's advance shipment notification for this item/shipment combination.
Gross Cost	No	Contains the gross cost.
Priority Level	No	Not used by RMS.
Order Line Number	No	Is used to carry the customer order line number value for customer orders. Is a derived value for non-customer orders.
Lot Number	No	Not used by RMS.
Final Location	Always	The destination location of the shipment.
From Disposition	No	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.
To Disposition	No	Not used by RMS.
Voucher Number	No	Contains the voucher number.
Voucher Expiration Date	No	Not used by RMS.
Container Quantity	Always	Not used by RMS. Defaulted to 1.
Comments	No	Contains any comment about the item in the shipment.
Unit Cost	Always	Contains the unit cost of the item in the shipment.
Base Cost	No	This value will be used to get the base cost (BC) from RFM for a transfer, which will flow into RMS.
Weight	No	Actual weight shipped.
Weight Unit of Measure	No	Unit of measurement for weight (e.g. pounds, kilograms) shipped.

Table 2-7 Advance Shipment Notification UIN

Message Element	Included?	Notes
UIN	No	Universal Identification Number

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Here is the filename that corresponds with the message type. Please consult the RIB documentation for this message type to get a detailed picture of the composition of the message.

Message Types	Message Type Description	XML Schema Definition (XSD)
asnoutcre	ASN Out Create Message	ASNOutDesc.xsd

Available Inventory for Store and Warehouse Publication API

This section describes the Store and Warehouse Publication API.

Functional Area

Inventory

Business Overview

Merchandising publishes store and warehouse available inventory information to support the order management system requirements for ordering. Internally, Oracle Retail Order Broker (OROB) subscribes to this information. The following criteria must be met in order to publish inventory information when the stock on hand on a store or warehouse is updated:

- Items must be sellable
- Items must be a customer orderable location (based on the flags at the store and virtual warehouse level)
- Locations must be stockholding

This API always publishes inventory for virtual warehouses but will publish store information only based on the setting of the system option Publish Store Available to Sell Updates. If checked, then store information will also be published.

Item-Location Level Available to Sell Message

Message Element	Included?	Notes
Item	Always	The transaction item whose inventory information is being communicated.
Location	Always	Store or virtual warehouse where the item is to be found.
Location Type	Always	This contains the type of location. Valid values are store (S) or warehouse (W).
Available to Sell Quantity	Always	Contains the updated quantity available to sell for the item-location in the standard unit of measure. Available to sell is calculated as: Stock on Hand - (Transfer Reserved + Customer Reservations + Non-sellable + return to vendor + in-progress sales quantity) All values above include also the pack component quantity, where applicable.

Message Element	Included?	Notes
System Code	Always	Identifies the originating application, which is required by some subscribing systems. This value is defaulted from the system option Integration System Code.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
COInvAvailMod	Customer Order Store Inventory Available	COInvAvailDesc.xsd

Available Inventory for Store Publication API

This section describes the Store Publication API.

Functional Area

Foundation Data

Business Overview

The existing bulk store inventory feed needs to be updated to better support OMS requirements for ordering. The current bulk inventory feed does not consider customer orderable locations and does not consider pack component inventory. The following updates are required:

- Consider only sellable items (item_master.sellable_ind = Y)
- Include only stores that are customer orderable locations (store.customer_order_loc_ind = Y) and stockholding (store.stockholding_ind = Y)

The revised calculation should be as follows:

- Additions:
 - Stock On Hand
- Subtractions:
 - Transfer Reserved Quantity

- Customer Reserved Quantity
- Non-Sellable Quantity
- RTV Quantity

Package Impact

File name: rmsmf_m_coinvavails/b.pls

Function Level Description - ADDTOQ

```
Function: ADDTOQ(
    O_error_message      OUT  VARCHAR2,
    I_message_type       IN   ITEM_LOC_SOH_MFQUEUE.MESSAGE_TYPE%TYPE,
    I_item_loc_soh_record IN   ITEM_LOC_SOH%ROWTYPE)
```

This public function puts an item location stock on hand message on ITEM_LOC_SOH_MFQUEUE for publishing to the RIB. It is called from item_loc_soh trigger where loc_type = 'S'. It also checks if the item is a sellable item and it also checks if the store is a customer_order_loc_ind = 'Y'.

Function Level Description - GETNXT

```
Procedure: GETNXT
(O_status_code      OUT  VARCHAR2,
 O_error_msg        OUT  VARCHAR2,
 O_message_type     OUT  VARCHAR2,
 O_message          OUT  RIB_OBJECT,
 O_bus_obj_id       OUT  RIB_BUSOBJID_TBL,
 O_routing_info     OUT  RIB_ROUTINGINFO_TBL,
 I_num_threads      IN   NUMBER DEFAULT 1,
 I_thread_val       IN   NUMBER DEFAULT 1)
```

This public procedure is called from the RIB to get the next messages. It performs a cursor loop on the unpublished records on the ITEM_LOC_SOH_MFQUEUE table (PUB_STATUS = 'U').

If any exception is raised in GETNXT, including the exception raised by an unsuccessful call to PROCESS_QUEUE_RECORD, HANDLE_ERRORS is called.

Function Level Description - PUB_RETRY

```
Procedure: PUB_RETRY
(O_status_code      OUT  VARCHAR2,
 O_error_msg        OUT  VARCHAR2,
 O_message          OUT  RIB_OBJECT,
 O_message_type     IN OUT VARCHAR2,
 O_bus_obj_id       IN OUT RIB_BUSOBJID_TBL,
 O_routing_info     IN OUT RIB_ROUTINGINFO_TBL)
```

This public procedure performs the same tasks as GETNXT except that it only loops for a specific row in the ITEM_LOC_SOH_MFQUEUE table. The record on ITEM_LOC_SOH_MFQUEUE must match the passed in sequence number (contained in the ROUTING_INFO).

Function Level Description - PROCESS_QUEUE_RECORD (local)

This private function controls the building of Oracle Objects (DESC or REF) given the business transaction's key values and a message type. It contains all of the shared processing between GETNXT and PUB_RETRY.

Function Level Description - HANDLE_ERRORS (local)

This private procedure is called from GETNXT and PUB_RETRY when an exception is raised. l_seq_no is the sequence number of the driving ITEM_LOC_SOH_MFQUEUE record.

If the error is a non-fatal error, HANDLE_ERRORS passes the sequence number of the driving ITEM_LOC_SOH_MFQUEUE record back to the RIB in the ROUTING_INFO. It sends back a status of 'H' - Hospital to the RIB as well. It then updates the status of the queue record to 'H', so that it will not get picked up again by the driving cursor in GETNXT.

If the error is a fatal error, a status of 'E' - Error is returned to the RIB. The error is considered non-fatal if no DML has occurred yet. Whenever DML has occurred, then the global variable LP_error_status is flipped from 'H' to 'E'.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
COInvAvailMod	Customer Order Store Inventory Available	COInvAvailDesc.xsd

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_LOC_SOH_MFQUEUE	Yes	Yes	Yes	Yes

Design Assumptions

Push off all DML statements as late as possible. Once DML statements have taken place, any error becomes a fatal error rather than a hospital error.

Banner and Channel Publication API

This section describes Banners and Channels API.

Functional Area

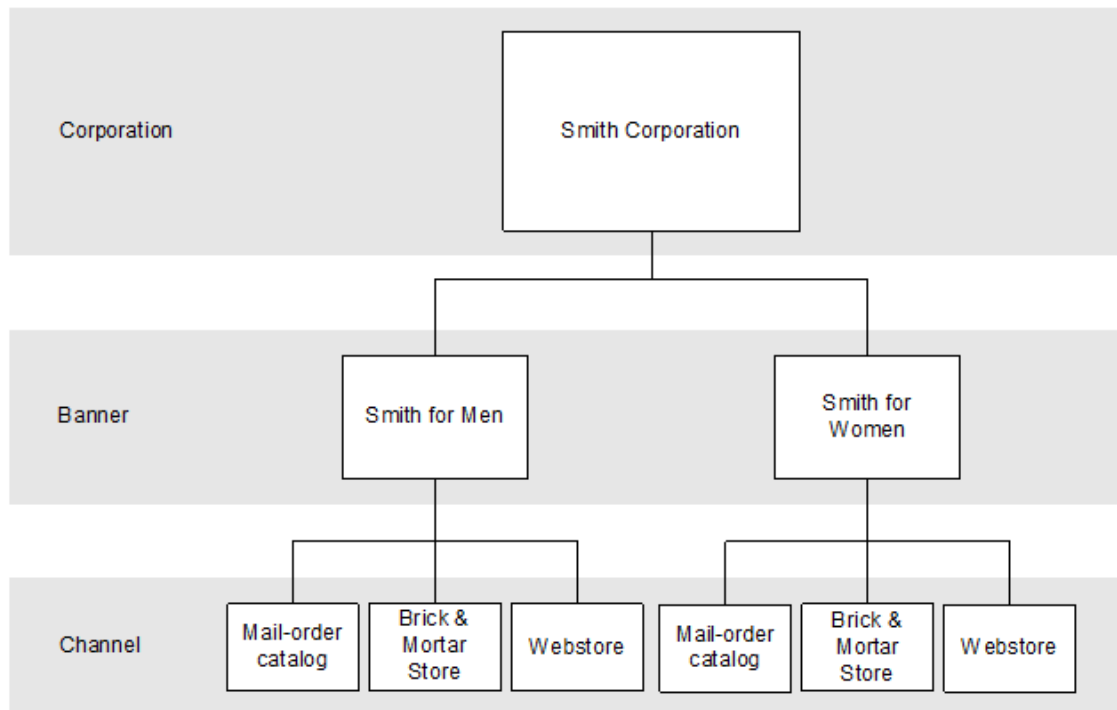
Foundation

Business Overview

Merchandising publishes details about new and updated banners and channels so that external systems that use this information can be informed of the updates. Updates are provided synchronously in a near-real-time manner. A banner usually represents the name on your “brand” name for your retail stores, and a channel is meant to represent the selling channels used by that banner, such as brick and mortar stores, web store, outlet, and so on.

The following diagram shows a sample of the structure of banners and channels within a retail organization.

Figure 2-1 Banners and Channels Within a Corporation



New Banners

Creating a new banner triggers a message to be sent through the RIB to notify external systems. The full details are sent for the new banner as part of the create message, the ID and description.

Updated Banners

When an existing banner is updated, an update message is triggered to provide the details of the update via the RIB. The update message, like create, will contain the full details in the message for all fields in the message.

Table 2-8 Banners – Create and Update

Message Element	Required?	Notes
Banner	Always	The unique number to identify a banner.
Banner Name	Always	The name of the banner in the primary language.

Deleted Banners

When an existing banner is deleted, this also triggers a delete transaction to be sent through the RIB to notify external systems that this banner is no longer valid. The delete message includes only the ID of the banner being deleted.

Table 2-9 Banners – Delete

Message Element	Required?	Notes
Banner	Always	The number of the banner being deleted.

New Channels

Creating a new channel triggers a message to be sent through the RIB to notify external systems. The full details are sent for the channel as part of the create message, including the ID and description for the channel, the banner in which the channel belongs, and a channel type.

Updated Channels

When an existing channel is updated, an update message is triggered to provide the details of the update through the RIB. The update message, like create, will contain the full details in the message for all fields in the message.

Table 2-10 Channels – Create and Update

Message Element	Required?	Notes
Channel	Always	The unique number used to identify the channel.
Channel Name	Always	The name of the channel in the primary language.
Channel Type	Always	Indicates the type of channel (e.g., Brick & Mortar, Ecommerce, and so on). The channel type sent is just the code; the descriptions for these codes are not sent but can be found on the Codes and Descriptions under code type CHTY.
Banner	Always	The number of the banner in which the channel exists.

Deleted Channels

When an existing channel is deleted, this also triggers a delete transaction to be sent through the RIB to notify external systems that this banner is no longer valid. The delete message will include only the ID of the channel being deleted.

Table 2-11 Channels – Create and Update

Message Element	Required?	Notes
Channel	Always	The number of the channel that is being deleted.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message in which the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish the messages is available. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
BannerCre	Banner Create	BannerDesc.xsd
ChannelCre	Channel Create	ChannelDesc.xsd
BannerMod	Banner Modify	BannerDesc.xsd
ChannelMod	Channel Modify	ChannelDesc.xsd
BannerDel	Transfer Delete	BannerRef.xsd
ChannelDel	Channel Delete	ChannelRef.xsd

Code and Diff Type Publication API

This section describes the seed data publication API.

Functional Area

Foundation Data

Integration Type

Oracle Retail Integration Bus (RIB)

Business Overview

Merchandising defines and publishes code types, codes, and their descriptions, along with differentiator (diff) type information for data seeding in external systems, as they are usually fairly static and do not frequently change after initial implementation. However, changes and deletes are also managed in this integration. Subscribing to this information in an external system allows it to interpret information included in other Merchandising integrations, such as

diffs associated with an item or statuses and other codes associated with purchase orders, transfers, and so on.

Code types and codes published are defined in Merchandising and are mostly used to display lists in the Merchandising UI like item types, shipping methods, supplier types, location types, and so on. Most code types and codes are part of the base implementation of Merchandising, but it is also possible to add or update codes and descriptions to suit your implementation. Code types and codes defined in Merchandising are also used in Sales Audit and Pricing. All code types and codes are published by this integration, regardless of whether or not they are flagged as Used in your implementation.

Diff Types are used to qualify the diff IDs included in other integration. Examples of diff types are size, color, flavor, and so on.

New Code Types

The creation of a new code type triggers the generation of a code header creation message. All code types defined in the merchandising is published to the external systems. The table below summarize the details included in this message.

Code Header

Table 2-12 Code Header Message Elements

Message Element	Included?	Notes
Code Type	Always	This field contains the code type which serves as a grouping mechanism for the codes.
Code Type Description	Always	This field contains the description of the code type.

Updated Code Types

Modifying the description of a code type will trigger the creation of a code header modification message. The update message will contain the details for all fields that changed to inform subscribing applications of the changes made in Merchandising. See the [New Code Types](#) section for details on the message.

Deleted Code Types

When a code type is removed, it will trigger a code header delete transaction message to an external system.

Code Header

Table 2-13 Code Header Message Elements

Message Element	Included?	Notes
Code Type	Always	This field contains the code type being deleted.

New Codes

The creation of a new code triggers the generation of a code detail creation message. All codes defined in the merchandising is published to the external systems. The table below summarize the details included in this message.

Code Detail

Table 2-14 New Codes Code Detail Message Elements

Message Element	Included?	Notes
Code Type	Always	This field contains a valid code type for the code.
Code	Always	This field contains the new code that has been added.
Code Description	Always	This field contains the description associated with the code.
Required Indicator	Always	This field indicates whether or not the code is required by Merchandising. Valid values are Yes (Y) or No (N).
Code Sequence	Always	This number is used to order the codes for display purposes.

Updated Codes

Modifying the description, required indicator, or code sequence of a code will trigger the creation of code detail modification message. The update message will contain the details for all fields that changed to inform subscribing applications of the changes made in Merchandising. See the [New Codes](#) section for details on the message.

Deleted Codes

When a code is removed, it will trigger a code detail delete transaction message to an external system.

Note

Setting a code's **Used** flag to No in Merchandising does not trigger a delete message to be published.

Code Detail

Table 2-15 Delete Codes Code Detail Message Elements

Message Element	Required?	Notes
Code Type	Always	This field contains the code type for the code being deleted.
Code	Always	This field contains the code being deleted.

New Diff Types

The creation of a new diff type triggers the generation of a diff type create message. The tables below summarize the details of the message.

Diff Type

Table 2-16 New Diff Types Diff Type Message Elements

Message Element	Required?	Notes
Diff Type	Always	Contains the value used to uniquely identify the diff type being added or updated.
Diff Type Description	Always	Contains the diff type description.

Child Nodes

- Custom Flex Attributes (optional)

Custom Flex Attributes (CFAS)

Table 2-17 Custom Flex Attributes

Message Element	Required?	Notes
Name	Always	Contains the attribute name configured for a diff type flex attribute.
Value	Optional	Contains the value of the attribute associated with the diff type for a character or number attribute.
Value Date	Optional	Contains the value of the attribute associated with the diff type, if the attribute is defined as a date type.

Updated Diff Types

Modifying diff type information will trigger the creation of a diff type modification message. The update message will contain the details for all fields that changed in the message in a format similar to that described above, to inform subscribing applications of the changes made in Merchandising.

Whenever a diff type is modified, all active CFAS defined for the diff type will be included in the message.

Note

Only active CFAS will be included

Deleted Diff Types

When a diff type is removed, it will trigger a diff type delete transaction message to an external system. CFAS attributes are not included in the delete message, but a delete of the diff type implies that the flex attributes should also be deleted in the subscribing solution.

Diff Type

Table 2-18 Diff Type Message Elements

Message Element	Required?	Notes
Diff Type	Always	Contains the value used to uniquely identify the diff type being deleted.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the particular message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish the messages is available. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Table 2-19 Message XSD

Message Types	Message Type Description	XML Schema Definition (XSD)
CodeHdrCre	Code Header Create	CodeHdrDesc.xsd
CodeHdrMod	Code Header Modify	CodeHdrDesc.xsd
CodeHdrDel	Code Header Delete	CodeHdrRef.xsd
CodeDtlCre	Code Detail Create	CodeDtlDesc.xsd

Table 2-19 (Cont.) Message XSD

Message Types	Message Type Description	XML Schema Definition (XSD)
CodeDtlMod	Code Detail Modify	CodeDtlDesc.xsd
CodeDtlDel	Code Detail Delete	CodeDtlRef.xsd
difftypecre	Diff Type Create	DiffTypeDesc.xsd
difftypemod	Diff Type Modify	DiffTypeDesc.xsd
difftypedel	Diff Type Delete	DiffTypeRef.xsd

Company Closed Publication API

This section describes the company closed publication API.

Functional Area

Foundation Data

Business Overview

Merchandising publishes details about new and updated company-wide closings for external systems that use this information to be informed of the updates, including company closed exceptions.

New Closings

Creating a company closing date triggers a message to be sent to notify external systems. The close date and description are sent as part of the create message.

Table 2-20 Company Closed - Create and Update

Message Element	Included?	Notes
Close Date	Always	This field contains the date that the company is closed.
Close Description	Optional	This field contains the description of the close, such as the reason for the company closure.

Updated Closings

Updating the description for a company closing date triggers a message to be sent to notify external systems. The close date and modified close description are sent as part of the message.

Deleted Closings

When a company closing date is deleted, this will also trigger a delete transaction to be sent to notify external systems that this company close date is no longer valid. The delete message will include only the close date.

Table 2-21 Company Closed - Delete

Message Element	Included?	Notes
Close Date	Always	This field contains the date that the company is closed.

New Closing Exceptions

Creating a company closing exception triggers a message to be sent to notify external systems. The close date, exception location, location type, and an indicator for whether or not the location is open for sales, receiving, and/or shipping are sent as part of the create message.

Table 2-22 Company Closed Exception - Create and Update

Message Element	Included?	Notes
Close Date	Always	This field contains the date on which an activity will not occur.
Location	Always	This field contains the unique location number.
Location Type	Always	This field contains the location type. Valid values are: • S - Store • W - Warehouse
Sales Indicator	Always	This field indicates whether or not the location is open for sales. A value of Yes (Y) indicates the location is open for sales. Valid values are Yes (Y) and No (N).
Receive Indicator	Always	This field indicates whether or not the location is open for receiving. A value of Yes (Y) indicates the location is open for receiving. Valid values are Yes (Y) and No (N).
Ship Indicator	Always	This field indicates whether or not the location is open for shipping. A value of Yes (Y) indicates the location is open for shipping. Valid values are Yes (Y) and No (N).

Updated Closing Exceptions

Updating a company closing exception triggers a message to be sent to notify external systems. The close date, exception location, location type, and an indicator for whether or not the location is open for sales, receiving, and/or shipping are sent as part of the message.

Deleted Closing Exceptions

When a company closing exception is deleted, this will also trigger a delete transaction to be sent to notify external systems that this company closed exception is no longer valid. The delete message will include only the close date and location.

Table 2-23 Company Closed Exception - Delete

Message Element	Included?	Notes
Location	Always	This field contains the unique location number.
Close Date	Always	This field contains the date on which an activity will not occur.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition
CoClosedCre	Company Closing Create	CompanyClosedDesc.xsd
CoClosedExcCre	Company Closing Exception Create	CompanyClosedExcepDesc.xsd
CoClosedMod	Company Closing Modify	CompanyClosedDesc.xsd
CoClosedExcMod	Company Closing Exception Modify	CompanyClosedExcepDesc.xsd
CoClosedDel	Company Closing Delete	CompanyClosedRef.xsd
CoClosedExcDel	Company Closing Exception Delete	CompanyClosedExcepRef.xsd

Country Publication API

This section describes the countries publication API.

Functional Area

Foundation Data

Business Overview

Seed object publication to the RIB allows Merchandising to send country information as well as currency rates so that external systems will have all of the latest information regarding countries and currency rates.

Seed object publication consists of a message containing country and currency rate information from the tables COUNTRY and CURRENCY_RATES. One message will be synchronously created and placed in the message queue each time a COUNTRY and CURRENCY_RATES record is created, modified or deleted in Merchandising. When a COUNTRY or CURRENCY_RATES record is created or modified, the message will contain a full snapshot of the modified record. When a COUNTRY record is deleted, the message will contain a partial snapshot of the deleted record. Messages are retrieved from the message queue in the order they were created.

New Country

The data seeding of country information and creation of a new country triggers the generation of a country creation message. The country creation message publishes the country ID, along with the country's attributes.

Country

Table 2-24 Country

Message Element	Included?	Notes
Country ID	Always	This contains the unique country identifier that has been added or updated.
Country Description	Always	This contains the name of the country.

Updated Country

Modifying a country information as part of various business processes will trigger the creation of country modification message. The update message will contain the details for all fields that changed in the message to inform subscribing applications of the changes made in Merchandising.

Deleted Country

When a country is removed, it will trigger a country delete transaction message to an external system.

Table 2-25 Deleted Country

Message Elements	Required?	Notes
Country ID	Always	This contains the unique country identifier that has been deleted.

Country

Message Elements	Included?	Notes
Country ID	Always	This contains the unique country identifier that has been deleted.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the particular message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish the messages

is available. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
countrycre	Country Create Message	CountryDesc.xsd
countrymod	Country Modify Message	CountryDesc.xsd
countrydel	Country Delete Message	CountryRef.xsd

Customer Order Fulfillment Confirmation Publication API

This section describes the customer order fulfillment confirmation publication API.

Functional Area

Customer Orders

Business Overview

When Merchandising is integrated with an order management system (OMS) or Oracle Retail Order Broker (OROB), one of the supported deployment methods is interfacing customer order fulfillment requests into Merchandising through the Oracle Retail Integration Bus (RIB). When Merchandising processes the customer order fulfillment request from OROB, it will also publish a confirmation message back indicating whether the order was created fully, partially, or not created. The confirmation message details are shown below.

Merchandising will only publish confirmation messages associated to customer orders that create a Purchase Order or Transfer, which is used for fulfillment from a supplier or a warehouse, or if the fulfillment requires movement between two locations, such as from warehouse to store for customer pickup.

Table 2-26 Fulfill Order Confirmation Message

Message Element	Included?	Notes
Customer Order Number	Always	This is the customer order number from OROB or OMS that is being confirmed.
Fulfillment Order Number	Always	This is the fulfillment order number from OROB or OMS.
Confirmation Type	Always	Contains the confirmation type. Valid values are: • C - Order Completely Created • P - Order Partially Created • X - Order Could not be Created
Confirmation Number	Optional	This field contains the PO or Transfer number in Merchandising related to the fulfillment order line. This is populated if fulfillment status is P or C.
Confirmation Details	Optional	Child Node

Table 2-26 (Cont.) Fulfill Order Confirmation Message

Message Element	Included?	Notes
Fulfillment Location ID	Optional	This field contains the fulfillment location for the order.
System Code	Always	Used by OROB to identify the system associated with this message. This is defaulted from the Merchandising system option.

Table 2-27 Fulfill Order Confirmation Detail Message

Message Element	Required?	Notes
Item	Always	This field indicates the item ordered by the customer.
Reference Item	Optional	This field indicates the reference item ordered by the customer. It is used only if a specific UPC is ordered.
Confirmed Quantity	Always	This field indicates the quantity of the item that can be sourced or fulfilled on the order.
Confirmed Quantity UOM	Always	This field indicates the unit of measure of the confirmation quantity.
Item Line Number	Optional	This field indicates the detail item line number on the order.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish the messages is available. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
fulfilordcfmcre	Customer Order Fulfillment Confirmation Create	FulfilOrdCfmDesc.xsd FulfilOrdCfmDtl.xsd

Delivery Slot Publication API

This section describes the delivery slot publication API.

Functional Area

Replenishment

Business Overview

This API publishes delivery slot details to external systems. This information is further subscribed by integrating systems like Oracle Retail Store Inventory Management (SIM). Within Merchandising, delivery slots are only used with the Store Order method of replenishment, allowing you the option to have multiple deliveries per day for the same store/day.

Create Delivery Slot

Delivery slot creation will result in publishing the ID, description, and a sequence to notify the integrating systems.

Update Delivery Slot

Modifying an existing delivery slot in Merchandising, will result in publishing a delivery slot modify message to notify the integrating systems. This includes the ID, description, and sequence value of delivery slot.

Table 2-28 Delivery Slot Create and Update

Message Element	Included?	Notes
Delivery Slot ID	Always	This field specifies a code indicating the delivery timeframe for stores that is being created or updated. For example, AM or PM.
Delivery Slot Description	Always	This field holds the delivery slot description. This could contain the time element for the delivery schedule (i.e., Afternoon Slot 3 PM).
Delivery Slot Sequence	Always	This column will specify the display sequence for the delivery slots.

Delete Delivery Slot

Deleting an existing delivery slot in Merchandising will result in publishing a delivery slot deletion message to notify the Integrating systems. Only the slot ID is included for deletes.

Table 2-29 Delivery Slot Delete

Message Element	Included?	Notes
Delivery Slot ID	Always	This field specifies a code indicating the delivery timeframe for stores that is being deleted.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
Dlvysltcre	Delivery Slot Create Message	DeliverySlotDesc.xsd
Dlvysltmod	Delivery Slot Modify Message	DeliverySlotDesc.xsd
Dlvysltdel	Delivery Slot Delete Message	DeliverySlotRef.xsd

Differentiator Group Publication API

This section describes the differentiator groups publication API.

Functional Area

Items

Business Overview

Merchandising publishes differentiator groups and its details through the RIB to external systems. Diff groups are published when a new diff group is created, updates are made to existing diff group, or an existing diff group is deleted.

New Diff Group Header

Creating a new diff group header triggers a message to be sent to notify external systems. The full details are also always sent for the new diff group.

New Diff Group Detail

Creating a new diff group detail record for an existing diff group also triggers a message to be sent to notify external systems. The details that are sent for the detail creation are the existing diff group ID, diff ID, and the display sequence, which indicates the order that diffs should be displayed within the group.

Updated Diff Group Header

When an existing diff group header is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields at the header level.

Updated Diff Group Detail

When an existing detail for a diff group is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields.

Table 2-30 Diff Group Header Create and Update

Message Element	Required?	Notes
Diff Group ID	Always	This field contains the ID of the diff group being added or updated.
Diff Group Type	Always	This field contains the code for the types of diffs contained in this group, such as S for size.
Diff Group Description	Always	This field contains the description of the diff group.
Diff Group Type Description	Optional	This field contains the description of the diff type, such as Size or Color.

Table 2-31 Diff Group Detail Create and Update

Message Element	Required?	Notes
Diff Group ID	Always	This field contains the ID of the diff group being added or updated.
Diff ID	Always	This field contains the ID of the diff that is included in the group.
Display Sequence	Optional	This field contains the display sequence for the diff in the group.

Deleted Diff Group

When an existing diff group is deleted, this will also trigger a delete transaction to be sent to notify external systems that this diff group is no longer valid. The delete message will include only the diff group ID. All the child records for the diff group should also be deleted when processing a diff group delete in an external system.

Deleted Diff Group Detail

When the existing details of diff group are deleted, this will also trigger a delete transaction to be sent to notify external systems that these details are no longer part of the diff group. The delete message will include the diff group ID and the diff ID. The diff ID would be the detail record from the diff group that has been deleted.

Table 2-32 Diff Group Header

Message Element	Required?	Notes
Diff Group ID	Always	This field contains the ID of the diff group being deleted.

Table 2-33 Diff Group Detail

Message Element	Required?	Notes
Diff Group ID	Always	This field contains the ID of the diff group whose detail is being deleted.
Diff ID	Always	This field contains diff being deleted from the group.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
DiffGrpHdrCre	Diff Group Create	DiffGrpHdrDesc.xsd
DiffGrpDtlCre	Diff Group Detail Create	DiffGrpDtlDesc.xsd
DiffGrpHdrMod	Diff Group Modify	DiffGrpHdrDesc.xsd
DiffGrpDtlMod	Diff Group Detail Modify	DiffGrpDtlDesc.xsd
DiffGrpDel	Diff Group Delete	DiffGrpRef.xsd
DiffGrpDtlDel	Diff Group Detail Delete	DiffGrpDtlRef.xsd

Differentiator Publication API

This section describes the differentiators publication API.

Functional Area

Items

Business Overview

Merchandising publishes details about new and updated differentiator identifiers (diff IDs) in order that external systems that use this information can be informed of the updates. Updates are provided synchronously in a near-real time manner. When the external system receives information about an item that includes the new differentiator ID, that system understands what the differentiator ID refers to.

New Differentiator IDs

When a new differentiator ID is created in Merchandising, it triggers a message to be sent to notify external systems. The full details are sent for the new differentiator ID as part of the create message, the differentiator ID, differentiator type, differentiator description, industry code, industry sub code, and differentiator type description.

Updated Differentiator IDs

When an existing differentiator ID is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields in the message.

Table 2-34 Differentiator Create and Update

Message Element	Required?	Notes
Differentiator ID	Always	This field contains a unique identifier of the differentiator.
Differentiator Type	Always	This field contains a code indicating the type of differentiator, for example C for Color.
Differentiator Description	Always	This field contains the description of the differentiator.
Industry Code	Optional	This field is used to hold the unique code used by industry standards to identify the differentiator.
Industry Sub Code	Optional	This field is used to hold a sub-grouping code used by industry standards to further identify the differentiator.
Differentiator Type Description	Optional	This field holds the description of the diff type (e.g., Color, Size, and so on).

Deleted Differentiator IDs

When an existing differentiator ID is deleted, this will also trigger a delete transaction to be sent to notify external systems that this differentiator ID is no longer valid. The delete message will include only the ID of the differentiator being deleted.

Table 2-35 Differentiator Delete

Message Element	Required?	Notes
Differentiator ID	Always	This field contains a unique identifier of the differentiator.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
diffcre	Diff Create	DiffDesc.xsd
diffmod	Diff Modify	DiffDesc.xsd
diffdel	Diff Delete	DiffRef.xsd

Item Location Publication API

This section describes the item location publication API.

Functional Area

Items

Business Overview

Merchandising defines and publishes item/location relationships. The details about item/location relation creation, modification, de-activation, and deletion are important for other systems for the smooth functioning of several business processes. Attributes included in this integration are pulled from both the Item Location and Item Location Traits tables in Merchandising.

To support multi-channel environments in Merchandising, a physical warehouse is divided into one or more virtual warehouses to better track and manage goods by channel. Stores and virtual warehouses are the locations that hold inventory and are associated to an item. Therefore, when creating, modifying, or deleting an item/location relationship where the location is a virtual warehouse, the physical warehouse information is also included in the message published. This is to cater to external systems that require the physical warehouse rather than the virtual warehouse.

In general, pricing updates are sent from Pricing to inform dependent solutions of a pending price change or clearance markdown. But, for solutions that do not subscribe to pricing events from Pricing, when the selling price for an item is changed, an item/location update will be published to inform other solutions of the updated price, including an update to the clearance flag.

New Item Location Relationship

The creation of a new item/location relationship triggers the generation of an item/location creation message. Please note though, that item/location records will not be published before the corresponding item record is published. Merchandising ensures that when an item creation message has not been published yet, the corresponding item/location creation message is not picked up for publication. The item/location relationship creation message publishes the item identifier, along with the location-specific attributes.

The message includes the following:

Item/Location Create and Update

Message Element	Included?	Notes
Item	Always	This is the item ID the item/location change is related to.
Item Level	Always	Indicates the level of the item within an item family. Valid values are 1, 2, or 3.
Tran Level	Always	This is the level at which inventory is tracked for this item's family. Valid values are 1, 2, or 3. Items having Item Level > Tran Level (indicating it is a below-transaction-level item) will not have its changes published.

Child Nodes

- Item/Location Physical Location Details

Item Physical Location Create and Update

Message Element	Included?	Notes
Physical Location	Always	This is the Physical WH associated to the virtual warehouse location. If the location is a Store, this will contain the Store ID.
Location Type	Always	This is the type of location in the location field. Valid values are Store (S) or Warehouse (W).
Store Type	Optional	This is populated for store location types only. This will indicate whether a particular store is a franchise (F) or company store (C).
Stockholding Indicator	Optional	This is populated for store location types only. If the store is a non-stockholding store, message will not be subscribed to SIM/SIOCS.
Returnable Indicator	Always	This contains a value of Yes (Y) if the item can be returned to the location and No (N) if not.

Child Nodes

- Item/Location Virtual Location Details (required for item/locations update)
- Item/Location Virtual Location Replenishment Details (required for Replenishment item/location update)

Item Virtual Location Create and Update

Message Element	Included?	Notes
Location	Always	Contains the location ID.
Location Type	Always	This is the type of location in the location field. Valid values are S (store) and W (warehouse).
Local Item Description	Optional	This contains the local description of the item.
Local Item Short Description	Optional	This contains the local short description of the item.
Status	Optional	This contains the status of the item at the location. Valid values are: • Active (A) - Item is valid and can be ordered and sold • Inactive (I) - Item is valid but cannot be ordered or sold • Discontinued (C) - Item is valid and sellable but no longer orderable • Delete (D) - Item is invalid and cannot be ordered or sold
Primary Supplier	Optional	This contains the primary supplier site for the item/location.
Primary Country	Optional	This contains the primary country of sourcing for the item/location.
Receive As Type	Optional	This determines whether the stock on hand for a pack component item or the buyer pack itself will be updated when a buyer pack is received at a warehouse. Valid values are Each (E) or Pack (P).

Message Element	Included?	Notes
Taxable Indicator	Optional	This field determines if the item is taxable at the store. Valid values are Yes (Y) or No (N).
Source Method	Optional	This field determines the primary sourcing for this item/location. Valid values are Supplier (S) or Warehouse (W).
Source WH	Optional	This determines which warehouse is the sourcing location for this item/location. This value is required if the sourcing method is Warehouse.
Unit Retail	Optional	This defines the unit retail price in the standard unit of measure for the item/location combination. This field will be stored in the local currency.
Selling Unit Retail	Optional	This defines the unit retail price in the selling unit of measure for the item/location combination. This field will be stored in the local currency.
Selling UOM	Optional	This defines the selling unit of measure for an item's single-unit retail at the location.
Store Price Indicator	Optional	This determines if an item at a particular store location can have the unit retail marked down by the store.
Purchase Type	Optional	This defines whether the item is owned, consignment stock, or a concession item at the location. Valid values are: • 0 - Owned • 1 - Consignment • 2 - Concession
UIN Type	Optional	This contains the type of unique identification number (UIN) used to identify the instances of the item at the location. Valid values are found in code type UINT.
UIN Label	Optional	This contains the label for the unique identification number (UIN) when displayed. Valid values are found in code type ULBL.
Capture Time	Optional	This determines when the unique identification number (UIN) should be captured for an item during transaction processing. Valid values are found in code type CPTM.
External UIN Indicator	Always	This indicates if a unique identification number (UIN) is being generated in the external system. Valid values are Yes (Y) or No (N).
Ranged Indicator	Optional	This determines if the location is ranged intentionally by a user for replenishment/selling or incidentally ranged by other processes. Valid values are Yes (Y) or No (N).
Item Parent	Optional	This uniquely identifies the item/group at the level above the item.
Item Grandparent	Optional	This uniquely identifies the item/group two levels above the item.
TI	Optional	This determines the number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of cases for a pallet.

Message Element	Included?	Notes
HI	Optional	This determines the number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of cases for a pallet.
Store Order Multiple	Optional	This contains the multiple in which the item needs to be shipped from a warehouse to the location. Valid values are found in ORML.
Daily Wastage Percentage	Optional	This defines the average percentage lost from inventory on a daily basis due to natural wastage.
Ticket Price Units	Optional	This defines the units in the ticket price in terms of the Price UOM for ticketing.
Ticket Price	Optional	This defines the retail to be used on the ticket in terms of the Price UOM for ticketing.
Ticket UOM	Optional	Unit of measure used on the ticket for this item.
Primary Variant	Optional	This field is used to address sales of PLUs (that is, above transaction level items) when inventory is tracked at a lower level (that is, UPC). This field will only contain a value for items one level higher than the transaction level. Valid choices will be any transaction level item that is a child of this item.
Primary Cost Pack	Optional	This contains a simple pack item containing the item in the item column for this record.
Inbound Handling Days	Optional	This indicates the number of inbound handling days for an item at a warehouse type location.
Regular Unit Retail	Optional	This defines the unit retail in the standard unit of measure for the item/location when not on clearance. This field is stored in the local currency.
Multi-units	Optional	This defines the quantity to be purchased in order to get the multi-unit retail in the multi-selling unit of measure for the item/location.
Multi-unit Retail	Optional	This defines the multi-unit retail in the multi-selling unit of measure for the item/location. This field is stored in the local currency.
Multi-selling UOM	Optional	This defines the selling unit of measure for the item/location if a multi-unit retail has been defined.
Clearing Indicator	Optional	This determines if item is on clearance at the store. Valid values are Yes (Y) or No (N).
Costing Location	Optional	This contains the costing location for the franchise store. This field may contain a store or a warehouse.
Costing Location Type	Optional	This defines the type of location is in the costing location field. Valid values are Store (S) or Warehouse (W).
Currency Code	Always	This defines the currency code under which the location operates.
Launch Date	Optional	This holds the date when the item is initially sold at the location.
Quantity Key Options	Optional	This field determines whether the quantity key on a POS can be used for this item at the location. Valid values are found in code type RPO.

Message Element	Included?	Notes
Manual Price Entry	Optional	This field determines whether the price for the item/location can be entered manually on a POS. Valid values are found in code type RPO.
Deposit Code	Optional	This determines whether a deposit is associated with this item at the location. Valid values are found in code DEPO.
Food Stamp Indicator	Optional	This determines whether the item is approved for food stamps at the location. Valid values are Yes (Y) or No (N).
WIC Indicator	Optional	This determines whether the item is approved for WIC at the location. Valid values are Yes (Y) or No (N).
Proportional Tare Percentage	Optional	This is the proportion of the total weight of a unit of an item that is the packaging. For example, if the tare item is bulk candy, this is the proportional of the total weight of one piece of candy that is the candy wrapper.
Fixed Tare Value	Optional	This is the tare of the packaging. For example, if the tare item is bulk candy, this is weight of the bag and twist tie.
Fixed Tare UOM	Optional	This contains the unit of measure value associated with the tare value.
Reward Eligible Indicator	Optional	This determines whether the item is legally valid for various types of bonus point/award programs at the location. Valid values are Yes (Y) or No (N).
National Brand Comparison Item	Optional	This contains the nationally branded item to which it will be compared to. Will contain a valid item ID.
Return Policy	Optional	This determines the return policy for the item at the location. Valid values are found in code type RETP.
Stop Sale Indicator	Optional	This defines if the sale of the item should be stopped immediately at the location (that is, in case of recall, or something similar). Valid values are Yes (Y) or No (N).
Elect Marketing Clubs	Optional	This contains the code that represents the marketing clubs to which the item belongs at the location. Valid values are found in code type MKTC.
Report Code	Optional	This determines to which reports the location should run. Valid values are found in code type REPC.
Required Shelf Life on Selection	Optional	This contains the required shelf life for an item on selection in days.
Required Shelf Life on Receipt	Optional	This contains the required shelf life for an item on receipt in days.
Store Re-orderable Indicator	Optional	This determines whether the store may re-order the item. Valid values are Yes (Y) or No (N).
Rack Size	Optional	This determines the rack size that should be used for the item.
Full Pallet Item	Optional	This determines whether a store must reorder an item in full pallets only. Valid values are Yes (Y) or No (N).

Message Element	Included?	Notes
In-Store Market Basket	Optional	This contains the in-store market basket code for the item/location combination. Valid values are found in code type STMB.
Storage Location	Optional	This contains the current storage location or bin number for the item at the location.
Alternate Storage location	Optional	This contains the preferred alternate storage location or bin number for the item at the location.
Refundable Indicator	Optional	This determines if the item is refundable at the location or not. Valid values are Yes (Y) or No (N).
Back Order Indicator	Optional	This determines if the item can be back-ordered to the location. Valid values are Yes (Y) or No (N).
Promotable Indicator	Optional	This determines whether the retailer is allowed to specify if the item is promotable or not. Valid values are Yes (Y) or No (N).
Unit Cost	Optional	This contains the current unit cost of the item based on the primary supplier/country for the location in local currency.
Pickup Lead Time	Optional	This defines the time it takes to get the item from the supplier to the location.
Cost UOM	Optional	This is used to allow costs to be managed in a different UOM than the standard UOM.
Calculation Basis	Optional	This determines if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are: • C - Cost per Unit • P - Purchase Rate
Purchase Rate	Optional	This contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item, if the calculation basis for the item/location is Purchase Rate.
RFID Indicator	Optional	This allows the retailer to specify if the item should be RFID tagged or not.
Pricing Control	Optional	This field contains the pricing control of the item which will determine the item price is controlled by Retailer or Supplier when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are R - Retailer, S – Supplier For owned items, the value is always NULL.

Child Nodes

- Custom Flex Attributes (optional)

Modified Item Location Relationship

Modifying an item/location attribute as part of various business processes will trigger the creation of an item/location modification message. The update message will contain the details

for all fields that changed in the message to inform subscribing applications of the changes made in Merchandising.

Deleted Item Location Relationship

When an item/location relationship is removed from an item (not just moved to Deleted status), it will trigger an item/location delete transaction message to an external system. The delete message contains the item/location to be deleted.

Item/Location Delete

Table 2-36 Item/Location Delete

Message Element	Included?	Notes
Item	Always	This is the item ID where a location relationship is being deleted.
Item/Physical Location		Child Node

Item/Physical Location Delete

Message Element	Included?	Notes
Physical Location	Always	This is the Physical WH associated to a virtual warehouse location being deleted. If the location is a Store, this will contain the Store ID.
Location Type	Always	This is the type of location in the location field. Valid values are Store (S) or Warehouse (W).
Store Type	Optional	This is populated for store location types only. This will indicate whether a particular store is a franchise (F) or company store (C).
Stockholding Indicator	Optional	This is populated for store location types only. If the store is a non-stockholding store, message will not be subscribed to SIM/SIOCS.
Item/Virtual Location		Child Node

Item/Virtual Location Delete

Message Element	Included?	Notes
Location	Always	Contains the location ID being deleted for the item.

Item Replenishment Attributes

The creation, modification or deletion of item replenishment attributes for a location will trigger the creation of an item/location replenishment modification message. This will publish a message to subscribing systems to inform them of the replenishment attributes updates for the item on a particular location.

Item/Location Virtual Location Replenishment Details

Message Element	Included?	Notes
Location	Always	This is the location ID where the item/location change is related to. This will contain the Virtual Warehouse if the location type is Warehouse or the Store ID if the location type is a Store.
Location Type	Always	This is the type of location in the location field. Valid values are Store (S) or Warehouse (W).
Primary Replenishment Supplier	Always	The supplier from which the specified location will source the replenishment demand for the specified item location.
Replenishment Method	Always	The code for the algorithm that will be used to calculate the recommended order quantity for the item location. Valid values are: • C - Constant • M - Min/Max • F - Floating point • T - Time Supply • D - Dynamic • SO - Store Orders
Reject Store Order Indicator	Optional	Contains an indicator that determines whether uploaded store orders can be rejected. If the indicator is No (N), then store orders for all need dates are valid. If Yes (Y), store orders with need dates on or after the next delivery date are valid.
Next Delivery Date	Optional	The next delivery date calculated for the next review cycle.
Multiple Runs Per Day Indicator	Always	Indicates if an item can be replenished multiple times per day at the location.

Custom Flex Attributes

If any custom flex attributes (CFAS) for the item/location has been added or modified, it will trigger an item/location modify message. All of the entity's active flex attributes from all attribute groups are published as key-value pairs based on the group set view. This CFAS object is embedded in the outbound Item/Location message.

Custom Flex Attributes

Message Element	Included?	Notes
Name	Always	The flex attribute defined by the business
Value	Optional	The value of the flex attribute defined by the business
Value Date	Optional	The date value of the flex attribute if flex attribute is defined as a date.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the particular message where the error was encountered, a status of Hospital (H) is sent to the

RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish the messages is available. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
ItemLocCre	Item Location Create Message	ItemLocDesc.xsd
ItemLocMod	Item Location Modify Message	ItemLocDesc.xsd
ItemLocDel	Item Location Delete Message	ItemLocRef.xsd
ItemLocReplMod	Item Location Replenishment Modify Message	ItemLocDesc.xsd

Item Publication API

This section describes the item publication API.

Functional Area

Items

Business Overview

Merchandising publishes messages about items to the Oracle Retail Integration Bus (RIB). In situations where a retailer creates a new item in Merchandising, the message that ultimately is published to the RIB contains a hierarchical structure of the item itself along with all components that are associated with that item. Items and item components make up what is called the Items message family.

After the item creation message has been published to the RIB for use by external applications, any modifications to the basic item or its components cause the publication of individual messages specific to that component. Deletion of an item and component records has similar effects on the message modification process, with the exception that the delete message holds only the key(s) for the record.

When publishing an item header modification, reference item modification, and reference item deletion, a second full replacement message from Merchandising will be published when the publish full objects indicator is Yes. This message payload will contain an item header (with custom flex attributes), item BOM, and UPC information for the item. Based on the message type, RIB will route the full replacement message to appropriate applications.

Deposit items

A deposit item is a product that has a portion which is returnable to the supplier and sold to the customer, with a deposit taken for the returnable portion. Because the contents portion of the item and the container portion of the item must be managed in separate financial accounts (as

the container item would be posted to a liabilities account) with different attributes, the retailer must set up two separate items. All returns of used deposit items (the returned item) are managed as a separate product, to track these products separately and as a generic item not linked to the actual deposit item (for example, bottles being washed and having no label).

The retailer can never put a container item on a transfer. The container item is added to returns to vendors (RTVs) automatically when the retailer adds the associated content item.

Deposit item attributes in Merchandising enable contents, container and crate items to be distinguished from one another. Additionally, it is possible to link a contents item to a container item for the purposes of inventory management.

In addition to contents and container items, many deposit items are delivered in plastic crates, which are also given to the customer on a deposit basis. These crates are sold to a customer as an additional separate product. Individual crates are not linked with contents or container items. Crates are specified in the system with a deposit item attribute.

From a receiving perspective, only the content item can be received. The receipt of a PO shows the container item, but the receipt of a transfer does not. Similar to RTV functionality, online purchase order functionality automatically adds the container. The system automatically replicates all transactions for the container item in the stock ledger. In sum, for POs and RTVs, the container item is included; for transfers, no replication occurs.

Catch-Weight Items

Retailers can order and manage products for the following types of catch-weight item:

- Type 1: Purchase in fixed weight simple packs: sell by variable weight (for example, bananas).
- Type 2: Purchase in variable weight simple packs: sell by variable weight (for example, ham on the bone sold on a delicatessen counter).
- Type 3: Purchase in fixed weight simple packs containing a fixed number of eaches: sell by variable weight eaches (for example, pre-packaged cheese).
- Type 4: Purchase in variable weight simple packs containing a fixed number of eaches: sell by variable weight eaches (for example, pre-packaged sirloin steak).

Note

Oracle Retail suggests that catch-weight item cases be managed through the standard simple pack functionality.

For catch-weight items to be managed in Merchandising, the following item attributes are available:

- Cost UOM: All items in Merchandising will be able to have the cost of the item managed in a separate unit of measure (UOM) from the standard UOM. Where this is in a different UOM class from the standard UOM, case dimensions must be set up.
- Catch-weight item pack details: Tolerance values and average case weights are stored for catch-weight item cases to allow the retailer to report on the sizes of cases received from suppliers.
- Maximum catch-weight tolerance threshold.
- Minimum catch-weight tolerance threshold.

Retailers can set up the following properties for a catch-weight item:

- Order type
- Sale type

Retailers can also specify the following, at the item-supplier-country level:

- Cost unit of measure (CUOM).

Receiving and inventory movement impact on catch-weight items

Inventory transaction messages include purchase order receiving, stock order receiving, returns to vendor, direct store delivery receiving, inventory adjustments and bill of lading. These messages include attributes that represent, for catch-weight items, the actual weight of goods involved in a transaction. These attributes are weight and weight UOM.

When Merchandising subscribes to inventory transaction messages containing such weight data, the transaction weight will be used for two purposes:

- To update weighted average cost (WAC) using the weight rather than the number of units and to update the average weight value of simple packs.

Note

The WAC calculation does not apply to return to vendors (RTVs).

Item Transformation

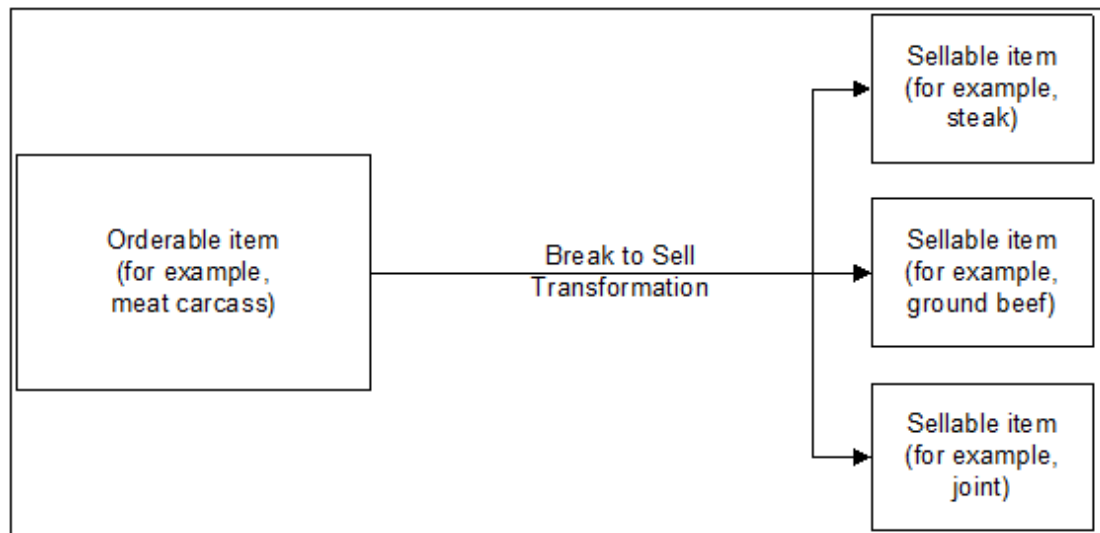
Item transformation allows retailers to manage items where the actual transformation of a product cannot be adequately recorded due to in-store processes.

With product transformation, new 'transform' items are set up as either sellable only or orderable only.

- **Sellable only items:** A sellable only item has no inventory in the system, so inventory records cannot be viewed from the item maintenance screens. Sellable only items do not hold any supplier links and therefore have no cost prices associated with them.
- **Orderable only items:** Orderable only items hold inventory but are not sellable at the POS system. Therefore, no information is sent to the POS system for these items, and no unit retail prices by zone are held for these items.

To hold the relationship between the orderable items and the sellable items, Merchandising stores the transformation details. These details are used to process sales and inventory transactions for the items.

The following diagram shows how item transformation works:

Figure 2-2 Item Transformation

Item and Item Component Descriptions

The item message family is a logical grouping for all item data published to the RIB. The components of item messages and their base tables in Merchandising are:

- Item from the ITEM_MASTER table
- Item-supplier from ITEM_SUPPLIER
- Item-supplier-country from ITEM_SUPP_COUNTRY
- Item-supplier-country-dimension from ITEM_SUPP_COUNTRY_DIM (DIM is the each, inner, pallet, and case dimension for the item, as specified)
- Item-image from ITEM_IMAGE
- Item-UDA identifier-UDA value from UDA_ITEM_LOV (UDA is a user-defined attribute and LOV is list of values)
- Item-UDA identifier from UDA_ITEM_DATE (for the item and UDA date)
- Item-UDA identifier from UDA_ITEM_FF (for UDA, free-format data beyond the values for LOV and date)
- Item-pack components (Bill of Material [BOM]) from PACKITEM_BREAKOUT
- Item UPC reference from ITEM_MASTER.ITEM_NUMBER_TYPE (values held as code type 'UPCT' on code_head and code_detail tables)
- Item ticket from ITEM_TICKET
- Item relationship details from RELATED_ITEM_HEAD
- Related Items details from RELATED_ITEM_DETAIL
- VAT item details from VAT_ITEM (only when default tax type is not GTS)

New Item Message Processes

The creation of a new item in Merchandising begins with an item in a worksheet status on the ITEM_MASTER table. At the time an item is created, other relationships are being defined as

well, including the item, supplier, and country relationships, user-defined attributes (UDAs), related items and others. These item relationship processes in effect become components of a new item message published to the RIB. This section describes the item creation message process and includes the basic item message itself along with the other component relationship messages that become part of the larger item message.

Basic Item Message

As described in the preceding section, item messages can originate in a number of Merchandising tables. Creating, modifying or deleting an item triggers a message to be sent through the RIB to notify external systems. Additions, modifications and deletions to item family records for existing approved items are published in the order that they are placed in the queue.

New Item Message Publication

The publication of a new item and its components to the RIB is done using a hierarchical message. All the components that comprise the creation of an item, the item/supplier for example, remain in the queue until the item approval modification message has been published. Any modifications or deletions that occur between item creation in “W” (worksheet) status and “A” (Approved) status are applied to the “create” messages or deleted from the queue as required.

For example, if an item UDA is added before item approval and then later deleted before item approval, the item UDA “create” message would be deleted from the queue before publishing the item. If an item/supplier record is updated for a new item before the item is approved, the “create” message for that item/supplier is updated with the new data before the item is published. When the “modify” message that contains the “A” (Approved) status is the next record on the queue, the procedure formats a hierarchical message that contains the item header information and all the child detail records to pass to RIB.

Subordinate Data and XML Tags

While a new item is being created, item components are also being created. Described earlier in this overview, these component item messages pertain to the item-supplier, item-supplier-country, UDAs, and so on.

For example, when a new item-supplier is created, this triggers a message to be created. When the item is approved, this item-supplier message will be added to the item creation message.

Table 2-37 Item Header

Message Element	Included?	Notes
Item	Always	Unique alphanumeric value that identifies the item.
Item Number Type	Always	Indicates the type of the number, for example UPC-12. The values for this column are maintained in code type UPCT.
Format Id	Optional	This field will hold the format ID that corresponds to the item's variable UPC. This value is only used for items with variable UPCs.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Prefix	Optional	This column holds the prefix for variable weight UPCs. The prefix determines the format of the eventual UPC and will be used to decode variable weight UPCs that are uploaded from the POS. It is the clients responsibility to download this value to their POS system.
Item Parent	Optional	Alphanumeric value that uniquely identifies the item/group at the level above the item. This value must exist as an item in another row on the item_master table.
Item Grandparent	Optional	Alphanumeric value that uniquely identifies the item/group two levels above the item. This value must exist as both an item and an item parent in another row on the item_master table.
Pack Indicator	Always	Indicates if the item is a pack. A pack item is a collection of items that may be either ordered or sold as a unit. Packs require details (that is, component items and quantities, and so on) that other items do not.
Item Level	Always	Number indicating which of the three levels the item resides. The item level determines if the item stands alone or if it is part of a family of related items. The item level also determines how the item may be used throughout the system.
Transaction Level	Always	Number indicating which of the three levels transactions occur for the item's group. The transaction level is the level at which the item's inventory is tracked in the system. The transaction level item will be counted, transferred, shipped, etc. The transaction level may be at the current item or up to 2 levels above or below the current item. Only one level of the hierarchy of an item family may contain transaction level items.
Diff 1	Optional	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
Diff 1 Type	Optional	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
Diff 2	Optional	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
Diff 2 Type	Optional	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Diff 3	Optional	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
Diff 3 Type	Optional	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
Diff 4	Optional	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
Diff 4 Type	Optional	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
Department	Always	Number identifying the department to which the item is attached. The item's department will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the deps table.
Department Name	Always	Contains the name of the department.
Class	Always	Number identifying the class to which the item is attached. The item's class will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the class table.
Class Name	Always	Contains the name of the class.
Subclass	Always	Number identifying the subclass to which the item is attached. The item's subclass will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the subclass table.
Subclass Name	Always	Contains the name of the subclass.
Status	Always	Status of the item. Valid values are: W = Worksheet: item setup in progress, cannot be used in system; S = Submitted: item setup complete and awaiting approval, cannot be use in system; A = Approved: item is approved and can now be used throughout the system. An item is not published until it is Approved.
Item Description	Always	Long description of the item. This description is used throughout the system to help online users identify the item. For items that have parents, this description will default to the parent's description plus any differentiators. For items without parents this description will default to null.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Short Description	Optional	Shortened description of the item. This description is the default for downloading to the POS. For items that have parents, this description will default to the parent's short description. For items without parents, this description will default to null.
Description Up	Optional	All uppercase description of the item (same as upper(item_desc)). This field is not displayed in the on-line item maintenance screen but is used in item list of value search processing throughout the system.
Primary Reference Item Indicator	Optional	Indicates if the sub-transaction level item is designated as the primary sub-transaction level item. For transaction level items and above the value in this field will be 'N'.
Cost Zone Group Id	Optional	Cost zone group associated with the item. This field is only required when elc_ind (landed cost indicator) is set to 'Y' on the system_options table.
Standard Unit of Measure	Always	Unit of measure in which stock of the item is tracked at a corporate level.
Unit of Measure Conv Factor	Optional	Conversion factor between an "Each" and the standard_uom when the standard_uom is not in the quantity class (e.g. if standard_uom = lb and 1 lb = 10 eaches, this factor will be 10). This factor will be used to convert sales and stock data when an item is retailed in eaches but does not have eaches as its standard unit of measure.
Package Size	Optional	Holds the size of the product printed on any packaging (for example, 24 ounces).
Package Unit of Measure	Optional	Holds the unit of measure associated with the package size.
Merchandise Indicator	Always	Indicates if the item is a merchandise item (Y, N).
Store Order Multiple	Always	Merchandise shipped from the warehouses to the stores must be specified in this unit type. Valid values are: C = Cases; I = Inner; E = Eaches.
Forecast Indicator	Always	Indicates if this item will be interfaced to an external forecasting system (Y, N).
Mfg Rec Retail	Optional	Manufacturer's recommended retail price for the item. Used for informational purposes only. This field is stored in the primary currency.
Retail Label Type	Optional	This field indicates any special label type associated with an item (i.e. pre-priced or cents off). This field is used for reporting purposes only. Values for this field are defined by the 'RTLT' code on code detail.
Retail Label Type Description	Optional	Contains the retail label type description corresponding to the retail_label_type value in item_master.
Retail Label Value	Optional	This field represents the value associated with the retail label type.
Handling Temp	Optional	Holds the temperature information associated with the item. Valid values for this field are in the code type 'HTMP' on the code_head and code_detail tables.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Handling Temp Description	Optional	Contains the Handling temperature description corresponding to the handling_temp value in item_master.
Handling Sensitivity Description	Optional	Contains the Handling sensitivity description corresponding to the handling_sensitivity value in item_master.
Catch Weight Indicator	Always	Indicates whether the item should be weighed when it arrives at a location. Valid values for this field are 'Y' and 'N'.
Waste Type	Optional	Identifies the wastage type as either sales or spoilage wastage. Sales wastage occurs during processes that make an item saleable (i.e. fat is trimmed off at customer request). Spoilage wastage occurs during the products shelf life (for example, evaporation causes the product to weigh less after a period of time). Valid values are: SP - spoilage; and SL - Sales.
Waste Type Description	Optional	Contains the Waste Type description corresponding to the Waste_Type value in item_master.
Waste Percent	Optional	Average percent of wastage for the item over its shelf life. Used in inflating the retail price for wastage items.
Default Waste Percent	Optional	Default daily wastage percent for spoilage type wastage items. This value will default to all item locations and represents the average amount of wastage that occurs on a daily basis.
Const Dimension Indicator	Optional	Indicates that the dimensions of the product are always the same, regardless of the supplier. If this field is Y, the dimensions for all suppliers will be defaulted to the primary supplier/primary country dimensions. Editing of dimensions for the item will only be allowed for the primary supplier/primary country.
Simple Pack Indicator	Always	Indicates if pack item is a simple pack or not. This field will only be available if the item is a pack item. A simple pack is an item whose components are all the same item (for example, a six pack of cola, and so on). Valid values are Y = Yes, this item is a simple pack; N = No if this is not a simple pack.
Contains Inner Indicator	Always	Indicates if pack item contains inner packs. Vendor packs will never contain inner packs and this field will be defaulted to 'N'. This field will only be available if the item is a pack item. Valid values are: Y = Yes, this pack contains inner packs; N = No, this pack does not contain inner packs.
Sellable Indicator	Always	Indicates if pack item may be sold as a unit. If it is 'Y' then the pack will have its own unique unit retail. If it is 'N' then the pack's unit retail is the sum of each individual item's total retail within the pack. This field will only be available if the item is a pack item. Value values are: Y = Yes, this pack may be sold as a unit; N = No, this pack may not be sold as a unit.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Orderable Indicator	Always	Indicates if pack item is orderable. If it is 'Y' then the suppliers of the pack must supply all components in the pack. If it is 'N' then the components may have different suppliers. This field will only be available if the item is a pack item. Valid values are: Y = Yes, this pack may be ordered, N = No, this pack may not be ordered.
Pack Type	Optional	Indicates if pack item is a vendor pack or a buyer pack. A vendor pack is a pack that the vendor or supplier recognizes and sells to the retailer. If the pack item is a vendor pack, communication with the supplier will use the vendor pack number. A buyer pack is a pack that a buyer created for internal ease of use. If the pack item is a buyer pack communication with the supplier will explode the pack out to its component items. This field will only be used if the item is a pack item. If the pack item is not orderable this field must be null. Valid values are: V = Vendor; B = Buyer.
Order As Type	Optional	Indicates if pack item is receivable at the component level or at the pack level (for a buyer pack only). This field is required if pack item is an orderable buyer pack. This field must be NULL if the pack is sellable only or a vendor pack. This field will only be available if the item is a pack item. Valid Values are: E = Eaches (component level) P = Pack (buyer pack only).
Comments	Optional	Holds any comments associated with the item.
Unit Retail	Optional	This field holds the unit retail in the standard unit of measure for the item/zone combination. This field is stored in the local currency.
Item Service Level	Optional	Holds a value that restricts the type of shipment methods that RCOM can select for an item.
Gift Wrap Indicator	Always	This field will contain a value of 'Y' if the item is eligible to be gift wrapped.
Ship Alone Indicator	Always	This field will contain a value of 'Y' if the item should be shipped to the customer is a separate package versus being grouped together in a box.
Vendor Style	No	Not used by RMS.
Standard Unit Weight	No	Not used by RMS.
Single Price Flag	No	Not used by RMS.
Pre Ticket Flag	No	Not used by RMS.
Planned Residual	No	Not used by RMS.
Sortable	No	Not used by RMS.
Item Master UDA1	No	Not used by RMS.
Item Master UDA2	No	Not used by RMS.
Item Master UDA3	No	Not used by RMS.
Item Master UDA4	No	Not used by RMS.
Item Master UDA5	No	Not used by RMS.
Item Master UDA6	No	Not used by RMS.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Item Master UDA7	No	Not used by RMS.
Item Master UDA8	No	Not used by RMS.
Item Master UDA9	No	Not used by RMS.
Item Master UDA10	No	Not used by RMS.
Item Master UDA11	No	Not used by RMS.
Item Master UDA12	No	Not used by RMS.
Item Master UDA13	No	Not used by RMS.
Item Master UDA14	No	Not used by RMS.
Item Master UDA15	No	Not used by RMS.
Ship Alone	No	Not used by RMS.
Slottable	No	Not used by RMS.
Freight Class	No	Not used by RMS.
New Item	No	Not used by RMS.
Brand	No	Not used by RMS.
Break Case Ups	No	Not used by RMS.
Rigid	No	Not used by RMS.
Fragile	No	Not used by RMS.
Container Type	No	Not used by RMS.
Conveyable Flag	No	Not used by RMS.
Hazard Matl Code	No	Not used by RMS.
Velocity	No	Not used by RMS.
High Value Indicator	No	Not used by RMS.
Ticket Type	No	Not used by RMS.
Color	No	Not used by RMS.
Size1	No	Not used by RMS.
Fit	No	Not used by RMS.
Shade	No	Not used by RMS.
Single Contain Bulk	No	Not used by RMS.
Unit Pick System Code	No	Not used by RMS.
Roundable Flag	No	Not used by RMS.
Perishable Indicator	No	Not used by RMS.
Division	No	Not used by RMS.
Vendor Number	No	Not used by RMS.
Knitting WIP Code	Optional	WIP code used to designate the Kitting process used in item_master
Unit Ticket Quantity	No	Not used by RMS.
Item Length	No	Not used by RMS.
Item Width	No	Not used by RMS.
Item Height	No	Not used by RMS.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Item Weight	No	Not used by RMS.
Item Cube	No	Not used by RMS.
Standard Unit Quantity	No	Not used by RMS.
Expiration Days	No	Not used by RMS.
Putaway By Volume	No	Not used by RMS.
Putaway Plan	No	Not used by RMS.
Item Type	No	Not used by RMS.
Sorter Group	No	Not used by RMS.
SKU Optimization	No	Not used by RMS.
Ext Source System	No	Not used by RMS.
Item Xform Indicator	Optional	This indicator will show that an item is associated with an item transformation. The item will be either the sellable item or orderable item in the transformation process.
Order Type	Optional	This determines how catch weight items are ordered. Valid values are: F - fixed weight V - Variable weight Valid values are held on the CODE_DETAIL table with a code type = ORDT
Sale Type	Optional	This indicates the method of how catch weight items are sold in store locations. Valid values are: V - variable weight each L - Loose weight. Valid values are held on the CODE_DETAIL table with a code type = STPE
Deposit Item Type	Optional	This is the deposit item component type. A NULL value in this field indicates that this item is not part of a deposit item relationship. The possible values are - E - Contents A - Container Z - Crate T - Returned Item (Empty bottle) P - Complex pack (with deposit items) The Returned item is flagged only to enable these items to be mapped to a separate GL account if required.
Container Item	Optional	This holds the container item number for a contents item. This field is only populated and required if the DEPOSIT_ITEM_TYPE = E.
Deposit in Price Per UOM	Optional	This field indicates if the deposit amount is included in the price per UOM calculation for a contents item ticket. This value is only required if the DEPOSIT_ITEM_TYPE = E. Valid values are I - Include deposit amount E - Exclude deposit amount
Transport Indicator	No	Not used by RMS.
Notional Pack Indicator	Always	Indicates to the Store Inventory System if a sellable simple pack needs to be broken down to its component item.
SOH Inquiry at Pack Indicator	Always	Indicates to the Store Inventory System if a notional simple pack item's inventory should be displayed in packs.
Purchase Type	Optional	Indicates if an item is a normal, consignment or concession item - 'N' for normal, 'S' for consignment, 'C' for concession.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Product Classification	Optional	This contains item combinability codes (with code type 'PCLA') which provide a way to define which items can be combined (packed or boxed) together.
Item Aggregate Indicator	Always	Indicator to aggregate inventory and sales for the item. Currently, this indicator is used by allocation and MFP to aggregate inventory for items. For staple items this indicator should be N.
Diff 1 Aggregate Indicator	Always	Indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (for example, Style/Color or Style/Size).
Diff 2 Aggregate Indicator	Always	Indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (for example, Style/Color or Style/Size).
Diff 3 Aggregate Indicator	Always	Indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (for example, Style/Color or Style/Size).
Diff 4 Aggregate Indicator	Always	Indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (for example, Style/Color or Style/Size).
Item Description Secondary	Optional	Secondary descriptions of the item. This field can only be populated when <code>system_options.secondary_desc_ind = Y</code> .
Original Retail	Optional	The field indicated the original retail price of the item per unit. This field is stored in the primary currency.
Original Retail Currency Code	Optional	The field indicates the currency code of the original retail price.
Mfg Retail Currency Code	Optional	The field indicates the currency code of the Manufacturer's retail price.
Catch Weight Type	Optional	This column will hold catch weight type for a simple pack catch weight item. The value is based on the component items <code>order_type</code> and <code>sale_type</code> : 2 - <code>order_type</code> = Variable Weight, <code>sale_type</code> = Loose Weight 4 - <code>order_type</code> = Variable Weight, <code>sale_type</code> = Variable Weight Each The column will be set only at the time of Item approval.
Catch Weight UOM	Optional	Indicates the UOM for Catchweight Items.
Brand Name	Optional	This field contains the brand associated to an item.
Unique Class	Optional	This field uniquely identify the class which the item belongs to. Class is not a unique ID in the merchandise hierarchy. The combination of Dept/Class is unique but requires use of a composite key. The composite key is generally used in user interfaces. The unique ID can be used in back end processing or in systems that cannot have a composite key in the merchandise hierarchy.

Table 2-37 (Cont.) Item Header

Message Element	Included?	Notes
Unique Subclass	Optional	This field uniquely identify the subclass which the item belongs to. Subclass is not a unique ID in the merchandise hierarchy. The combination of Dept/Class/Subclass is unique but requires use of a composite key. The composite key is generally used in user interfaces. The unique ID can be used in back end processing or in systems that cannot have a composite key in the merchandise hierarchy.
Diff 1 Level	Optional	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
Diff 1 Description	Optional	This field will hold the description of differentiator.
Diff 2 Level	Optional	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
Diff 2 Description	Optional	This field will hold the description of differentiator.
Diff 3 Level	Optional	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
Diff 3 Description	Optional	This field will hold the description of differentiator.
Diff 4 Level	Optional	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
Diff 4 Description	Optional	This field will hold the description of differentiator.
Prod Classification Description	Optional	This field will hold the description of the product classification.
Primary Image URL	Optional	This field will hold the primary item image URL. It is only included for the ItemCre and ItemFulRep message types. Updates to the primary image for an item are published in the Item Image node.
Long Description	Optional	Long description of the item.

Table 2-38 Item Supplier

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
Primary Supplier Indicator	Always	This field indicates whether this supplier is the primary supplier for the item. An item can have one and only one primary supplier.
VPN	Optional	This field contains the Vendor Product Number for the Item/Supplier.
Supp Label	Optional	This field contains the supplier label. It will only be populated if item is a parent item.
Consignment Rate	Optional	This field contains the consignment rate for this item for the supplier.
Supplier Diff 1	Optional	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.

Table 2-38 (Cont.) Item Supplier

Message Element	Included?	Notes
Supplier Diff 2	Optional	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
Supplier Diff 3	Optional	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
Supplier Diff 4	Optional	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
Pallet Name	Always	Code referencing the name used to refer to the pallet.
Case Name	Always	Code referencing the name used to refer to the case.
Inner Name	Always	Code referencing the name used to refer to the inner.
Supplier Discontinue Date	Optional	Date when the supplier discontinues an item.
Direct Ship Indicator	Always	A value of 'Y' (Yes) in this field indicates that any item associated with this supplier is eligible for a direct shipment from the supplier to the customer.
Primary Case Size	Optional	This field indicates the primary case size for the item supplier when an orderable item is configured for informal case types.

Table 2-39 Item Supplier Country

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
Origin Country Id	Always	The country where the item was manufactured or significantly altered.
Primary Supplier Indicator	Always	This field indicates whether this supplier is the primary supplier for the item. Each item can have one and only one primary supplier.
Primary Country Indicator	Always	This field indicates whether this country is the primary country for the item/supplier. Each item/supplier combination can have one and only one primary country.
Unit Cost	Always	This field contains the current corporate unit cost for the item from the supplier /origin country. This field is stored in the Supplier's currency.
Lead Time	Optional	This field contains the number of days that will elapse between the date an order is written and the delivery to the store or warehouse from the supplier.
Pick up Lead Time	Optional	Contains the time that will be taken to get the item from the supplier to the initial receiving location.
Supplier Pack Size	Always	This field contains the quantity that orders must be placed in multiples of for the supplier for the item.
Inner Pack Size	Always	This field contains the break pack size for this item from the supplier.

Table 2-39 (Cont.) Item Supplier Country

Message Element	Included?	Notes
Round Level	Always	This column is used to determine how order quantities will be rounded to case, layer and pallet.
Minimum Order Quantity	Optional	This field contains the minimum quantity that can be ordered at once from the supplier for the item.
Max Order Quantity	Optional	This field contains the maximum quantity that can be ordered at once from the supplier for the item.
Packing Method	Optional	This field indicates whether the packing method for the item in the container is flat or hanging.
Default Unit of Purchase	Optional	Contains the default unit of purchase for the item/supplier/country. Valid values include 'C' (Case) and 'P' (Pallet).
Ti	Always	Number of shipping units (cases) that make up one tier of the pallet. Multiply TI x HI to get total number of units (cases) in a Pallet.
Hi	Always	Number of tiers that make up a complete pallet. Multiply TI x HI to get total number of units (cases) in a Pallet.
Cost Unit of Measure	Always	A cost UOM is held to allow costs to be managed in a separate UOM to the standard UOM.
Tolerance Type	Optional	The unit of tolerance for catch weight simple packs. Valid values are: 'A' - actual 'P' - percent.
Max Tolerance	Optional	The maximum tolerance value for the catch weight simple pack.
Min Tolerance	Optional	The minimum tolerance value for a catch weight simple pack.
Round to Inner Pct	Always	This column will hold the Inner Rounding Threshold value. During rounding, this value is used to determine whether to round partial Inner quantities up or down. If the Inner-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
Round to Case Pct	Always	This column will hold the Case Rounding Threshold value. During rounding, this value is used to determine whether to round partial Case quantities up or down. If the Case-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
Round to Layer Pct	Always	This column will hold the Layer Rounding Threshold value. During rounding, this value is used to determine whether to round partial Layer quantities up or down. If the Layer-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
Round to Pallet Pct	Always	This column will hold the Pallet Rounding Threshold value. During rounding, this value is used to determine whether to round partial Pallet quantities up or down. If the Pallet -fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
Supplier Hierarchy Type 1	Optional	This field identifies partner type of supplier hierarchy level 1. This field will always have the partner type S1 which indicates manufacturer.

Table 2-39 (Cont.) Item Supplier Country

Message Element	Included?	Notes
Supplier Hier Level 1	Optional	Highest level of supplier hierarchy which is there to indicate a partner, such as a manufacturer, in the supply chain that gives rebates to the retailer.
Supplier Hierarchy Type 2	Optional	This field identifies partner type of supplier hierarchy level 2. This field will always have the partner type S2 which indicates distributor.
Supplier Hier Level 2	Optional	Second highest level of supplier hierarchy which is there to indicate a partner, such as a distributor, in the supply chain that gives rebates to the retailer.
Supplier Hierarchy Type 3	Optional	This field identifies partner type of supplier hierarchy level 3. This field will always have the partner type S3 which indicates wholesaler.
Supplier Hier Level 3	Optional	Third highest level of supplier hierarchy which is there to indicate a partner, such as a wholesaler, in the supply chain that gives rebates to the retailer.
Negotiated Item Cost	Optional	This will hold the supplier negotiated item cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations negotiated item cost will be stored in this field.
Extended Base Cost	Optional	This will hold the extended base cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations extended base cost will be stored in this field. Extended base cost is the cost inclusive of all the taxes that affect the WAC. In case of GTAX, Extended Base Cost = Base Cost + Non-recoverable taxes. In case of VAT, Extended Base Cost = Base Cost.
Inclusive Cost	Optional	This will hold the inclusive cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations inclusive cost will be stored in this field. This cost will have both the recoverable and non-recoverable taxes included. In case of GTAX, Inclusive Cost = Base Cost + Non-recoverable taxes + Recoverable Taxes. In case of VAT, Inclusive Cost = Base Cost + VAT.
Base Cost	Optional	This field will hold the tax exclusive cost of the item.
Purchase Type	Optional	This contains a code to indicate whether the item is normal merchandise (i.e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
Calculation Basis	Optional	This indicates if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
Purchase Rate	Optional	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 2-39 (Cont.) Item Supplier Country

Message Element	Included?	Notes
Pricing Control	Optional	This field contains the pricing control of the item which will determine the item price is controlled by Retailer or Supplier when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are • R – Retailer • S – Supplier For owned items, the value is always NULL.

Table 2-40 Item Supplier Country Dimension

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
Origin Country Id	Always	The country where the item was manufactured or significantly altered.
DIM Object	Always	Specific object whose dimensions are specified in this record (e.g. case, pallet, each). Valid values for this field are in the code type DIMO on the codes tables.
Presentation Method	Optional	Describes the packaging (if any) being taken into consideration in the specified dimensions. Valid values for this field are in the code type PCKT on the codes tables.
Method Description	Optional	The description for the Presentation Method.
Length	Optional	Length of dim_object measured in units specified in terms of the LWH Unit of Measure.
Width	Optional	Width of dim_object measured in units specified in terms of the LWH Unit of Measure.
Height	Optional	Height of dim_object measured in units specified in terms of the LWH Unit of Measure.
LWH Unit of Measure	Optional	Unit of measurement for length, width, and height.
Weight	Optional	Weight of dim_object measured in units specified in terms of the Weight Unit of Measure.
Net Weight	Optional	Net weight of the dim_object (weight without packaging) measured in units specified in terms of the Weight Unit of Measure.
Weight Unit of Measure	Optional	Unit of measurement for Weight and Net Weight.
Liquid Volume Unit of Measure	Optional	Unit of measurement for liquid volume. Liquid volumes are only convertible to other liquid volumes.
Stat Cube	Optional	Statistical value of the dim_object's dimensions to be used for loading purposes.
Tare Weight	Optional	Amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dim_object.

Table 2-40 (Cont.) Item Supplier Country Dimension

Message Element	Included?	Notes
Tare Type	Optional	Indicates whether the tare weight for the dim_object is wet or dry.

Table 2-41 Item User Defined Attribute List of Value

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
UDA ID	Always	This field contains a number identifying the user-defined attribute.
UDA Value	Always	Contains the value for the user-defined attribute.
UDA Description	Always	This field contains a description of the User-Defined Attribute.
UDA Value Description	Always	This field contains a description of the UDA value.

Table 2-42 Item User Defined Attribute FF

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
UDA ID	Always	This field contains a number identifying the user-defined attribute.
UDA Text	Always	This field contains the text value of the user-defined attribute for the item.

Table 2-43 Item User Defined Date

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
UDA Id	Always	This field contains a number identifying the user-defined attribute.
UDA Date	Always	This field contains the effective date for the UDA ID.

Table 2-44 Item Image

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Image Name	Always	Name of the image.
Image Addr	Always	Path where the image file is stored.
Image Type	Always	This field contains the type of the image of the item. Valid values are defined as member of IITD code type.
Primary Indicator	Always	Field to specify the display sequence order of images associated to the item per priority.

Table 2-44 (Cont.) Item Image

Message Element	Included?	Notes
Image Description	Always	This field contains the description associated with the image of the item.

Table 2-45 Item UPC

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Item Parent	Always	Unique identifier for the item/group at the level above the item.
Primary Ref Item Indicator	Always	Indicates if the sub-transaction level item is designated as the primary sub-transaction level item. For transaction level items and above, the value in this field will be 'N'o.
Format ID	Optional	This field will hold the format id that corresponds to the item's variable UPC. This value is only used for items with variable UPCs.
Prefix	Optional	This field holds the prefix for variable weight UPCs. The prefix determines the format of the eventual UPC and will be used to decode variable weight UPCs that are uploaded from the POS.
Item Number Type	Always	Code specifying what type the item is. Valid values for this field are in the code type UPCT on the code_head and code_detail tables.

Item BOM information will be published only for an item creation message and full replacement message.

Table 2-46 Item BOM

Message Element	Included?	Notes
Pack No	No	Unique identifier for the pack item.
Item	No	Unique identifier for the component item.
Pack Quantity	No	Contains the quantity of each item in the pack item.

Table 2-47 Item Ticket Header

Message Element	Included?	Notes
Item	Always	Unique identifier for the pack item.
Ticket Type Id	Always	This field uniquely identifies the ticket type which is associated with the item.
Ticket Type Description	Always	This field contains a description of the ticket or label type.
Shelf Edge Label Indicator	Always	Indicates if the ticket type is a shelf edge label.
PO Print Type	Optional	This field indicates when the ticket type for the given item should be printed by default, upon the approval or receipt of the purchase order. Valid values are: A and R.

Table 2-47 (Cont.) Item Ticket Header

Message Element	Included?	Notes
Print on PC Indicator	Always	This field indicates whether this type of ticket should be printed for this item when a permanent price change goes into effect. Valid values in this field are: Y - Yes N - No.
Ticket Over Pct	Optional	This field contains a percent which indicates the number of tickets which should be printed for a given event. If the event causing tickets to be printed is approving or receiving a purchase order, then this field indicates the quantity of tickets which should be printed greater than the quantity of the purchase order. If the event causing the tickets to be printed is a permanent price change, this field would indicate the quantity of tickets which should be printed greater than the stock on hand.

Table 2-48 Item Ticket Detail

Message Element	Included?	Notes
Ticket Item Id	Optional	This field contains a character string which uniquely identifies an attribute which will appear on a ticket or label such as retail price or price per unit of measure.
Ticket Item Id Description	Optional	This field contains the description of the ticket_item_id.
UDA Id	Optional	This field contains a number which uniquely defines a user-defined attribute which is to be printed on this ticket type.

Table 2-49 Item Supplier Country Manufacturer

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
Manufacturer Country Id	Always	Country ID of manufacturer.
Primary Manufacturer Country Indicator	Always	Primary country of manufacturer indicator.

Table 2-50 Related Item Header

Message Element	Included?	Notes
Item	Always	The main item against which the relationship is defined.
Relationship Id	Always	A system generated numeric field that uniquely identifies the relationship record in related_item_head table.
Relationship Name	Always	This is the user entered name of the relationship.
Relationship Type	Always	Describes the type of relationship. Valid values are in the codes table under code_type 'IREL'.
Mandatory Indicator	Always	Indicates whether the relationship is mandatory or not. Valid values: 'Y' (yes), 'N' (no).

Table 2-51 Related Item Detail

Message Element	Included?	Notes
Related Item	Always	Item for which related items are defined. Valid transaction level/sellable item. Multiple items can be defined for a relationship.
Priority	Optional	Relationship priority. Applicable only for relationship type Substitute.
Effective Date	Optional	The related item can be used on transactions starting this date.
End Date	Optional	The related item can be used on transactions until this date. A value of null means that it is always effective.

VAT Item BOM information will be published only when the default tax type is not GTS.

Table 2-52 Vat Item

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Active Date	Always	This field holds the active date for record on future cost.
Vat Type	Always	This field indicates the type of vat either Retail or Cost or Both.
Vat Code	Always	This field indicates a code to uniquely identify a VAT rate.
Vat Rate	Always	This column contains vat rate associated with a given VAT code.
Reverse Vat Indicator	Always	This field indicates if the item is subject to reverse charge VAT at the vat region. Valid values are Y and N.
Vat Region	Always	Contains the number of the Value Added Tax region a location belongs to.

Table 2-53 Item Supplier Unit of Measure

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
UOM	Always	The unit of measure that the item is being measured in.
Value	Always	This field will store the equivalent value of the Item/ Suppliers shipping carton in the associated unit of measure.

Modify and Delete Messages

Updates and deletions of item data can be included in a larger item creation message. If not part of a larger hierarchical message, they are published individually as a flat, non-hierarchical message. Update and delete messages are much smaller than the large hierarchy in a newly created item message (ItemDesc).

Modify Messages

If an existing item is modified, this will trigger an update message to provide details of the update through the RIB. The message will include any updated levels in the item hierarchy.

Delete Messages

Delete messages are published in the same way as modify messages. For example, if an item-supplier-country relationship is deleted from Merchandising, the dependent item-supplier-country dimension records are also deleted.

Table 2-54 Item

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.

Table 2-55 Item Supplier

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.

Table 2-56 Item Supplier Country

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
Country Id	Always	The country where the item was manufactured or significantly altered.

Table 2-57 Item Supplier Country Dimension

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
Country Id	Always	The country where the item was manufactured or significantly altered.
DIM Object	Always	Dim_object description fetched from the code_detail table.

Table 2-58 Item UPC

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.

Table 2-59 Item User Defined Attribute FF

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
UDA Id	Always	This field contains a number identifying the user-defined attribute.
UDA Text	Always	This field contains the text value of the user-defined attribute for the item.

Table 2-60 Item User Defined Attribute Date

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
UDA Id	Always	This field contains a number identifying the user-defined attribute.
UDA Date	Always	This field contains the effective date for the UDA ID.

Table 2-61 Item User Defined Attribute List of Value

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
UDA Id	Always	This field contains the unique identified of the Attribute Group being added or delete to the item list.
UDA Value	Always	Contains the value of the Attribute Group.

Table 2-62 Item Image

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Image Name	Always	Name of the image.

Table 2-63 Item Ticket

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Ticket Type Id	Always	This field uniquely identifies the ticket type which is associated with the item.

Table 2-64 Item Supplier Country Manufacturer

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	This field uniquely identifies the ticket type which is associated with the item.

Table 2-64 (Cont.) Item Supplier Country Manufacturer

Message Element	Included?	Notes
Country Id	Always	The country where the item was manufactured or significantly altered.

Table 2-65 Related Item

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Relationship Id	Always	A system generated numeric field that uniquely identifies the relationship record in related_item_head.
Related Item	Always	Item for which related items are defined. Valid transaction level/sellable item. Multiple items can be defined for a relationship.

Table 2-66 VAT Item

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Active Date	Always	This field holds the active date for record on future cost.
Vat Type	Always	This field indicates the type of vat either Retail or Cost or Both.
Vat Region	Always	Contains the number of the Value Added Tax region a location belongs to.

Table 2-67 Item Supplier Unit of Measure

Message Element	Included?	Notes
Item	Always	Unique identifier for the item.
Supplier	Always	Unique identifier for the supplier.
UOM	Always	The unit of measure that the item is being measured in.

Flex Attributes

If any custom flex attributes (CFAS) for the item have been added or modified, it will trigger an item header modify message. All the entity's active flex attributes from all attribute groups are published as key-value pairs based on the group set view. This CFAS object is embedded in the outbound Item message.

If any CFAS for the item-supplier has been added or modified, it will trigger an item supplier modify message. All the entity's active flex attributes from all attribute groups are published as key-value pairs based on the group set view. This CFAS object is embedded in the outbound item supplier message.

Similarly, if any CFAS for the item-supplier-country has been added or modified, it will trigger an item supplier modify message. All the entity's active flex attributes from all attribute groups are published as key-value pairs based on the group set view. This CFAS object is embedded in the outbound item supplier country message.

Flex attributes can only be added to or updated; they cannot be deleted.

Table 2-68 Flex Attributes

Message Element	Required?	Notes
Name	Always	Holds the attribute name.
Value	Optional	Holds the value of the attribute for number and character type attributes
Value Date	Optional	Holds the date for date type attributes.

Full Message Updates

In cases where the integrating system is unable to receive only the changes, another option is provided that can resend the item details whenever there is a change, in addition to the deltas. This will be published when the system option Publish Full Objects (PUB_FULL_OBJECTS_IND) is set to Deltas and Full (Y). This message will contain the header-level information, its flex attributes, pack and reference item information. This is used for Oracle WMS Cloud integration.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message in which the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Here are the filenames that correspond with each message type. Please consult RIB documentation for each message type to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
itemcre	Item Create Message	ItemDesc.xsd
itemmod	Item Modify Message	ItemDesc.xsd
itemdel	Item Delete Message	ItemRef.xsd
itemsupcre	Item Supplier Create Message	ItemSupDesc.xsd
itemsupmod	Item Supplier Modify Message	ItemSupDesc.xsd
itemsupdel	Item Supplier Delete Message	ItemSupRef.xsd
itemsupctycre	Item Supplier Country Create Message	ItemSupCtyDesc.xsd
itemsupctymod	Item Supplier Country Modify Message	ItemSupCtyDesc.xsd
itemsupctydel	Item Supplier Country Delete Message	ItemSupCtyRef.xsd
iscdimcre	Item Supplier Country Dimension Create Message	ISCDimDesc.xsd
iscdimmod	Item Supplier Country Dimension Modify Message	ISCDimDesc.xsd
iscdimdel	Item Supplier Country Dimension Delete Message	ISCDimRef.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
itemupccre	Item UPC Create Message	ItemUPCDesc.xsd
itemupcmod	Item UPC Modify Message	ItemUPCDesc.xsd
itemupcdel	Item UPC Delete Message	ItemUPCRef.xsd
itemudaffcre	Item UDA Free Form Text Create Message	ItemUDAFFDesc.xsd
itemudaffmod	Item UDA Free Form Text Modify Message	ItemUDAFFDesc.xsd
itemudaffdel	Item UDA Free Form Text Delete Message	ItemUDAFFRef.xsd
itemudalovcre	Item UDA LOV Create Message	ItemUDALOVDesc.xsd
itemudalovmod	Item UDA LOV Modify Message	ItemUDALOVDesc.xsd
itemudalovdel	Item UDA LOV Delete Message	ItemUDALOVRef.xsd
itemudadatecre	Item UDA Date Create Message	ItemUDADateDesc.xsd
itemudadatemod	Item UDA Date Modify Message	ItemUDADateDesc.xsd
itemudatedel	Item UDA Date Delete Message	ItemUDADateRef.xsd
itemimagecre	Item Image Create Message	ItemImageDesc.xsd
itemimagemod	Item Image Modify Message	ItemImageDesc.xsd
itemimagedel	Item Image Delete Message	ItemImageRef.xsd
relitemheadcre	Item Relationship Create Message	RelatedItemDesc.xsd
relitemheadmod	Item Relationship Modify Message	RelatedItemDesc.xsd
relitemheaddel	Item Relationship Delete Message	RelatedItemRef.xsd
relitemdetcre	Related Item Create Message	RelatedItemDesc.xsd
relitemdetmod	Related Item Modify Message	RelatedItemDesc.xsd
relitemdetdel	Related Item Delete Message	RelatedItemRef.xsd
itemfulrep	Item Full Replacement Message	ItemDesc.xsd

Merchandise Hierarchy Publication API

This section describes the merchandise hierarchy publication API.

Functional Area

Merchandise Hierarchy

Business Overview

This API publishes information regarding all the levels of the merchandise hierarchy to the RIB such that all the downstream applications may subscribe to it and have merchandise hierarchy information in sync with Merchandising.

New Division

When a new division is created in Merchandising, it triggers a message to be sent to notify external systems. The full details are sent for the new division as part of the create message, the division ID, division name, buyer, merchandiser, and total market amount.

Updated Division

When an existing division is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields.

The format used when creating or modifying a division is shown below.

Table 2-69 Division Create and Update

Message Element	Included?	Notes
Division	Always	Contains the number which uniquely identifies the division of the company.
Division Name	Always	Contains the name which, along with the division number, identifies the division of the company.
Buyer	Optional	Contains the number of the buyer associated with the division.
Merchandiser	Optional	Contains the number of the merchandiser associated with the division.
Total Market Amount	Optional	Contains the total market amount that is expected for the division. The field can be used for analytics or reporting.

Deleted Division

When an existing division is deleted, this will also trigger a delete transaction which to be sent to notify external systems that this division is no longer valid. The delete message will include only the ID of the division being deleted.

Table 2-70 Division Delete

Message Element	Included?	Notes
Division	Always	Contains the number which uniquely identifies the division of the company.

New Group

When a new group is created in Merchandising, it triggers a message to be sent to notify external systems. The full details are sent for the new group as part of the create message, the group ID, group name, buyer, merchandiser, and division.

Updated Group

When an existing group is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields.

Table 2-71 Group Create and Update

Message Element	Included?	Notes
Group Number	Always	Contains the number which uniquely identifies the group.
Group Name	Always	Contains the description which, along with the group number, identifies the group.
Buyer	Optional	Contains the number of the buyer associated with the group.
Merchandise	Optional	Contains the number of the merchandiser associated with the group.
Division	Always	Contains the number of the division of which the group is a member.

Deleted Group

When an existing group is deleted, this will also trigger a delete transaction which to be sent to notify external systems that this group is no longer valid. The delete message will include only the group ID being deleted.

Table 2-72 Group Delete

Message Element	Included?	Notes
Group Number	Always	Contains the number which uniquely identifies the group.

New Department

When a new department is created in Merchandising, it triggers a message to be sent to notify external systems. The full details are sent for the new department as part of the create message, the department ID, department name, buyer, purchase type, total market amount, merchandiser, group, budgeted markup, profit calc type, markup calc type, OTB calc type, budgeted intake percentage, and department-level VAT inclusive indicator. The custom flex attributes, if applicable, are also sent as a part of the message.

Updated Department

When an existing department is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields, including custom flex attributes.

Table 2-73 Department Create and Update

Message Element	Included?	Notes
Department	Always	Contains the number which uniquely identifies the department.
Department Name	Always	Contains the description which identifies the department.

Table 2-73 (Cont.) Department Create and Update

Message Element	Included?	Notes
Buyer	Optional	Contains the number which uniquely identifies the buyer for that department.
Purchase Type	Always	Contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment, or concession. Items in the department can be a mix of these item types, this acts only as a default. Valid values are: 0 - Normal Merchandise 1 - Consignment Stock 2 - Concession Stock
Total Market Amount	Optional	Contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
Merchandiser	Optional	Contains the number which uniquely describes the merchandiser for that department.
Group Number	Always	Contains the number of the group in which the department exists.
Budgeted Markup	Always	Contains the markup percent of cost that is a target for this department.
Profit Calc Type	Always	Contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department. Valid values are: 1 = Direct Cost 2 = Retail Inventory
Markup Calc Type	Always	Contains the code which determines how markup is calculated in this department. Valid values are: C - Cost R - Retail
OTB Calc Type	Always	Contains the code letter which determines how open to buy is calculated in this department. Valid values are: C - Cost R - Retail
Maximum Average Counter	Optional	A maximum average counter will hold the maximum count of days with acceptable data to include in an average for items within the department.
Average Tolerance Percentage	Optional	A tolerance percentage value used in averaging for items within this value. This value will set up a range for appropriate data and constrain outliers. This column will hold 70% as 70 - not 0.70.
Budgeted Intake	Always	Contains the markup percent of retail that is a target for this department.
Department VAT Include Indicator	Always	This flag is no longer used by Merchandising and will always be defaulted to Y.
Flex Attributes	Optional	Child Node

Deleted Department

When an existing department is deleted, this will also trigger a delete transaction which to be sent to notify external systems that this department is no longer valid. The delete message will include only the department ID being deleted.

Table 2-74 Department Delete

Message Element	Included?	Notes
Department	Always	Contains the number which uniquely identifies the department being deleted.

New Class

When a new class is created in Merchandising, it triggers a message to be sent to notify external systems. The full details are sent for the new class as part of the create message, the class ID, class name, class VAT indicator, and department. The custom flex attributes, if applicable, are also sent as a part of the message.

Updated Class

When an existing class is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields including custom flex attributes.

Table 2-75 Class Create and Modify

Message Element	Included?	Notes
Class	Always	Contains the number which uniquely identifies the class within the department.
Class name	Always	Contains the name of the class which, along with the class number, identifies the class.
Class VAT Indicator	Always	This field is no longer used by Merchandising and will always default to N.
Department	Always	Contains the number of the department of which the class is a member.
Unique Class ID	Always	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
Custom Flex Attributes	Optional	Child Node

Deleted Class

When an existing class is deleted, this will also trigger a delete transaction which to be sent to notify external systems that this class is no longer valid. The delete message will include only the class being deleted and its department.

Table 2-76 Class Delete

Message Element	Included?	Notes
Department	Always	Contains the number which uniquely identifies the department.
Class	Always	Contains the number which uniquely identifies the class within the department.

New Subclass

When a new subclass is created in Merchandising, it triggers a message to be sent to notify external systems. The full details are sent for the new subclass as part of the create message, the subclass ID, subclass name, department, and class. It also contains a unique ID for the subclass and class, which are not dependent on knowing the department and class IDs displayed in Merchandising. The custom flex attributes, if applicable, are also sent in as a part of the message.

Updated Subclass

When an existing subclass is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields including custom flex attributes.

Table 2-77 Subclass Create and Update

Message Element	Included?	Notes
Subclass	Always	Contains the number which uniquely identifies the subclass within the department and class.
Subclass name	Always	Contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
Class	Always	Contains the class number of which the subclass is a member.
Department	Always	Contains the department number of which the subclass is a member.
Custom Flex Attributes	Optional	Child Node
Unique Subclass ID	Always	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
Unique Class ID	Always	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Deleted Subclass

When an existing subclass is deleted, this will also trigger a delete transaction which to be sent to notify external systems that this class is no longer valid. The delete message will include only the subclass being deleted, along with its class and department.

Table 2-78 Subclass Delete

Message Element	Included?	Notes
Department	Always	Contains the number which uniquely identifies the department.
Class	Always	Contains the number which uniquely identifies the class within the department.
Subclass	Always	Contains the number which uniquely identifies the subclass within the department and class.

Custom Flex Attributes

If any custom flex attributes (CFAS) for the department, class, or subclass have been added or modified, it will trigger an update message. The node of the integration that supports this will contain the name of the attribute as it is defined in the group set level view, the value of the custom attribute. If it is a date attribute, the date value is in a separate field. Flex attributes can only be added to or updated; they cannot be deleted.

Table 2-79 Flex Attributes

Message Element	Included?	Notes
Name	Always	Holds the attribute name.
Value	Optional	Holds the value of the attribute for number and character type attributes
Value Date	Optional	Holds the date for date type attributes.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
divisoncre	Division Create Message	MrchHrDivDesc.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
divisonmod	Division Modify Message	MrchHrDivDesc.xsd
divisiondel	Division Delete Message	MrchHrDivRef.xsd
groupcre	Group Detail Create Message	MrchHrGrpDesc.xsd
groupmod	Group Detail Modify Message	MrchHrGrpDesc.xsd
groupdel	Group Detail Delete Message	MrchHrGrpRef.xsd
deptcre	Department Detail Create Message	MrchHrDeptDesc.xsd
deptmod	Department Detail Modify Message	MrchHrDeptDesc.xsd
deptdel	Department Detail Delete Message	MrchHrDeptRef.xsd
classcre	Class Detail Create Message	MrchHrClsDesc.xsd
classmod	Class Detail Modify Message	MrchHrClsDesc.xsd
classdel	Class Detail Delete Message	MrchHrClsRef.xsd
subclasscre	Subclass Detail Create Message	MrchHrScIsDesc.xsd
subclassmod	Subclass Detail Modify Message	MrchHrScIsDesc.xsd
subclassdel	Subclass Detail Delete Message	MrchHrScIsRef.xsd

Order Publication API

This section describes the order publication API.

Functional Area

Purchase Orders

Business Overview

Merchandising publishes new and updated purchase orders to inform external subscribing systems, such as the warehouse management system or store inventory management (SIM/SIOCS), of the changes. Only orders that have been approved are published and will be visible to external systems. Updates are provided synchronously in a near-real-time manner.

There are several ways a purchase order can be created in Merchandising, such as through the purchase order screen, using spreadsheet upload, from customer orders and franchise orders fulfilled from a supplier, through item replenishment, by a supplier in a vendor-managed inventory environment, and so on. All purchase orders are published in a similar manner, regardless of their source.

Merchandising publishes two sets of order messages to the RIB for two kinds of subscribing applications. The primary order information is sent at the physical location level, which includes both physical warehouses and stores. This is intended to be used by the warehouse and store applications. The second set of information contains order quantity by virtual location. Applications that understand virtual warehouses subscribe to these messages.

New Purchase Orders

Creating a new purchase order triggers a message that notifies external systems of the changes. The full details in the message are sent for the new order as part of the create message. It includes details such as which supplier is being ordered from, order dates, the

items that are ordered and the quantity, the size of the pack being ordered, the location that each item is going to be delivered to, and so on. If the order is for a customer order, (order_type = 'CO'), the customer order number and fulfillment order number retrieved from the ORDCUST table will be included in the header message and published.

New Purchase Order Details

Adding item/locations to an existing order triggers a detail create message to be created and published after the order is re-approved. This action will create one header modify message for the status change and each detail addition will create one detail create message. The detail create message will still contain both complete header and detail information.

Purchase Order Header

Table 2-80 Purchase Order Header

Message Element	Included?	Notes
Document Type	Always	This field indicates the type of document this message is for. Valid value is P for purchase order.
Order Number	Always	This field contains the number to uniquely identify an order within the system.
Order Type	Always	This field indicates the type of order. Valid values are: • N/B - Non Basic • ARB - Automatic Reorder of Basic • BRB - Buyer Reorder of Basic • CO - Customer Order
Order Type Description	Always	This field contains the order type description as shown above.
Department	Optional	This field contains the department number for orders limited to a single department and will be null for orders involving items in more than one department.
Department Name	Optional	This field contains the name of the department.
Buyer	Optional	This field contains the number associated with the buyer for the order.
Buyer Name	Optional	This field contains the name of the authorized buyer.
Supplier	Always	This field contains the supplier site that will provide the merchandise specified in the order.
Promotion	Optional	This field contains the promotion associated with the order.
Promotion Description	Optional	This field contains a description of the promotion. This value comes from Pricing.
QC Indicator	Always	This field determines whether quality control will be required when items for this order are received. Valid values are Yes (Y) and No (N).
Not Before Date	Optional	This field contains the first date that delivery of the order will be accepted.
Not After Date	Optional	This field contains the last date that delivery of the order will be accepted.

Table 2-80 (Cont.) Purchase Order Header

Message Element	Included?	Notes
Open To Buy End of Week Date	Optional	This field contains the OTB week the order amount should be placed into for the order type.
Earliest Ship Date	Optional	This field contains the earliest date when the items on the purchase order can be shipped by the supplier. This represents the earliest ship date of all the items on the order.
Latest Ship Date	Optional	This field contains the date after which the items on the purchase order can no longer be shipped by the supplier. This represents the greatest latest ship date of all the items on the order.
Close Date	Optional	This field contains the date when the order was closed.
Terms	Always	This field is an indicator identifying the sales terms for the order. These terms specify when payment is due and if any discounts exist for early payment.
Terms Code	Always	This field contains the alphanumeric representation of Term Name which acts as the Term code in Oracle Financials.
Freight Terms	Optional	This field contains the ID of the freight terms are related to the order. Valid values are in the FREIGHT_TERMS table.
Customer Order Indicator	Always	This field indicates whether the order is generated for a customer order fulfillment.
Payment Method	Optional	This field indicates how the purchase order will be paid. Valid options are: • LC - Letter of Credit • WT - Wire Transfer • OA - Open Account
Payment Method Description	Optional	This field contains the description of the payment method as shown above.
Backhaul Type	Optional	This field contains the type of backhaul allowance that will be applied to the order. Valid values are Calculated (C) or Flat rate (F).
Backhaul Type Description	Optional	This field contains the description of the backhaul type as shown above.
Backhaul Allowance	Optional	This field contains the backhaul allowance amount.
Ship Method	Optional	This field contains the method used to ship the items on the purchase order from the country of origin to the country of import.
Ship Method Description	Optional	This field contains the description for the shipping method.
Purchase Type	Optional	This field indicates what is included in the supplier's cost of the item. Valid values are: • BACK - Backhaul • DLV - Delivered • FOB - Free on Board • PUP - Pick-up

Table 2-80 (Cont.) Purchase Order Header

Message Element	Included?	Notes
Purchase Type Description	Optional	Contains the description for the purchase type as shown above.
Status	Always	Indicates the current status of the order. Valid values include: • W - Worksheet • S - Submitted • A - Approved • C - Closed
Ship Pay Method	Optional	This field contains the code indicating the payment terms for freight charges associated with the order. Valid values are found in code type SHMT.
Ship Pay Method Description	Optional	This field contains the description of the given shipment payment method. Valid values are found in code type SHMT.
Transportation Responsibility	Optional	This field contains the code indicating the type of location that is responsible for the transportation of the order. Valid values are found in code type FOBT.
Transportation Responsibility Code Description	Optional	This field contains the description of the transportation responsibility code. Valid values are found in code type FOBT.
Transportation Responsibility Description	Optional	This field contains a user entered field describing the code for the location responsible for the transportation of the order.
Title Pass	Optional	This field contains the code indicating where the title for goods is passed from the vendor to the purchaser. Valid values are found in code type FOBT.
Title Pass Code Description	Optional	This field contains the description for title pass code. Valid values are found in code type FOBT.
Title Pass Description	Optional	This contains a user entered field describing the code where the title of the merchandise is to be passed.
Vendor Order Number	Optional	This field contains the vendor's number for an order. These orders may have originated by the vendor or this number can be associated to a Merchandising order when the order is created on-line.
Exchange Rate	Optional	The exchange rate used to convert the order cost from the order's currency to local currency. This value is used when the entity exchange rate configuration's Ignore Entity Exchange Rate option for purchase order cost is set to No (N). When the Ignore Entity Exchange Rate flag is Yes (Y), the exchange rate is ignored and the value from the Merchandising currency rates table on receipt date or approval date is used.
Factory	Optional	This field contains the factory at which the items on the purchase order are made. This field is only available when the system is running Import Management functionality.
Factory Description	Optional	This field contains the description of the factory.

Table 2-80 (Cont.) Purchase Order Header

Message Element	Included?	Notes
Agent	Optional	This field contains the agent that is used in the purchase of the items on the purchase order. This field is only available when the system is running Import Management functionality.
Agent Description	Optional	This field contains the description of the agent.
Discharge Port	Optional	This field contains the port at which the items on the purchase order will enter the country of import. This field is only available when the system is running Import Management functionality.
Discharge Port Description	Optional	This field contains the description of the discharge port.
Lading Port	Optional	This field contains the port from which the items on the purchase order are shipped. This field is only available when the system is running Import Management functionality.
Lading Port Description	Optional	This field contains the description of the lading port.
Freight Contract Number	Optional	This field contains the number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import Management functionality.
Order Context (PO Type)	Optional	Contains a code representing the order context or type, which is a retailer defined attribute. Valid values are in the PO_TYPE table.
PO Type Description	Optional	This field contains the description of the PO Type.
Pre Mark Indicator	Always	This field indicates whether a supplier has agreed to pre-mark boxes on the order for the final destination location for a cross-docked order. Valid values are Yes (Y) and No (N).
Currency Code	Always	This field contains the currency code for the order. When creating an order within Merchandising, this field will always default to the currency of the supplier, but it can be overwritten.
Contract Number	Optional	This field contains the contract number associated with this order.
Pickup Location	Optional	This field contains the location at which the order will be picked up, if the order is a Pickup order.
Pickup Number	Optional	This field contains the reference number for the Pickup order.
Pickup Date	Optional	This field contains the date when the order can be picked up from the supplier. This field is only required if the Purchase Type of the order is Pickup.
Appointment Datetime	Optional	This field contains the date and time of the receiving appointment at the warehouse.
Comments	Optional	This field contains the miscellaneous comments attached to the purchase order.
Customer Order Number	Optional	This field holds the customer order number from the order management system for orders to be fulfilled from the supplier.

Table 2-80 (Cont.) Purchase Order Header

Message Element	Included?	Notes
Fulfill Order Number	Optional	This field holds the number from the order management system (OMS) related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS.
PO Detail	Optional	Child node
Flex Attribute	Optional	Child node
Reinstate Indicator	Always	This field indicates if the order was previously cancelled and has now been re-approved. It will contain a value of Y when the order is first reinstated. Any subsequent updates to the PO will have a value of N.

Purchase Order Detail

This component of the order rolls the order quantities for the purchase order up to the physical warehouse level. For stores, the quantity values here are the same as on the purchase order.

Table 2-81 Purchase Order Detail

Message Element	Included?	Notes
Item	Always	This field contains the unique alphanumeric value to identify the item included in the purchase order.
Reference Item	Optional	This field contains the unique alphanumeric value to identify the reference item (barcode).
Physical Location Type	Always	This field contains the type of location in the location field. Valid values are store (S) or warehouse (W).
Physical Location	Always	This field contains the actual physical location that the item will be shipped to. This field may contain a store or a physical warehouse.
Physical Store Type	Optional	This field indicates the store type of the physical location, if the physical location type is a store. Valid values are company store (C) or franchise store (F).
Physical Stockholding Indicator	Optional	Indicates if the store is stockholding or not, if the physical location type is a store. Valid values are Yes (Y) and No (N).
Physical Quantity Ordered	Optional	This field indicates the total quantity ordered to the physical location.
Transaction UOM	Optional	This field holds the original transaction unit of measure to be passed on to the store system. It is only used for customer orders shipped to a store for fulfillment. This will be based on the standard UOM for the item.
Unit Cost	Optional	This field contains the unit cost for the item/location on the order, including discounts, but not landed cost components in order currency.
Origin Country ID	Optional	This field contains the identifier of the country into which the items on the order are being imported.

Table 2-81 (Cont.) Purchase Order Detail

Message Element	Included?	Notes
Supplier Pack Size	Optional	This field contains the supplier pack size for this order/item.
Earliest Ship Date	Optional	This field contains the earliest date that the item can be shipped by the supplier.
Latest Ship Date	Optional	The date after which the item can be shipped by the supplier.
Pickup Location	Optional	This field contains the location at which the item will be picked up, if the order is a Pickup order.
Pickup Number	No	Not used in Merchandising
Packing Method	Optional	This field indicates whether the packing method of the item in the container is flat or hanging. Valid values are defined under code type PKMT.
Round Level	Optional	This column will be used to determine how order quantities will be rounded. Valid values are in code type ORL.
Door Indicator	No	Not used in Merchandising
Priority Level	No	Not used in Merchandising
New Item	Optional	This flag is used to identify a new item versus an existing item on the order. This is set to Y only if the system option for Publish RIB Objects is set to Full and Deltas (Y).
Quarantine	No	Not used in Merchandising
Received Unit Quantity	No	Not used in Merchandising
TSF PO Link ID	Optional	This field contains a reference number to link the item on a transfer to the purchase orders. This is used by orders created for items on Warehouse Cross Link replenishment.
PO Virtual Location Detail	Optional	Child node
Cost Source	Optional	This field indicates the basis for the cost of the item/location on the order. Valid values are found in code type ORCS.
Estimated In-Stock Date	Optional	This field contains the date that the item on the PO is expected to be in stock at the location.
Flex Attribute	Optional	Child node
Item Line Number	Optional	This field indicates the item line number. This is used for customer order POs.

Purchase Order Virtual Location Detail

This node of the message breaks out the order quantities down to the virtual warehouse level. For stores, the order quantities will be the same as in the physical location detail node.

Table 2-82 Purchase Order Virtual Location Detail

Message Element	Included?	Notes
Location Type	Always	This field contains the type of location in the location field. Valid values are store (S) or virtual warehouse (W).
Location	Always	This field contains the store number or the virtual warehouse number, depending on the virtual location type.
Quantity Ordered	Optional	This field contains the total quantity of the item ordered to this location.
Flex Attribute	Optional	Child node

Updated Purchase Orders

Purchase order updates trigger a message to notify external systems, based on updates made at the order header level and/or the detail level. The message that will be published will either contain the header information that changed in the message for header updates such as date changes, just the detail level information for detail level changes such as quantity changes or addition of items, or both.

Header Only Updates

When updates are made at the order header level such as updating dates or changing the status of the order, only header information is published. Information specific to items and locations are not included. Changes in header information, unapproving a purchase order, or any action that may change of status of a pre-approved order, such as closing or reinstating an order trigger header level publication. Creating and updating header level custom flex attributes (CFAS) also trigger header level update publishing.

Detail Only Updates

When updates are made at the order detail level without changes to the header information, only detail information is published. Detail level updates include changes in quantity ordered, unit cost, and estimated in stock date. When an order has already been approved, no deletes can be made at the detail level; but the quantity on the order can be cancelled. This action will trigger an update message to be sent. Creating and updating detail level custom flex attributes also trigger detail level publishing.

Full Message Updates

In cases where the solution receiving purchase orders cannot support just receiving the changes, another option is provided that can resend the full purchase order details whenever there is a change. This will be published, along with the delta messages, in cases where the system option Publish Full Objects (PUB_FULL_OBJECTS_IND) is set to Deltas and Full (Y). This message payload will contain a full snapshot of the order with the flex attributes (of both header and detail). This is also used for Oracle WMS Cloud integration.

Deleted Purchase Orders

Deleting approved purchase orders can be done by cancelling all items in the order, which will set the order status to closed. The purchase order will not be removed from the system.

Because both the order status and order quantities are changed, both header and detail information are published to external systems. The information included in the header and detail messages are the same as that described under the Updated Purchase Orders section.

PO Header Delete Message

Table 2-83 PO Header Delete Message

Message Element	Included?	Notes
Document Type	Always	This field indicates the type of document this message is for. Valid value is P for purchase order.
Order Number	Always	This field contains the number that uniquely identifies an order within the system.
PO Detail	Optional	Child node

PO Detail Delete Message

Table 2-84 PO Detail Delete Message

Message Element	Included?	Notes
Item	Always	This field contains the unique alphanumeric value to identify the item being deleted.
Physical Location Type	Always	This field contains the type of physical location where an item is being deleted. Valid values are store (S) or warehouse (W).
Physical Location	Always	This field contains the identifier of the physical location of the item being deleted.
Physical Store Type	Optional	This field indicates the store type of the physical location, if a store. Valid values are company store (C) or franchise store (F).
Physical Stockholding Indicator	Optional	This field indicates if the physical location is stockholding, if the location type is store. Valid values are Yes (Y) and No (N).

Flex Attributes

If any custom flex attributes (CFAS) for the order have been added or modified, it will trigger an order header or detail update message, as described above. The node of the integration that supports this will contain the name of the attribute as it is defined in the group set level view, the value of the custom attribute. If it is a date attribute, the date value is in a separate field. You can define group set views at the header level (ORDHEAD), order/item level (ORDSKU) or at the order/item/location (ORDLOC) levels. Flex attributes can only be added to or updated on an order; they cannot be deleted.

Flex Attributes

Table 2-85 Flex Attributes

Message Element	Required?	Notes
Name	Always	Holds the attribute name.

Table 2-85 (Cont.) Flex Attributes

Message Element	Required?	Notes
Value	Optional	Holds the value of the attribute for number and character type attributes
Value Date	Optional	Holds the date for date type attributes.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message in which the error occurred, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish these messages is available. In case the error is a fatal error, for example, when changes to data have already been made, a status of Error (E) is sent to the RIB and the message status in the queue will be in Error status. The error message as well as the object containing the order number and order details is returned to Merchandising.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
POCre	Purchase Order Create Message	PODesc.xsd
POCre (CustFlexAttriVo)	Purchase Order Flex Attribute Create Message	PODesc.xsd
PODtICre	Purchase Order Detail Create Message	PODesc.xsd
PODtICre (CustFlexAttriVo)	Purchase Order Detail Flex Attribute Create Message	PODesc.xsd
POHdrMod	Purchase Order Modify Message	PODesc.xsd
POHdrMod (CustFlexAttriVo)	Purchase Order Header Flex Attribute Modify Message	PODesc.xsd
PODtIMod	Purchase Order Detail Modify Message	PODesc.xsd
PODtIMod (CustFlexAttriVo)	Purchase Order Detail Flex Attribute Modify Message	PODesc.xsd
PODel	Purchase Order Delete Message for unpublished orders	PORef.xsd
PODtIDel	Purchase Order Detail Delete Message for unpublished orders	PORef.xsd
POFulRep	Purchase Order with Full payload Message	PODesc.xsd

Organizational Hierarchy Publication API

This section describes the organization hierarchy publication API.

Functional Area

Organizational Hierarchy

Business Overview

Merchandising publishes details about new, updated, and deleted levels of the organizational hierarchy (chain, area, region, and district) to external systems such that all the downstream applications and external systems may be informed of the updates. Publishing is provided synchronously in a near-real-time manner.

The entities sent as part of the new, update, or delete messages will always include the hierarchy level and hierarchy value. The following values will be published:

- If the message relates to chain, the message will contain a hierarchy level of 'CH' and the chain number.
- If the message relates to area, the message will contain a hierarchy level of 'AR' and the area number.
- If the message relates to a region, the message will contain a hierarchy level of 'RE' and the region number.
- If the message relates to a district, the message will contain a hierarchy level of 'DI' and the district number.

New Chains/Areas/Regions/Districts

Creating a new organizational level triggers a message to be sent to notify external systems. The hierarchy level, hierarchy value, description, manager name, currency code, parent ID and parent level are sent for the new organizational level as part of the create message.

Table 2-86 Organization Hierarchy – Create and Update

Message Element	Included?	Notes
Hierarchy level	Always	This field contains the level of the organizational hierarchy. Valid values are: • CH - Chain • AR - Area • RE - Region • DI - District
Hierarchy Value	Always	This field contains the hierarchy value.
Hierarchy Description	Optional	This field contains the description of the hierarchy value.
Manager Name	Optional	This contains the manager's name.
Currency Code	Optional	This field contains the currency code for the hierarchy.
Parent ID	Optional	This field contains the number that identifies the hierarchy level.
Parent Level	Optional	This field contains the parent level of the current organizational hierarchy. Valid values are: • CH - Chain • AR - Area • RE - Region

Updated Chains/Areas/Regions/Districts

Updating an organizational level triggers a message to notify external systems. The hierarchy level, hierarchy value, description, manager name, currency code, parent ID and parent level are sent for the updated organizational level as part of the update message.

Deleted Chains/Areas/Regions/Districts

Deleting an organizational level triggers a message to be sent to notify external systems. The hierarchy level, hierarchy value, parent ID, and parent level are sent for the deleted organizational level.

Table 2-87 Organization Hierarchy – Delete

Message Element	Included?	Notes
Hierarchy level	Always	This field contains the level of the organizational hierarchy. Valid values are: • CH - Chain • AR - Area • RE - Region • DI - District
Hierarchy Value	Always	This field contains the hierarchy value.
Parent ID	Optional	This field contains the number that identifies the hierarchy level.
Parent Level	Optional	This field contains the parent level of the current organizational hierarchy. Valid values are: • CH - Chain • AR - Area • RE - Region

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
chaincre	Chain Create Message	OrgHierDesc.xsd
chainmod	Chain Modify Message	OrgHierDesc.xsd
chaindel	Chain Delete Message	OrgHierRef.xsd
areacre	Area Create Message	OrgHierDesc.xsd
areamod	Area Modify Message	OrgHierDesc.xsd
areadel	Area Delete Message	OrgHierRef.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
regioncre	Region Create Message	OrgHierDesc.xsd
regionmod	Region Modify Message	OrgHierDesc.xsd
regiondel	Region Delete Message	OrgHierRef.xsd
districtcre	District Create Message	OrgHierDesc.xsd
districtmod	District Modify Message	OrgHierDesc.xsd
districtdel	District Delete Message	OrgHierRef.xsd

Partner Publication API

This section describes the partner publication API.

Functional Area

Partners

Business Overview

This API publishes partner, partner address information and custom flex attributes through the RIB to external systems. This information is further subscribed by integrating systems like Oracle Warehouse Management Cloud (WMS) and Store Inventory and Operations Cloud Service (SIOCS) that need to keep their partners synchronized with Merchandising. Partner information is published when new partners are created, updates are made to existing partners, or existing partners are deleted. Similarly, addresses are published when they are added, modified, or deleted.

Partner Creation

Partners can be setup in Merchandising though the application UI and it results in this message being published when a partner is created. The partner creation message is published from Merchandising only after the partner has been completely setup, including the mandatory addresses. Once all these criteria are met for a valid create message, the messages will be combined and sent to the RIB.

Table 2-88 Partner Header Create and Update

Message Element	Required?	Notes
Partner Type	Always	Specifies the type of the partner. Valid partner types are held on the code tables under code type PTNR.
Partner ID	Always	Unique number that identifies the partner within the system.
Partner Description	Always	Contains the partner's name.
Currency Code	Always	The code identifying the currency the partner uses for business transactions.
Language	Optional	The partner's preferred language.
Status	Always	Indicates if the partner is currently active. Valid values are A = Active or I = Inactive.
Contact Name	Optional	Name of the partner's representative contact

Table 2-88 (Cont.) Partner Header Create and Update

Message Element	Required?	Notes
Contact Phone	Optional	Phone number of the partner's representative contact.
Contact Fax	Optional	Fax number of the partner's representative contact.
Contact Telex	Optional	The telex number of the partner's representative contact.
Contact Email	Optional	The e-mail address of the partner's representative contact.
Manufacturer's ID	Optional	The manufacturer's tax identification number. Used for partner type Manufacturer.
Principal Country ID	Optional	The country id to which the partner is assigned.
Line of Credit	Optional	The line of credit the company has at the bank in the partner's currency. Used for partner type Bank.
Outstanding Credit	Optional	The total amount of credit that the company has used or has charged against in the partner's currency.
Open Credit	Optional	The total amount that the company can still charge against in the partner's currency.
Year to Date Credit	Optional	The total amount of credit the company has used this year to date in the partner's currency.
Year to Date Drawdowns	Optional	The year to date payments the bank has made on behalf of the company in the partner's currency.
Tax ID	Optional	The unique tax identification number of the partner. This will be used for reporting during the Customs Entry process.
Terms	Always	Payment terms for the partner.
Service Performed Required Indicator	Always	Indicates if the partner's services (e.g., snow removal) must be confirmed as performed before paying an invoice from that partner.
Invoice Pay Location	Optional	Indicates where invoices from this supplier are paid - at the store or centrally through corporate accounting.
Invoice Receive Location	Optional	Indicates where invoices from this supplier are received - at the store or centrally through corporate accounting.
Import Country ID	Optional	Import country of the Import Authority.
Primary Import Authority Indicator	Always	Indicates if an Import Authority is the primary Import Authority for an import country.
Comments	Optional	Comments associated with the partner.
Transfer Entity ID	Optional	The Transfer entity with which an external finisher is associated.
VAT Region	Optional	Tax region with which a partner is associated.
Org Unit ID	Optional	The org unit id with which partner is associated.
Partner Name Secondary	Optional	Secondary name of the partner.
Auto Receive Stock Indicator	Optional	Indicates whether the system will update the stock for the external finisher when the 1st leg of the transfer is shipped.

Table 2-88 (Cont.) Partner Header Create and Update

Message Element	Required?	Notes
Partner Address 1	Optional	The partner's primary address line 1.
Partner Address 2	Optional	The partner's primary address line 2.
City	Optional	The city of the partner's primary address.
County	Optional	The county of the partner's primary address.
State	Optional	The state of the partner's primary address.
Country ID	Optional	The country of the partner's primary address.
Postal Code	Optional	The postal code of the partner's primary address.
Partner Org Unit Details	Optional	Child Node
Address Details	Optional	Child Node
Flex Attributes	Optional	Child Node

Table 2-89 Partner Org Unit Detail

Message Element	Required?	Notes
Org Unit ID	Optional	Org unit ID of the partner.

Table 2-90 Address Detail

Message Element	Required?	Notes
State Name	Optional	The state name for the partner address of this type.
Country Name	Optional	The country name for the partner address of this type.
Address Key	Always	The unique address ID from Merchandising.
Address Type	Always	The type for the address. All address types are published for partners.
Primary Address Type Indicator	Optional	Indicates if the address type is primary.
Primary Address Indicator	Always	Indicates whether this address is the primary address for the address type.
Address Line 1	Always	The first line of the address.
Address Line 2	Optional	The second line of the address.
Address Line 3	Optional	The third line of the address.
City	Always	The name of the city that is associated with the address.
State	Optional	The state or province code for the address.
Country ID	Always	The country code for the address.
Post	Optional	The postal code for the address.
Contact Name	Optional	The name of the contact for the partner at this address.
Contact Phone	Optional	The phone number of the contact person at this address.

Table 2-90 (Cont.) Address Detail

Message Element	Required?	Notes
Contact Telex	Optional	The telex number of the partner representative contact.
Contact Fax	Optional	The fax number of the contact person at this address.
Contact Email	Optional	The email address of the partner representative contact.
Oracle Vendor Site ID	Optional	The unique identifier of this address in the Oracle Financials systems, if used.
County	Optional	The county name that is associated with the address.
Jurisdiction Code	Optional	ID associated to the tax jurisdiction of the country-state relationship.
Flex Attributes	Optional	Child Node

Partner Header Modification

Updating an existing partner will trigger header modification message to notify the integrating systems. This message will contain the header level information, the primary address and flex attributes of the partner that changed.

Partner Detail Creation

Adding new addresses for an existing partner will trigger a partner detail creation message. This message is sent along with the header information.

Partner Detail Modification

Any updates made to the partner's address information will trigger a partner detail modification message. This message is sent along with the partner header information.

Deleting Partners

Deleting an existing partner will trigger a partner delete message to notify the integrating systems. This message will contain the partner type and partner ID that was deleted.

Deleting Partner Details

Partner addresses can be deleted only if they are not mandatory or if they are mandatory but are not marked as the primary address for that address type. The deletion will trigger a partner detail delete message. The message will contain the partner type, partner ID and the address key of the address that was deleted.

Table 2-91 Partner Delete

Message Element	Required?	Notes
Partner Type	Always	Specifies the type of the partner.
Partner ID	Always	Unique number that, with the partner type, identifies the partner being deleted.

Table 2-91 (Cont.) Partner Delete

Message Element	Required?	Notes
Address Detail	Optional	Child Node

Table 2-92 Address Detail Delete

Message Element	Required?	Notes
Address Key	Always	The unique identifier of the address being deleted.

Flex Attributes

If any custom flex attributes (CFAS) for the partner or partner address have been added or modified, it will trigger an update message. The node of the integration that supports this will contain the name of the attribute as it is defined in the group set level view, the value of the custom attribute. If it is a date attribute, the date value is in a separate field. Flex attributes can only be added to or updated; they cannot be deleted.

Table 2-93 Flex Attributes

Message Element	Required?	Notes
Name	Always	Holds the attribute name.
Value	Optional	Holds the value of the attribute for number and character type attributes
Value Date	Optional	Holds the date for date type attributes.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
partnercre	Partner Create	PartnerDesc.xsd
partnermod	Partner Modify	PartnerDesc.xsd
partnerdel	Partner Delete	PartnerRef.xsd
partnerdtlcre	Partner Detail Create	PartnerDesc.xsd
partnerdtlmod	Partner Detail Modify	PartnerDesc.xsd
Partnerdtldel	Partner Detail Delete	PartnerRef.xsd

Receiver Unit Adjustment Publication API

This section describes the receiver unit adjustment publication API.

Functional Area

Receipts

Business Overview

When mistakes are made during the receiving process at the store or warehouse, receiver unit adjustments (RUAs) are made to correct the mistake. Merchandising publishes messages about receiver unit adjustments in cases where RUAs are initiated through Invoice Matching or Merchandising, a message is published to store and warehouse inventory systems, like Oracle Retail Store Inventory and Operations Cloud Service (SIOCS), to keep inventory in synch.

Note

Oracle WMS Cloud and Oracle Retail Warehouse Management (RWMS) do not subscribe to Receiver Unit Adjustment messages from Merchandising.

Creating Receiver Unit Adjustments

When a receiver unit adjustment is created, it triggers a message sent to notify external systems. The full details are sent for the receiver unit adjustment as part of the create message, as shown below.

Table 2-94 Receiver Unit Adjustment Header

Message Element	Required?	Notes
Location	Always	The location (store or physical warehouse) impacted by the receiver unit adjustment.
Location Type	Always	This is the type of location in the location field. Valid values are Store (S) or Warehouse (W).

Table 2-95 Receiver Unit Adjustment Detail

Message Element	Required?	Notes
Order Number	Always	This contains the order number against which a receiver unit adjustment was performed
ASN Number	Optional	This contains the ASN number associated with the shipment for which a receiver unit adjustment was performed.
Item Number	Always	This contains item whose receipt quantity was adjusted.
Carton ID	Optional	This contains the carton that contained the item against which the receiver unit adjustment was performed, if applicable.

Table 2-95 (Cont.) Receiver Unit Adjustment Detail

Message Element	Required?	Notes
Adjustment Quantity	Always	This contains the quantity, either positive or negative, for the adjustment.
From Disposition	Optional	If the adjustment quantity is less than zero, then this will contain ATS. If it is a positive adjustment, then this will be NULL.
To disposition	Optional	If the adjustment quantity is greater than zero, then this will contain ATS. Otherwise, this will be NULL.
Universal identification number (UIN)	optional	Not used by Merchandising.
Status of UIN	optional	Not used by Merchandising.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This will shut down the RIB adapter, preventing any further messages from being processed until this error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
rcvunitadjcre	Receive unit adjust Create	RcvUnitAdjDesc.xsd
rcvunitadjcre	Receive unit adjust Create Detail	RcvUnitAdjDtl.xsd

RTV Request Publication API

This section describes the RTV request publication API.

Functional Area

Inventory Movement

Business Overview

A return to vendor (RTV) order is used to send merchandise back to the supplier. The RTV message is published by Merchandising to the store or warehouse to provide visibility to the corporately created RTV. Consequently, when the store or warehouse ships the RTV, it communicates the original RTV order number back to Merchandising to update the original RTV record.

Create RTV Request

Approval of an RTV order in Merchandising results in this message being published. The RTV Request message includes the following:

Table 2-96 RTV Request Header

Message Element	Included?	Notes
RTV Order No	Always	The number that uniquely identifies the return to vendor within the system.
Supplier	Always	The supplier site to which the merchandise is being returned.
Status Indicator	Always	The status of the return. Valid values include: 10 - Approved and 20 - Cancelled.
Location	Always	The store or virtual warehouse number from which the return will occur.
Location Type	Always	The type of location returning the items. Valid values are store (S) or warehouse (W).
Physical Location	Always	The store number or the physical warehouse from which the items are being returned.
Total Order Amount	Optional	The total amount of the return in the supplier's currency.
Ship To Address Line 1	Always	The first line of the supplier's address for returns.
Ship To Address Line 2	Optional	The second line of the supplier's address for returns.
Ship To Address Line 3	Optional	The third line of the supplier's address for returns.
Ship To City	Always	The city name of the supplier's address for returns.
State	Optional	The state code of the supplier's address for returns.
Ship To Country Code	Always	The country ID of the supplier's address for returns.
Ship To Postal Code	Optional	The postal code of the supplier's address for returns.
Return Authorization Number	Optional	The number that the supplier provides when the decision is made that an order may be returned, if required.
Return Courier	Optional	The courier's name if the supplier requires that returns be sent by a designated carrier.
Freight Charge	Optional	The freight cost associated with the vendor return. This field is stored in the supplier's currency.
Creation Date	Always	The date the vendor return was created.
Completed Date	Optional	The date the vendor return was completed (for updates only).
Handling Percentage	Optional	The handling (restocking) percent charged to the retailer for returns, if specified.
Handling Cost	Always	The handling (restocking) cost for the return if specified. This is specified in the supplier's currency.
External Reference No	Optional	Reference number used in an external system; only applicable if the RTV was initiated in the store or warehouse (for updates only).
Comments	Optional	Comments associated with the return.
RTV Request Details	Optional	Child Node

Table 2-96 (Cont.) RTV Request Header

Message Element	Included?	Notes
Not After Date	Optional	The last date that the return can be shipped to the vendor.
Flex Attributes	Optional	Child Node.
Supplier Name	Always	The supplier site's name in the system's primary language.

Table 2-97 RTV Request Detail

Message Element	Included?	Notes
Sequence Number	Always	Unique number identifying the RTV Detail record.
Item	Always	Unique identifier for the item.
Shipment	Optional	Not used.
Inventory Status	Optional	The inventory status code. Only applicable if the reason field has a value of 'U' for unavailable inventory.
RTV Quantity	Always	The quantity of items requested to be returned to the supplier.
Unit Cost	Always	The cost per unit for the SKU being returned in the supplier's currency.
Reason	Always	The reason for the return. Valid values are found in code type RTVR.
From Disposition	Optional	This contains the status of the inventory that needs to be returned to the vendor. If there are multiple inventory status codes mapped to an inventory status, the values will be comma delimited. This can be comma delimited in case item is returned from multiple unavailable buckets

Update RTV Request

Updating an existing RTV request will trigger a header modification message to notify the integrating systems. After approval, limited updates can be made to an RTV that would result in a modification being published, such as updating comments or cancelling the RTV. This message will contain only the details that changed for the RTV.

Delete RTV Request

When RTV information is deleted for a closed RTV based on the history retention days defined in Merchandising's system options, the delete will trigger an RTV Request detail delete message. A header-level message is not published. CFAS attributes, if associated with the RTV, are also not included in the delete publication. It is assumed the subscribing system would delete the header and CFAS attributes with the details.

The RTV Request delete message includes the following:

Table 2-98 RTV Request Delete

Message Element	Included?	Notes
RTV Order No	Always	Specifies the RTV Order number of the detail record that is being deleted.
RTV Request Detail	Optional	Child node
Location	Always	The store or warehouse number of the RTV detail record that is being deleted.
Location Type	Always	The location type of the RTV detail record that is being deleted.

Table 2-99 RTV Request Detail Delete

Message Element	Included?	Notes
Sequence Number	Always	The sequence number of the RTV detail record that is being deleted.
Item	Always	The item number of the RTV detail record that is being deleted.

Flex Attributes

If any custom flex attributes (CFAS) for the RTV Request header have been added or modified, it will trigger a header modify message. All the entity's active flex attributes from all attribute groups are published as key–value pairs based on the group-set view. This CFAS object is embedded in the outbound RTV Request message.

Table 2-100 Flex Attributes Detail

Message Element	Included?	Notes
Name	Always	The flex attribute defined by the business.
Value	Optional	The value of the flex attribute defined by the business.
Value Date	Optional	The date value of the flex attribute if flex attribute is defined as a date.

RTV Request Full Publish

When publishing an RTV Request header modification or an RTV Request detail create or modification or delete message, a second full replacement message will be published if the Publish RIB Objects system option is configured to be Deltas and Full (Y). This message will contain the header-level information, flex attributes, and the details of the RTV request. Based on the message type, RIB will route the full replacement message to appropriate applications. Currently this is used only by WMS Cloud.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of

Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
RtvReqCre	RTV Request Create Message	RTVReqDesc.xsd
RtvReqMod	RTV Request Modify Message	RTVReqDesc.xsd
RtvReqDtlDel	RTV Request Detail Delete Message	RTVReqRef.xsd
RtvReqFulRep	RTV Request Full Replacement Message	RTVReqDesc.xsd

Season and Phase Publication API

This section describes the season phase publicatoin API.

Functional Area

Foundation Data

Business Overview

Merchandising publishes details about new and updated seasons and phases in order that external systems that use this information can be informed of the updates. Updates are provided synchronously in a near-real time manner.

New Seasons

Creating a new season triggers a message to be sent to notify external systems. The full details are sent for the new season as part of the create message, the season ID, description, start date and end date.

Table 2-101 Seasons Header

Message Element	Included?	Notes
Season ID	Always	This field contains the season identifier.
Season Description	Always	This field contains the description associated with the season. This is populated when creating and updating seasons.
Start Date	Always	This field contains the starting date for the season. This is populated when creating and updating seasons.
End Date	Always	This field contains the end date for the season. This is populated when creating and updating seasons.
Phase Detail	Optional	Child Node – only included when adding/updating phases

Updated Seasons

When an existing season is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields in the message. See above for the format of an update message.

New Phases

Creating a new phase triggers a message to be sent to notify external systems. The full details are sent for the phase as part of the create message, including the ID, description, start and end date for the phase, the season in which the phase belongs.

Table 2-102 Phases Detail

Message Element	Included?	Notes
Phase ID	Always	This field contains the phase identifier.
Phase Description	Always	This field holds the description of the phase.
Start Date	Always	This field contains the starting date for the phase.
End Date	Always	This field contains the end date for the phase.

Updated Phases

When an existing phase is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields in the message.

Deleted Seasons

When an existing season is deleted, this will also trigger a delete transaction to be sent to notify external systems that this season is no longer valid. The delete message will include only the ID of the season being deleted.

Table 2-103 Seasons Header

Message Element	Included?	Notes
Season ID	Always	This field contains the season identifier.
Phases Delete Detail	Optional	Child Node – only included when deleting phases

Deleted Phases

When an existing phase is deleted, this will also trigger a delete transaction to be sent to notify external systems that this phase is no longer valid. The delete message will include the ID of the phase being deleted and the ID of the season in which the phase belongs. See above for the full structure of a delete phase message.

Table 2-104 Phases Delete Detail

Message Element	Included?	Notes
Phase ID	Always	This field contains the phase identifier.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
seasoncre	Season Create	SeasonDesc.xsd
seasondtlcre	Phase Create	SeasonDesc.xsd
seasonmod	Season Modify	SeasonDesc.xsd
seasondtlmod	Phase Modify	SeasonDesc.xsd
seasondel	Season Delete	SeasonRef.xsd
seasondtldel	Phase Delete	SeasonRef.xsd

Store Publication API

This section describes the store publication API.

Functional Area

Organizational Hierarchy

Business Overview

This API publishes store and store information through RIB to external systems. Store information is published when new stores are created, updates are made to existing stores, or existing stores are deleted. Only the primary addresses and mandatory address types are published through this message, as it is assumed that integration subsystems only require one address.

Create Stores

Stores can be set up in Merchandising through the application user interface or through subscription of store information from external systems. Both methods result in this message being published when a store is created. The store creation message is published from Merchandising only after the store has been completely set up, including the mandatory addresses. Once all of these criteria are met the store details are published.

Modify Stores

Updating an existing store will trigger a header modification message to notify the integrating systems. The message will contain only header level information that was changed.

Create Store Addresses

New primary addresses for the mandatory types are sent in this message for a new or an existing store. This message is sent without the store header information for existing stores. The `key_value_1` field on this message will map to the store ID.

Modify Store Addresses

Any updates made to a store's primary address information for the mandatory types will trigger an address modification message. This message is sent without the store header information if no store level changes are made at the same time. The `key_value_1` field on this message will map to a store ID. When details changes are sent, only the values that changed are included in the message.

Create Store Hours

Store hours can be set up for all 7 days in a week for an existing store or when the store is newly created. This message is also sent along with header information.

Modify Store Hours

Any updates to the store hours are specified in this message. This message is also sent with header information.

Table 2-105 Store Create and Update

Message Element	Included?	Notes
Store	Always	This field contains the number which uniquely identifies the store.
Store Type	Always	This field indicate whether a particular store is a franchise or company store.
Store Name	Always	This field contains the name of the store which, along with the store number, identifies the store.
Store Name10	Always	This field contains a ten-character abbreviation of the store name.
Store Name3	Always	This field contains a three-character abbreviation of the store name.
Store Add1	Optional	This field contains the first line of the address of primary address type.
Store Add2	Optional	This field contains the second line of the address of primary address type.
Store City	Optional	This field contains the name of the city that is associated with the address of primary address type.

Table 2-105 (Cont.) Store Create and Update

Message Element	Included?	Notes
County	Optional	This field contains county name for the location associated with the address of primary address type.
State	Optional	This field contains the state abbreviation for the address of primary address type.
Country Id	Optional	This column contains the country code where the address exists of primary address type.
Store Pin code	Optional	This field contains the zip code for the address of primary address type.
Store Class	Always	This field contains the code letter indicating the class of which the store is a member. Valid values are A through E.
Store Manager Name	Always	This field contains the name of the store manager.
Store Open Date	Always	This field contains the date on which the store opened.
Store Close Date	Optional	This field contains the date on which the store closed.
Acquired Date	Optional	This field contains the date on which the store was acquired.
Remodel Date	Optional	This field contains the date on which the store was last remodeled.
Fax Number	Optional	This field contains the fax number for the store.
Phone Number	Optional	This field contains the phone number for the store.
Email	Optional	This field the email address for the location.
Total Square Feet	Optional	This field contains the total square footage of the store.
Selling Square Feet	Optional	This field contains the total square footage of the stores selling area.
Linear Distance	Optional	This field contains the total merchandisable space of the location.
Stockholding Indicator	Always	This field indicates whether the store can hold stock.
Channel Id	Optional	This field contains the channel with which the store is associated.
Store Format	Optional	This field contains the number indicating the format of the store. Valid values are found on the store format table.
Mall Name	Optional	This field contains the name of the mall in which the store is located.
District	Always	This field contains the number of the district in which the store is a member.
District Name	Optional	This field contains the name of the district in which the store is a member.
Promo Zone	Optional	Not used.
Promo Description	Optional	Not used.

Table 2-105 (Cont.) Store Create and Update

Message Element	Included?	Notes
Transfer Zone	Optional	This field contains the transfer zone ID in which the store is located.
Description	Optional	This field contains the transfer zone description in which the store is located.
Default WH	Optional	This field contains the number of the warehouse that is the primary sourcing warehouse for the store. This value will always be a virtual warehouse.
Stop Order Days	Optional	This field contains the number of days before a store closing that the store will stop accepting orders. This column will be used when the store close date is defined.
Start Order Days	Always	This field contains the number of days before the store open date that the store will begin accepting orders.
Currency Code	Always	This field contains the currency code under which the store operates.
Language	Always	This field contains the language code to be used for the given store.
Integrated POS	Always	This field indicates whether Sales Audit should expect files from this store for processing. Valid values are Yes (Y) or No (N).
Original Currency Code	Always	Not used.
DUNS Number	Optional	This field contains the Dun and Bradstreet number to identify the store.
DUNS Location	Optional	This field contains the Dun and Bradstreet number to identify the location.
Address Details	Always	Child Node
Pricing Location	Optional	This field contains the location from which pricing records will be copied for a new store or will be used to determine the price for this store if not otherwise defined for an item.
Pricing Location Currency	Optional	This field contains the currency code of the location from which pricing records will be copied.
Org Unit	Optional	This field contains the organizational unit ID that this store is associated with.
Time Zone Name	Always	This field contains the text value of the time zone of the store.
Flex Attributes	Optional	Child Node
Store Name Secondary	Optional	This field contains the secondary name of the store.
Store Class Description	Optional	This field contains the description of the class of which the store is a member.
VAT Region	Optional	This field contains the ID of the tax region the store is associated with.
VAT Include Indicator	Optional	This field contains whether tax will be included in the retail prices for the store. Valid values are Yes (Y) or No (N).

Table 2-105 (Cont.) Store Create and Update

Message Element	Included?	Notes
Channel Name	Optional	This field contains the name of the channel.
Store Format Name	Optional	This field contains the name of the Store Format.
Sister Store	Optional	This field contains store number which will be used to relate the current store to the historical data of an existing store. This is used by Allocation.
Transfer Entity	Optional	This field contains the transfer entity the store is associated with.
Auto Receive	Always	This field contains whether receipts for the store will have receipts made automatically by Merchandising. Valid Values are Y (Yes), N (No), D (System Default). Default value should be D, which means it is driven by the system option definition.
Re Merch Indicator	Always	Not used.
Franchise Customer	Optional	Indicates which franchise customer this store is associated with.
Customer Order Loc Indicator	Optional	This field indicates whether the location is customer order location. If the indicator is Y then the location can be used for sourcing and fulfillment orders; else it cannot be used and inventory information will not be shared with OMS.
Gift Wrapping Indicator	Optional	This field indicates if the location will support gift wrapping.
Customer Order Ship Indicator	Optional	This field indicates if the location will support shipping to a customer for customer orders.
Store Hours Details	Always	Child Node
Online Store Indicator	Optional	This field indicates that the store is an online store.

Table 2-106 Address Create and Update

Message Element	Included?	Notes
City Id	Optional	Not used.
State Name	Optional	This field contains the state abbreviation for the address.
Country Name	Always	This field contains the country where the address exists.
Address	Always	This field contains a unique number used to distinguish between different addresses.
Address type	Always	This field indicates the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance
Primary Address Type	Always	This field indicates if this address type is the primary address type for the store. Valid values are Yes (Y) or No (N).
Primary Address Indicator	Always	This field indicates whether the address is the primary address for the address type.

Table 2-106 (Cont.) Address Create and Update

Message Element	Included?	Notes
Address Line 1	Optional	This field contains the first line of the address.
Address Line 2	Optional	This field contains the second line of the address.
Address Line 3	Optional	This field contains the third line of the address.
City	Always	This field contains the name of the city that is associated with the address.
State	Optional	This field contains the state abbreviation for the address.
Country	Always	This field contains the country code where the address exists.
Post	Optional	This field contains the postal code for the address.
Contact Name	Optional	This field contains the name of the contact for the supplier at this address.
Contact Phone	Optional	This column contains the phone number of the contact person at this address.
Contact Telex	Optional	This field contains the telex number of the store representative contact.
Contact Fax	Optional	This field contains the fax number of the contact person at this address.
Contact Email	Optional	This field contains the email address of the store representative contact.
Oracle Vendor Site Id	Optional	This field contains the Oracle Vendor Site Id.
County	Optional	This field contains the county name for the location.
Jurisdiction Code	Optional	This field contains the jurisdiction code for the country-state relationship.

Table 2-107 Store Hours Create or Update

Message Element	Included?	Notes
Store	Always	This field contains the store number.
Day Number	Always	This field contains the numeric value that indicates the day that the open and close times pertain to. Valid values are 1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, or 7-Saturday.
Store Open Time	Optional	This column contains the open time for the store for each day of the week. The time format will be in HH:MM AM or HH:MM PM.
Store Close Time	Optional	This column contains the close time for the store for each day of the week. The time format will be in HH:MM AM or HH:MM PM.

Flex Attributes

If any custom flex attributes (CFAS) for the store have been added or modified, it will trigger a header modify message. All the entity's active flex attributes from all attribute groups are

published as key–value pairs based on the group set view. This CFAS object is embedded in the outbound store message.

Table 2-108 Flex Attribute

Message Element	Included?	Notes
Name	Always	The flex attribute defined by the business.
Value	Optional	The value of the flex attribute defined by the business
Value Date	Optional	The date value of the flex attribute if flex attribute is defined as a date.

Delete Stores

When a store is deleted that has already published to external systems, a delete transaction will be sent to notify external systems that the store is no longer valid. If an address is deleted for a store, this also triggers a delete transaction. The message includes the store information along with the address key. There is not a separate delete for store hours – it is assumed that external systems will delete the store hours when they receive the store delete message.

Table 2-109 Store Delete

Message Element	Included?	Notes
Store	Always	This field contains the number which uniquely identifies the store.
Store Type	Always	This field indicate whether a particular store is a franchise or company store.
Stock Holding Indicator	Always	This field indicates whether the store can hold stock.
Address Detail	Optional	Child Node

Table 2-110 Address Delete

Message Element	Included?	Notes
Address	Always	This field contains the address key.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This will shut down the RIB adapter, thus preventing any further messages from being processed until this is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
Storecre	Store Create	StoreDesc.xsd
Storedtlcre	Store Address Create	StoreDesc.xsd
Storehrcre	Store Hours Create	StoreDesc.xsd
Storemod	Store Header Modify	StoreDesc.xsd
Storedtlmod	Store Address Modify	StoreDesc.xsd
Storehrmod	Store Hours Modify	StoreDesc.xsd
Storedel	Store Delete	StoreRef.xsd
Storedtldel	Store Address Delete	StoreRef.xsd

Transfers Publication API

This section describes the transfers publication API.

Functional Area

Transfer

Business Overview

A transfer is a movement of stock on hand from one stockholding location within the company to another. Transfers consist of header-level information in which source and destination locations are specified, and detail information regarding what items and the quantity of each item is to be transferred. In some cases, transfers can also be created to pass through a finisher location to have activities performed on the items being transferred, such as ticketing, re-packing, or item transformations. Transfers that have been approved are published and will be visible to external systems. Updates are provided synchronously in a near-real-time manner.

New Transfer

Creating and approving a new transfer triggers a message to be sent to notify external systems of the changes. The full details in the message are sent for the new transfer as part of the create message. It includes the details such as the source and destination location, the items that are to be transferred, component ticketing information (if applicable), and the quantity. If the transfer is for a customer order (tsf_type = 'C0'), customer-related information pulled from the ORDCUST table is included in the header message and published. A transfer with header record will have at least one detail before it can be approved.

Transfer Details

Adding transfer details like item and transfer quantity to an existing transfer triggers a detail create message to be created and published after the transfer is approved. This action creates one header modify message for the status change and each detail addition will create one detail create message. The detail create message still contains both complete header and detail information. The details of the payload used for transfer is as below.

Updated Transfer

Any updates to an existing transfer trigger a message to be sent to notify external systems based on updates made at the transfer header level and/or the detail level. The message that will be published will either contain the header information that changed in the message for header updates such as date changes, just the detail level information for detail level changes such as quantity changes, or addition of new items, or both.

Header Only Updates

When updates are made at the transfer-header level such as updating dates, comment description, or changing the status of the transfer, only header information is published. Information specific to items and locations is not included. Changes in header information, unapproving a transfer, or any action that may change the status of a pre-approved transfer, such as closing or reinstating a transfer trigger header level publication. Creating new and updating header level custom flex attributes (CFAS) also triggers header-level update publishing.

Detail Only Updates

When updates are made at the transfer-detail level without changes to the header information, only detail information is published. The detail-level update includes the change in quantity transferred. When a transfer has already been approved, no deletes can be made at the detail level.

Full Message Updates

In cases where the solution receiving transfers cannot support just receiving the delta changes, another option is provided that can resend the full transfer header and transfer details, along with any flex attributes configured for transfers, whenever there is a change. This message payload will contain a full snapshot of the updated transfer in cases where the system option Publish Full Objects (PUB_FULL_OBJECTS_IND) is set to Deltas and Full (Y). This is used for Oracle Retail Warehouse Management (RWMS) Cloud integration.

Create or Update Transfer Header

Message Element	Required?	Notes
Transfer Number	Always	This field contains the number that uniquely identifies the transfer.
Document Type	Always	This field contains the document type. For transfers it is defaulted to T.
Physical From Location	Always	This field contains the physical location that the source location belongs to. For stores or external finishers, it will be the store ID, for a virtual warehouse or internal finisher, it will be the physical warehouse ID.
From Location Type	Always	Type of location in the source location field. Valid values are store (S), virtual warehouse (which could be an internal finisher) (W), or external finisher (E).
From Store Type	Optional	If the from location is a store, this field contains the type of store. Valid values are company store (C), franchise store (F).

Message Element	Required?	Notes
From Stockholding Indicator	Optional	If the from location is a store, this field indicates if it is stockholding. Valid values are yes (Y) or no (N).
From Location	Always	This field contains the source location - either a store, external finisher, or virtual warehouse number (which could be an internal finisher) based on the from location type.
Physical To Location	Always	This field contains the physical location ID for the destination location. For stores or external finishers, it will be the store ID, for a virtual warehouse or internal finisher, it will be the physical warehouse ID.
To Location Type	Always	This field contains the type of location the transfer will be shipped to. Valid values are store (S), warehouse (which could be an internal finisher) (W), or external finisher (E).
To Store Type	Optional	If the to location is a store, this field contains the type of store. Valid values are company store (C), franchise store (F).
To Stockholding Indicator	Optional	If the to location is a store, this field indicates if it is stockholding. Valid values are yes (Y) or no (N).
To Location	Always	This field contains the destination location - either a store, external finisher, or virtual warehouse number (which could be an internal finisher) based on the to location type.
Transfer Type	Always	This field holds the code for the type of the transfer. Valid values are: • SR - Store Requisition • CO - Customer Order • RV - RTV • CF - Confirmation • NS - Non-Salable • AD - Administrative • MR - Manual Requisitions • PL - PO-Linked Transfer. • IC -Intercompany
Pick Not Before Date	Always	This field holds the earliest date on which the transfer may be picked. This field is populated with the date the transfer was approved.
Pick Not After Date	Optional	The last date on which the order may be picked. This is calculated as transfer approval date + days from code_detail where code_type = 'DEFT' and code = 'DATE'
Order Type	Always	This is the order type associated with transfers. If the transfer type is customer order, then the order type is set to MANUAL, else it is defaulted from the default order type set in the Merchandising.
Priority	Optional	Not used in Merchandising.
Break By Distro	Always	This field holds the delivery type for the customer order. If it is shipped direct to the customer, this will be yes (Y), otherwise N (no).

Message Element	Required?	Notes
Delivery Date	Optional	This field holds the earliest date that the transfer can be delivered to the store.
Customer Name	Optional	This field contains the name of the customer.
Delivery Address 1	Optional	This column contains the first line of the delivery address of the customer for customer order transfers. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Delivery Address 2	Optional	This column contains the second line of the delivery address of the customer, for customer order transfers. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Delivery City	Optional	This column contains the city of the delivery address, for customer order transfers. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Delivery State	Optional	This column contains the state code of the delivery address, for customer order transfers. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Delivery Postal	Optional	This column contains the postal code of the delivery address, for customer order transfers. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Delivery Country	Optional	This column contains the country code of the delivery address, for customer order transfers. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Message	Optional	This column contains any comments associated with the transfer.
Parent Transfer Number	Optional	This column contains the transfer number for the first leg (from location to finisher) for transfers with finishing.
Expected WH Date	Optional	This column contains the date the inventory is expected to arrive at the warehouse.
Approval ID	Optional	This column contains the ID of the user who approved the transfer.
Approval Date	Optional	This column contains the date the transfer was approved
From Location Transfer Entity	Optional	This column contains the transfer entity of the source location.
To Location Transfer Entity	Optional	This column contains the transfer entity of the destination location.

Message Element	Required?	Notes
Inventory Type	Optional	This column will indicate whether the transfer is for available inventory or unavailable inventory. Valid Values are: - A: Available - U: Unavailable
Transfer Status	Optional	This field indicates the status of the transfer. Valid values are: - I - Input - B - Submitted - A - Approved - P - Picked - S - Shipped - N - In Progress - C - Closed - D - Deleted
Not After Date	Optional	This field contains the last day delivery of the transfer will be accepted.
Context Type	Optional	This field indicate the transfer reason or category. Valid values are found in code type CNTX. Note: context type Repairing (REPAIR) has specific logic tied to it and must be associated with a return to warehouse or a transfer using an external finisher.
Context Value	Optional	This field holds value relating to the context type, for example Promotion Number for context type Promotion.
Delivery Slot ID	Optional	This field indicates the delivery slot that will be used for the transfer. The valid values are defined in the DELIVERY_SLOT table.
Delivery Slot Description	Optional	This field contains the delivery slot description.
Customer Order Number	Optional	This field contains the customer order number associated with the transfer.
Fulfill Order Number	Optional	This field contains the fulfillment order number for the transfer.
Carrier Code	Optional	This field indicates the carrier the customer order is to be shipped with, if specified on the order.
Carrier Service Code	Optional	This field indicates the code for the service level for the carrier (e.g., overnight shipping) for a customer order shipment. Valid values are found in code type CSVC.
Consumer Delivery Date	Optional	This field indicates the desired date the delivery is required by the customer.
Consumer Delivery Time	Optional	This field indicates the desired time, the delivery is required by the customer.
Deliver First Name	Optional	This column contains the first name for the delivery address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).

Message Element	Required?	Notes
Deliver Phonetic First	Optional	This column contains the phonetic first name for the delivery address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver Last Name	Optional	This column contains the phonetic last name for the delivery address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver Phonetic Last	Optional	This column contains the phonetic last name for the delivery address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver Preferred Name	Optional	This column contains the preferred name for the delivery address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver Company Name	Optional	This column contains the company name for the delivery address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver Address 3	Optional	This column contains the third line of the delivery address of the customer, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver County	Optional	This column contains the county of the delivery address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Deliver Phone	Optional	This column contains the delivery phone number, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill First Name	Optional	This column contains the first name for the billing address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Phonetic First	Optional	This column contains the phonetic first name for the billing address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Last Name	Optional	This column contains the last name for the billing address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).

Message Element	Required?	Notes
Bill Phonetic Last	Optional	This column contains the phonetic last name for the billing address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Preferred Name	Optional	This column contains the preferred name for the billing address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Company Name	Optional	This column contains the company name for the billing address on the order, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Address 1	Optional	This column contains the first line of the billing address of the customer, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Address 2	Optional	This column contains the second line of the billing address of the customer, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Address 3	Optional	This column contains the third line of the billing address of the customer, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill County	Optional	This column contains the county of the billing address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill City	Optional	This column contains the city of the billing address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Country	Optional	This column contains the country code of the billing address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill Post	Optional	This column contains the postal code of the billing address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Bill State	Optional	This column contains the state code of the billing address, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).

Message Element	Required?	Notes
Bill Phone	Optional	This column contains the billing phone number, for a customer order transfer. This field will be set to NULL if the Retain Customer Information system option is unchecked (N).
Partial Delivery Indicator	Optional	This column indicates if a customer order transfer can be picked and shipped partially or if it should be shipped only when complete. Valid values are Y or N.
Consumer Direct	Optional	This column indicates the order as being consumer direct (Y) or not (N). An order is consumer direct if it is a customer order.

Create or Update Transfer Detail

Message Element	Required?	Notes
Item	Always	This field contains unique alphanumeric value that identifies the item.
Transfer Quantity	Always	This field contains the total quantity of the item reserved at the from location for this transfer. For customer order transfers the values are fetched from the customer order tables.
Price	Optional	This field contains the unit retail price in the selling unit of measure for the item/ to location combination
Selling UOM	Optional	This field contains the selling unit of measure of the item.
Priority	Optional	Not used in Merchandising.
Expedite Flag	Optional	This field will be set to yes (Y) if the freight type is Expedite else it is set to no (N).
Store Order Multiple	Optional	This field contains the multiple in which the item needs to be shipped from a warehouse to the location.
Transfer PO Link Number	Optional	If this transfer was generated out of the replenishment process using the WH/Cross Link stock category, then this field contains a reference number to link the item on the transfer to the purchase order that will fulfill part of the need.
Ticket Type	Optional	This field contains the ticket type ID associated with the item.
Inventory Status	Optional	This field contains the inventory status that should be used as the source of inventory on the transfer. If the transfer is from available inventory this field will contain -1.
Transaction UOM	Optional	This field contains the original transaction unit of measure. It is only used for customer direct orders coming from an order management system.
Item Line Number	Optional	This field contains the detail item line number for customer order transfers.
Comments	Optional	This field contains any comments associated with the transfer line item for a customer order transfer.

Pack Component Detail

Message Element	Required?	Notes
Component Item	Conditional	This field contains an alphanumeric value that identifies the item within the pack.
Component item Price	Conditional	This field contains the unit retail price in the selling unit of measure for the component item/location combination
Component Selling UOM	Conditional	This field contains the selling unit of measure for the component item.

Flex Attributes

If any custom flex attributes (CFAS) for the transfer have been added or modified, it will trigger a transfer header message, as described above. The node of the integration that supports this contains the name of the attribute as it is defined in the group set level view, the value of the custom attribute. If it is a date attribute, the date value is in a separate field. You can define group set views at the header level (TSFHEAD) levels. Flex attributes can only be added to or updated on a transfer; they cannot be deleted.

Table 2-111 Custom Flex Attribute Detail

Message Element	Required?	Notes
Name	Always	The flex attribute defined by the business.
Value	Optional	The value of the flex attribute defined by the business
Value Date	Optional	The date value of the flex attribute if flex attribute is defined as a date.

Deleted Transfer

A transfer can only be deleted when it is still in Approved status or when it has been Closed. Only a detail that has not been shipped may be deleted, and it cannot be deleted if it is currently being worked on by an external system. Deleting a transfer removes it from the system. External systems are notified by a published delete message that contains the number of the transfer to be deleted.

Table 2-112 Header Information

Message Element	Required?	Notes
Transfer Number	Always	This field contains the number that uniquely identifies the transfer.
Documentation Type	Always	This field contains the document type. For transfers it is defaulted to T.
Physical From Location	Optional	This field contains the physical location that the source location belongs to. For stores or external finishers, it will be the store ID, for a virtual warehouse or internal finisher, it will be the physical warehouse ID.

Table 2-112 (Cont.) Header Information

Message Element	Required?	Notes
From Location	Optional	This field contains the source location - either a store, external finisher, or virtual warehouse number (which could be an internal finisher) based on the from location type.
From Location Type	Optional	Type of location in the source location field. Valid values are store (S), virtual warehouse (which could be an internal finisher) (W), or external finisher (E).
From Store Type	Optional	If the from location is a store, this field contains the type of store. Valid values are company store (C), franchise store (F).
From Stockholding Indicator	Optional	If the from location is a store, this field indicates if it is stockholding. Valid values are yes (Y) or no (N).
Physical To Location	Always	This field contains the physical location ID for the destination location. For stores or external finishers, it will be the store ID, for a virtual warehouse or internal finisher, it will be the physical warehouse ID.
To Location	Always	This field contains the destination location - either a store, external finisher, or virtual warehouse number (which could be an internal finisher) based on the to location type.
To Location Type	Optional	This field contains the type of location the transfer will be shipped to. Valid values are store (S), warehouse (which could be an internal finisher) (W), or external finisher (E).
To Store Type	Optional	If the to location is a store, this field contains the type of store. Valid values are company store (C), franchise store (F).
To Stockholding Indicator	Optional	If the to location is a store, this field indicates if it is stockholding. Valid values are yes (Y) or no (N).
Transfer Parent Number	Optional	This column contains the transfer number for the first leg (from location to finisher) of a transfer with finishing.

Table 2-113 Detail Information

Message Element	Required?	Notes
Item	Always	This field contains unique alphanumeric value that identifies the item being deleted from the transfer.

Transfer Unapprove

When the transfer is moved back to input status from approved, the RIB processes the transfer as a delete message when publishing it to external systems. When the transfer is re-approved, the transfer is processed as a new transfer for publishing.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
TransferCre	Transfer Create Message	TsfDesc.xsd
TransferDtlCre	Transfer Detail Create Message	TsfDesc.xsd
TransferHdrMod	Transfer Modify Message	TsfDesc.xsd
TransferDtlMod	Transfer Detail Modify Message	TsfDesc.xsd
TransferDel	Transfer Delete Message	TsfRef.xsd
TransferDtlDel	Transfer Detail Delete Message	TsfRef.xsd
Transferfulrep	Transfer Full Replacement Message	TsfDesc.xsd
TransferUnapp	Transfer Unapprove	TsfRef.xsd

UDA Publication API

This section describes the UDA publication API.

Functional Area

Foundation Data

Business Overview

Merchandising publishes details about user-defined attributes (UDAs) to the Oracle Retail Integration Bus (RIB). UDAs provide a method for defining attributes and associating the attributes with specific items, items on an item list, or items in a specific department, class, or subclass. UDAs are useful for information and reporting purposes. Unlike traits or indicators, UDAs are not interfaced with external systems. UDAs do not have any programming logic associated with them. UDA messages are specific to basic UDA identifiers and values defined in Merchandising. The UDAs can be displayed in one or more of three formats: Dates, Freeform Text, or a List of Values (LOV).

New UDAs

Creating a new UDAs triggers a message to be sent via the RIB to notify external systems. The full details are sent for the new UDA as part of the create message: the UDA ID, description, display type, data type, data length and single value indicator.

Table 2-114 UDA

Message Element	Included?	Notes
UDA ID	Always	This field contains the unique identifier of the user defined attribute.
UDA Description	Always	This field contains a description of the user defined attribute.
Module	Always	This field indicates which system module the UDA belongs to. This will always be 'ITEM'.
Display Type	Always	This field contains a code to indicate how the UDA will be displayed to the user. Valid values are in the UDIS code type.
Data Type	Optional	This field indicates what value types are valid for this UDA. Valid values are: • NUM • ALPHA • DATE
Data Length	Optional	This field indicates the length of the field.
Single Value Indicator	Optional	This field indicates whether the UDA is limited to having at most one value. Valid values are Yes (Y) and No (N).
Host Indicator	Optional	
Carton Group	Optional	
Combinability	Optional	

Updated UDAs

When an existing UDA is updated, an update message is triggered to provide the details of the update via the RIB. The update message, like create, will contain the full details in the message for all fields in the message.

Deleted UDAs

When an existing UDA is deleted, this will also trigger a delete transaction to be sent via the RIB to notify external systems that this UDA is no longer valid. The delete message will include only the UDA ID, UDA value and description of the UDA being deleted.

Table 2-115 UDA

Message Element	Included?	Notes
UDA ID	Always	This field contains the unique identifier of the user defined attribute being deleted.

New UDA Details

Creating a new UDA details triggers a message to be sent via the RIB to notify external systems. The UDA value and description are sent for the existing UDA as part of the create message.

Table 2-116 UDA Values

Message Element	Included?	Notes
UDA ID	Always	This field contains the unique identifier of the user defined attribute.
UDA Value	Always	This field contains a unique number identifying the User Defined Attribute value for the UDA. A UDA can have multiple values. For example, Color can be a UDA and it can have different values like Green, Red, Blue, etc.
UDA Value Description	Always	This field contains a description of the UDA.

Updated UDAs

When an existing UDA detail is updated, an update message is triggered to provide the details of the update via the RIB. The update message, will contain the UDA value and description in the message.

Deleted UDAs

When an existing UDA detail is deleted, this will also trigger a delete transaction to be sent via the RIB to notify external systems that this UDA detail is no longer valid. The delete message will include only the UDA ID and UDA value being deleted.

Table 2-117 UDA Values

Message Element	Included?	Notes
UDA ID	Always	This field contains the unique identifier of the user defined attribute being deleted.
UDA Value	Always	This field contains a unique number identifying the User Defined Attribute value for the UDA. A UDA can have multiple values. For example, Color can be a UDA and it can have different values like Green, Red, Blue, etc.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed.

For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XML)
UdaHdrCre	UDA Header Create	UDADesc.xsd
UDAValCre	UDA Detail Create	UDAValDesc.xsd
UDAHdrMod	UDA Header Modify	UDADesc.xsd
UDAValMod	UDA Detail Modify	UDAValDesc.xsd
UDAHdrDel	UDA Header Delete	UDAREf.xsd
UDAValDel	UDA Detail Delete	UDAValRef.xsd

Vendor Publication API

This section describes the vendor publication API.

Functional Area

Suppliers

Business Overview

This API publishes suppliers and supplier address information through the RIB to external systems. This information is further subscribed by integrating systems like Oracle Retail Store Inventory and Operations Cloud Service (SIOCS). Supplier information is published when new supplier sites are created, updates are made to existing sites, or existing supplier sites are deleted. Similarly, addresses are published when they are added, modified, or deleted. The address types that are published as part of this message are:

- Returns (3)
- Order (4)
- Invoice (5)

Only supplier site level information is published. The supplier level information will not be published.

Vendor Creation

Supplier sites can be set up in Merchandising through the application UI, or through subscription of vendor information from external systems. Both methods result in this message being published when a supplier is created. The vendor creation message is published from Merchandising only after the supplier site has been completely set up, including the mandatory addresses and its org unit. Once all these criteria are met for a valid create message, the messages will be combined and sent to RIB. Both active and inactive supplier sites are published.

Vendor Header Modification

Updating an existing supplier site will trigger a header modification message to notify the integrating systems. This message will contain only the header level information that changed.

Table 2-118 Create and Update

Message Element	Included?	Notes
Vendor Header	Always	Child node
Vendor Address	Optional	Child node
Vendor Org Unit	Optional	Child node

Table 2-119 Vendor Header Create and Update

Message Element	Included?	Notes
Supplier	Always	Unique identifying number for a supplier site within the system.
Supplier Name	Always	This field contains the supplier site name.
Supplier Name Secondary	Optional	This type can hold secondary name for the supplier site with a max length of 240 characters.
Contact Name	Always	This field contains the name of the supplier representative contact for this site.
Contact Phone	Always	This field contains a telephone number for the supplier's representative contact.
Contact Fax	Optional	This field contains a fax number for the supplier's representative contact.
Contact Pager	Optional	This field contains a pager number for the supplier's representative contact.
Supplier Status	Always	This field contains the status of the supplier site. Valid values are: • A - Active • I - Inactive
QC Ind	Always	This field determines whether orders from this supplier will require quality control.
QC Percentage	Optional	This field contains the percentage of items per receipt that will be marked for quality checking.
QC Frequency	Optional	This field contains the frequency for which items per receipt will be marked for quality checking.
VC Ind	Always	This field determines whether orders from this supplier will require vendor control.
VC Percentage	Optional	This field contains percentage of items per receipt that will be marked for vendor checking.
VC Frequency	Optional	This field contains the frequency for which items per receipt that will be marked for vendor checking.
Currency Code	Always	This field contains code identifying the currency the supplier site uses for business transactions.
Language	Optional	This field contains the supplier's preferred language.
Terms	Always	This field contains an indicator identifying the purchase terms that will default when an order is created for the supplier site. These terms specify when payment is due and if any discounts exist for early payment.
Freight Terms	Always	This field contains code indicating what freight terms will default when an order is created for the supplier site.

Table 2-119 (Cont.) Vendor Header Create and Update

Message Element	Included?	Notes
Return Allow Indicator	Always	This field indicates whether the supplier site will accept returns. Valid values are Yes (Y) or No (N).
Return Authorization Required	Always	This field indicates if returns must be accompanied by an authorization number when sent back to the vendor.
Returns Minimum Amount	Optional	This field contains a value if the supplier site requires a minimum merchandise value to be returned to accept the return. Returns of less than this amount will not be processed by the system. This field is stored in the supplier's currency.
Return Courier	Optional	This field contains the name of the courier that should be used for returns to the supplier site.
Handling Percentage	Optional	This field contains the default percent to be multiplied by the return's total cost to determine the handling cost for the return.
EDI PO Ind	Always	This field indicates whether purchase orders will be sent to the supplier via EDI.
EDI PO Change	Always	This field indicates whether purchase order changes will be sent to the supplier via EDI.
EDI PO Confirmation	Always	This field indicates whether acknowledgements of purchase orders will be sent from the supplier via EDI.
EDI ASN	Always	This field indicates whether the supplier will send Advance Shipment Notifications electronically.
EDI Sales Report Frequency	Optional	This field contains the EDI sales report frequency for the supplier. Valid values are weekly (W) or daily (D).
EDI Supplier Availability Indicator	Always	This field indicates whether the supplier will send availability via EDI.
EDI Contract Indicator	Always	This field indicates whether the supplier site supports contract ordering sent via EDI.
EDI Invoice Indicator	Always	This field indicates whether invoices, debit memos, and credit note requests will be sent to/from the supplier via EDI.
Cost Change Percentage Variance	Optional	This field contains a percent that determines whether a cost change can be auto approve via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
Cost Change Amount Variance	Optional	This field contains an amount (in supplier currency) that determines whether a cost change can be auto approve via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
Replenishment Approval Indicator	Always	This field indicates whether contract orders created via replenishment should be created in Approved status.
Ship Method	Optional	This field contains the default method used to ship the items on the purchase order from the supplier site. Valid values are held in code type SHPM.

Table 2-119 (Cont.) Vendor Header Create and Update

Message Element	Included?	Notes
Payment Method	Optional	This field indicates the default method for how purchase orders for this site will be paid. Valid options are: • LC - Letter of Credit • WT - Wire Transfer • OA - Open Account
Contact Telex	Optional	This field contains a telex number for the supplier's representative contact.
Contact Email	Optional	This field contains an email address for the supplier's representative contact.
Settlement Code	Always	This field indicates which payment process method is used for the supplier. Valid values are N/A (N) or Evaluated Receipts Settlement (E).
Pre-Mark Indicator	Always	This field indicates whether the supplier site supports pre-marking containers for cross dock orders.
Auto Approved Invoice Indicator	Always	This field indicates whether the supplier's invoices can be automatically approved for payment.
Debit Memo Code	Optional	This field indicates when a debit memo will be sent to the supplier site to resolve a discrepancy. Valid values are: • Y - if debit memos are always to be sent • L - if debit memos are used only if a credit note is not sent by the invoice due date • N - if debit memos are never sent
Freight Charge Indicator	Always	This field indicates whether a supplier site can charge freight costs.
Auto Approve Debit Memo Indicator	Always	This field indicates whether debit memos sent to the supplier site can be automatically approved on creation.
Inventory Management Level	Optional	This field indicates the level for managing supplier inventory information. Valid values are: supplier (S), supplier/location (L), supplier/department (D), or supplier/department/location (A).
Backorder Indicator	Always	This field indicates if backorders or partial shipments will be accepted.
VAT Region	Optional	This field contains the unique identifying number for the VAT region applicable for this site.
Prepay Invoice Indicator	Always	This field indicates whether all invoices for the supplier can be pre-paid.
Service Performed Required Indicator	Always	This field indicates if the supplier's services must be confirmed as performed before paying an invoice from that supplier site.
Invoice Payment Location	Optional	This field indicates where invoices from this supplier site are paid - at the store (S) or centrally through corporate accounting (C).
Invoice Received Location	Optional	This field indicates where invoices from this supplier site are received - at the store (S) or centrally through corporate accounting (C).
Invoice At	Optional	This field indicates if the supplier site invoice lists items at gross cost (G) or net cost (N).

Table 2-119 (Cont.) Vendor Header Create and Update

Message Element	Included?	Notes
Delivery Policy	Always	This field contains the default delivery policy of the supplier site. Valid values come from the DLVY code type.
Comments	Optional	This field contains any miscellaneous comments associated with the supplier.
Default Item Lead Time	Optional	This field holds the default lead time for the supplier site. The lead time is the time the supplier needs between receiving an order and having the order ready to ship. This value will be defaulted to item/supplier relationships.
DUNS Number	Optional	The Dun and Bradstreet number of the supplier.
DUNS Location	Optional	The Dun and Bradstreet number of the location of the supplier.
Bracket Costing Indicator	Always	This field will determine if the supplier site supports bracket costing pricing structures.
VMI Order Status	Optional	This field determines the status in which any inbound POs from this supplier will be created. A NULL value indicates that the supplier is not a VMI supplier.
End Active Date	No	Always null
DSD Supplier Indicator	Always	This field specifies whether the vendor supports DSD ordering, where the supplier replenishes the store directly, creating the PO and receipt at the same time.
Supplier Quantity Level	Always	This field indicates the supplier site order quantity level. Valid values are cases (CA) or eaches (EA).
Supplier Parent	Optional	This is the supplier number for the supplier sites.
Store Delivery Discrepancy	No	Always null
Final Destination Indicator	Always	This field indicates whether the supplier site can ship to final destination or not. Valid values are Yes (Y) or No (N).
Custom Flex Attributes	Optional	Child Node
External Reference Indicator	Optional	This column holds the ID for the supplier used in the external financial system.
Deal Upload Status	No	This field indicates the status in which the deal will be created when uploaded. The valid values are: • W - Worksheet • S - Submitted • A - Approved
Tax ID	No	This field contains the unique tax identification number of the supplier site.
Default Pricing Control	No	Indicates the default value of the pricing control attribute at the supplier site level when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: • R – Retailer • S – Supplier

Table 2-119 (Cont.) Vendor Header Create and Update

Message Element	Included?	Notes
Generate Consignment Order	No	This field specifies whether Merchandising is required to generate POs against consignment sales. Valid values are: Y - Merchandising creates a PO, N - POs are created externally.
Generate Self Invoicing	No	This field specifies whether self-invoicing needs to be done for consignment purchases, or if the vendor will send invoices to be matched. Valid values are: Y - Merchandising creates the invoice, N - Invoice will be created by the vendor.

Vendor Address Creation

Adding new addresses of the types sent in this message for an existing supplier site will trigger an address detail creation message. This message is sent without the vendor header information. The 'key_value_1' field on this message will map to the supplier site.

Adding a new org unit to the supplier site will trigger an org unit create message. This message is also sent along with the header information.

Vendor Address Modification

Any updates made to the supplier site's address information of the types specified above will trigger an address modification message. This message is sent without the vendor header information. The 'key_value_1' field on this message will map to the supplier site. When detail changes are sent, only the values that changed are included in the message.

Table 2-120 Vendor Address Create or Update

Message Element	Included?	Notes
Module	Always	This field indicates the data type that the address is attached to. In this case, it will always be 'SUPP'.
Key Value 1	Always	This field holds the ID the address is attached to. In this case, it will be the supplier number.
Key Value 2	No	This is not used.
Sequence Number	Always	Number indicating the sequence that addresses within the same type were entered.
Address Type	Always	This field contains the address type. For suppliers, the following address types are sent: Returns (3) Order (4) Invoice (5)
Primary Address Indicator	Always	This field indicates whether the address is the primary address for the address type.
Address 1	Always	This field contains the first line of the address.
Address 2	Optional	This field contains the second line of the address.
Address 3	Optional	This field contains the third line of the address.

Table 2-120 (Cont.) Vendor Address Create or Update

Message Element	Included?	Notes
City	Always	This field contains the name of the city that is associated with the address.
State	Optional	This field contains the state abbreviation for the address.
Country ID	Always	This field contains the country where the address exists.
Jurisdiction Code	Optional	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
Post	Optional	This field contains the zip code for the address.
Contact Name	Optional	This field contains the name of the contact for the supplier at this address.
Contact Phone	Optional	This field contains the phone number of the contact person at this address.
Contact Telex	Optional	This field contains the telex number of the contact person at this address.
Contact Fax	Optional	This field contains the fax number of the contact person at this address.
Contact Email	Optional	This field contains the email address of the supplier site's contact person.
Custom Flex Attributes	Optional	Child Node

Vendor Org Unit Creation and Modification

Whenever an org unit is associated with a supplier site, a vendor org unit creation message is triggered. This message is sent with vendor header information. Similarly, when the primary pay site associated with a supplier site is updated, a vendor org unit modification message will be published.

Table 2-121 Vendor Org Unit Create or Update

Message Element	Included?	Notes
Org Unit ID	Always	This field contains org unit ID added or updated for the supplier site.
Primary Pay Site Indicator	Always	This field contains the primary pay site indicator.

Vendor Detail Deletion

Vendor addresses can be deleted only if they are not mandatory, or if they are mandatory but are not marked as the primary address for that address type. The deletion will trigger an address delete message.

Table 2-122 Vendor Address Delete

Message Element	Included?	Notes
Supplier	Always	This field contains the unique identifying number for a supplier within the system. The user determines this number when a new supplier is first added to the system.
Sequence Number	Always	This field contains the number indicating the sequence that addresses within the same type were entered.
Address Type	Always	This field contains the address type. Valid values (e.g. 01 - Business, 02 - Postal, etc.) are on the add_type table.

Deletion of an org unit will be allowed if there are no open purchase orders for the supplier site. This will trigger an org unit delete message.

Table 2-123 Vendor Org Unit Delete

Message Element	Included?	Notes
Org Unit ID	Always	This field contains org unit ID being removed for the supplier site.

Flex Attributes

If any custom flex attributes (CFAS) for the supplier have been added or modified, it will trigger a header modify message. All the entity's active flex attributes from all attribute groups are published as key-value pairs based on the group set view. This CFAS object is embedded in the outbound Vendor message.

Similarly, if any CFAS for the supplier address has been added or modified, it will trigger a vendor address modify message. All the entity's active flex attributes from all attribute groups are published as key-value pairs based on the group set view. This CFAS object is embedded in the outbound Vendor Address message.

Flex attributes can only be added to or updated; they cannot be deleted.

Table 2-124 Flex Attributes

Message Element	Required?	Notes
Name	Always	Holds the attribute name.
Value	Optional	Holds the value of the attribute for number and character type attributes
Value Date	Optional	Holds the date for date type attributes.

Full Message Updates

In cases where the integrating system is unable to receive only the changes, another option is provided that can resend the full supplier details whenever there is a change, in addition to the deltas. This will be published when the system option Publish Full Objects (PUB_FULL_OBJECTS_IND) is set to Deltas and Full (Y). The full message updates, like the vendor create message, will contain the full details in the message (that is, the vendor header

and its flex attributes, the vendor address and its flex attributes, and the org unit information). See above for the format of a vendor create message. This is used for Oracle WMS Cloud integration.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter, thus preventing any further messages from being processed until this is resolved

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
VendorCre	Vendor Create	VendorDesc.xsd
VendorAddrCre	Vendor Address Create	VendorAddrDesc.xsd
VendorOUCre	Vendor Org Unit Create	VendorOUDesc.xsd
VendorHdrMod	Vendor Header Modify	VendorHdrDesc.xsd
VendorAddrMod	Vendor Address Modify	VendorAddrDesc.xsd
VendorOUMod	Vendor Org Unit Modify	VendorOUDesc.xsd
VendorDel	Vendor Delete	VendorRef.xsd
VendorAddrDel	Vendor Address Delete	VendorAddrRef.xsd
VendorOUDel	Vendor Org Unit Delete	VendorOURef.xsd
VendorFulRep	Full message	VendorDesc.xsd

Warehouse Publication API

This section describes the warehouse publication API.

Functional Area

Foundation Data

Business Overview

This API facilitates the transmission of warehouse data to external integrating applications. Merchandising publishes information for both virtual and physical warehouses. Those applications that do not have virtual location logic, such as Store Inventory and Operations Cloud Service (SIOCS), depend upon the RIB to transform the warehouse messages for physical warehouses only.

These messages are triggered by the insert, update, and delete of warehouses and warehouse addresses. Only the primary address of the primary address type is included in this message.

Warehouse Creation

Creating a new warehouse triggers a message to be sent to notify external systems. The warehouse details, along with the addresses, are sent for the new warehouse as part of the create message. Physical warehouses are published only after the primary addresses have been added. Address information is sent only for the physical warehouses.

Warehouse Modification

Creating a new warehouse triggers a message to be sent to notify external systems. The warehouse details, along with the addresses, are sent for the new warehouse as part of the create message. Physical warehouses are published only after the primary addresses have been added. Address information is sent only for the physical warehouses.

Warehouse Create or Update

Message Element	Included?	Notes
Warehouse	Always	The number which uniquely identifies the warehouse. This could be either a physical or virtual warehouse.
Warehouse Name	Always	This field contains the warehouse name. This is published for both physical and virtual warehouse
Address Line 1	Optional	Not used
Address Line 2	Optional	Not used
City	Optional	Not used
County	Optional	Not used
State	Optional	Not used
Country	Optional	Not used
Postal Code	Optional	Not used
E-mail	Optional	This field holds the email address for the location. This is published for only the physical warehouse
Stockholding Indicator	Optional	Indicates if the warehouse is a stock holding location. It will be N for a physical warehouse and Y for a virtual warehouse.
Channel ID	Optional	The channel for which the virtual warehouse has been assigned. This is published for the virtual warehouse
Currency Code	Optional	The code for the currency that the warehouse uses. This is published for the physical and virtual warehouse
DUNS Number	Optional	This field holds the Dun and Bradstreet number to identify the physical warehouse. This is published for the physical warehouse
DUNS Location	Optional	This field holds the Dun and Bradstreet number to identify the physical warehouse. This is published for the physical warehouse
Physical Warehouse	Optional	The number of the physical warehouse corresponding to this virtual warehouse. For physical warehouses, this field will be null.

Message Element	Included?	Notes
Break Pack Indicator	Optional	Indicates whether the physical warehouse can distribute less than the supplier case quantity. Valid values are Y or N. This is published for both physical and virtual warehouse
Redistribution Warehouse Indicator	Optional	A re-distribution warehouse is used to flag purchase orders for review closer to the time of receipt. If this is set to yes (Y) for a physical warehouse, then it indicates that this is a “dummy” location for a purchase order and that the order will have its order quantities redistributed to actual receiving locations close to expected receipt. Valid values are yes (Y) and no (N). This is published for both physical and virtual warehouse
Delivery Policy	Optional	The delivery policy of the physical warehouse. Next Day (NEXT) indicates that if a location is closed, the warehouse will deliver on the next day. Next Valid Delivery Day (NDD) indicates that the warehouse will wait until the next scheduled delivery day. This is published for both physical and virtual warehouse
Contact Person	Optional	Not used
Fax Number	Optional	Not used
Phone Number	Optional	Not used
Default Route	Optional	Not used
Default Carrier Code	Optional	Not used
Default Service Code	Optional	Not used
Expedited Route	Optional	Not used
Expedited Carrier Code	Optional	Not used
Expedited Service Code	Optional	Not used
BOL Upload Type	Optional	Not used
BOL Print Type	Optional	Not used
Lead Time	Optional	Not used. Defaulted to 0.
Distance to Destination	Optional	Not used. Defaulted to 0.
Drop Trailers Accepted	Optional	Not used
Receiving Dock Available	Optional	Not used
Container Type	Optional	Not used
MLD Default Route	Optional	Not used
Unit Pick Container Type	Optional	Not used
Destination Sequence Number	Optional	Not used. Defaulted to 0.
Owning DC	Optional	Not used
Addresses	Optional	Child Node. This is published for physical warehouse only.
Pricing Location	Optional	The location for which pricing information was copied for this virtual warehouse. This Is published for virtual warehouse

Message Element	Included?	Notes
Pricing Location Currency Code	Optional	This is the currency code of the pricing location. . This is published for virtual warehouse
Org Unit ID	Optional	This field holds the org unit ID of the virtual warehouse. This is published for virtual warehouse
Flex Attributes	Optional	Child Node. This is published for physical and virtual warehouses.
Warehouse Secondary Name	Optional	This field holds the secondary name of the warehouse. This Is published for physical and virtual warehouse
VAT Region	Optional	This field holds the VAT region where the physical warehouse belongs. This Is published for physical and virtual warehouse
Organization Hierarchy Type	Optional	Contains the organization type that will be used for reporting purposes for the physical warehouse. The type comes from the organizational hierarchy. Valid values are: • 1 = Company • 10 = Chain • 20 = Area • 30 = Region • 40 = District • 50 = Store
Organization Hierarchy Value	Optional	This field contains the code associated with the specific organizational hierarchy type. Valid values include the company number, chain number, area number, etc.
Primary Virtual Warehouse	Optional	This field holds the virtual warehouse that will used as the basis for all transactions for which only a physical warehouse and not a virtual warehouse has not been specified.
Channel Description	Optional	The channel description for which the virtual warehouse has been assigned.
Restricted Indicator	Optional	Indicates whether the inventory for this virtual warehouse is restricted, meaning it is impacted last an inbound type of transaction occurring at the physical warehouse level. Valid values are Y and N.
Protected Indicator	Optional	Indicates whether the inventory for this virtual warehouse is protected, meaning it is affected last in outbound transactions occurring at the physical warehouse level. Valid values are Y and N.
Forecast Indicator	Optional	This indicator determines if a virtual warehouse should be forecasted. Valid values are Y and N.
Transfer Entity	Optional	Indicates the transfer entity with which this virtual warehouse is associated.
Finisher Indicator	Optional	This field indicates whether a virtual warehouse is an internal finisher or not. Valid values are Y and N.
Inbound Handling days	Optional	This field indicated the number of days that the physical warehouse requires to receive any item and get it to the shelf so that it is ready to pick.

Message Element	Included?	Notes
Virtual Warehouse Type	Optional	This attribute holds values:CS_NT - Non-traditional CS_RG - CSC XD_GS - Global Sourcing XD_RG - Cross Dock
Organization Entity Type	Optional	This field will specify if the virtual warehouse is a legal entity (Importer, Exporter) or a regular warehouse. Valid values are R - regular warehouse (including finisher); M - importer; X - exporter.
Customer Order Location Indicator	Optional	This field indicates if the virtual warehouse can source or fulfill customer orders. Valid values are Y or N.
Default Warehouse	Optional	This field contains the default sourcing warehouse linked to the virtual warehouse.
Gift wrapping Indicator	Optional	This field indicates whether the warehouse supports gift wrapping.
Customer Order Shipping Indicator	Optional	This field indicates whether the virtual warehouse supports shipping customer orders.

Warehouse Addresses

Adding a new primary address for an existing physical warehouse will trigger an address detail create message.

Any updates made to the primary address information of the warehouse will trigger an address modification message. The `key_value_1` field on this message will map to the physical warehouse.

Warehouse Address Create or Update

Message Element	Included?	Notes
Module	Always	This field indicates the data type to which the address is attached. In this case, it will always be 'WH'.
Key Value 1	Always	This field holds the warehouse ID the address is attached to.
Key Value 2	No	This is not used.
Sequence Number	Always	Number indicating the sequence that addresses within the same type were entered.
Address Type	Always	This field contains the address type. Only the primary address of the primary address type Business (01) is published for the physical warehouse.
Primary Address Indicator	Always	This field indicates whether the address is the primary address for the address type.
Address 1	Always	This field contains the first line of the address.
Address 2	Optional	This field contains the second line of the address.
Address 3	Optional	This field contains the third line of the address.

Message Element	Included?	Notes
City	Always	This field contains the name of the city that is associated with the address.
State	Optional	This field contains the state abbreviation for the address.
Country ID	Always	This field contains the country where the address exists.
Jurisdiction Code	Optional	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
Post	Optional	This field contains the postal code for the address.
Contact Name	Optional	This field contains the name of the contact for this address.
Contact Phone	Optional	This field contains the phone number of the contact person at this address.
Contact Telex	Optional	This field contains the telex number of the contact person at this address.
Contact Fax	Optional	This field contains the fax number of the contact person at this address.
Contact Email	Optional	This field contains the email address of the contact person.
Custom Flex Attributes	Optional	Child Node

Deleting Warehouses

If a warehouse is deleted in Merchandising, a delete message will be triggered to inform dependent systems.

Warehouse Delete

Message Element	Included?	Notes
Warehouse	Always	This field contains the unique warehouse number that has been deleted – either a virtual or physical warehouse.
Address	Optional	Child Node

Warehouse Address Delete

Warehouse addresses can be deleted only if they are not mandatory or if they are mandatory but are not marked as the primary address for that address type. The deletion will trigger an address delete message.

Message Element	Included?	Notes
Address	Always	This field contains the unique address key that has been deleted.

Flex Attributes

If you have defined any custom flex attributes (CFAs) for warehouses or warehouse addresses, then they can be integrated as part of this API. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute.

Custom Flex Attributes

Message Element	Required?	Notes
Name	Always	Indicates the name of the column defined in the group set view for flex attributes defined for transfers.
Value	Always	Indicates the value of the attribute for the transfer if the attribute is a character or number.
Value Date	Always	Indicates the value of the attribute for the transfer if the attribute is a date.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
WHCre	WH Create Message	WHDesc.xsd
WHMod	WH Modify Message	WHDesc.xsd
WHDel	WH Delete Message	WHRef.xsd
WHDtlCre	WH Detail Create Message	WHDesc.xsd
WHDtlMod	WH Detail Modify Message	WHDesc.xsd
WHDtlDel	WH Detail Delete Message	WHRef.xsd
WHAddCre	WH Address Create	WHAddrDesc.xsd
WHAddMod	WH Address Modify	WHAddrDesc.xsd

Work Orders In Publication API

This section describes the work order in publication API.

Functional Area

Purchase Orders

Business Overview

A work order provides direction to a warehouse, such as Oracle Retail Warehouse Management (RWMS), about work that needs to be completed on items contained in a purchase order prior to shipping them on to stores or other warehouses. Merchandising publishes work orders soon after it publishes the purchase order itself. This is referred to as a Work Order In message. Work orders are defined at the physical location level. The message family manager will send to the warehouse at which the work order will be done. This is used by the RIB publication adaptor for routing messages to the appropriate warehouse.

Note

This integration is not used in Merchandising's integration to Oracle WMS Cloud.

New Work Order

Creating a new work order for an approved order triggers a message to be sent to notify the warehouse of additional activities that must be performed on the order upon receipt. The message includes the following: work order ID, order number, warehouse that will perform the work, the locations where the items on the order will be sent onto, and details on the items and work to be done, including a sequence and WIP (work in progress) code, which is stored in Merchandising codes table, under code type WWIP. It is assumed that the WIP codes used by Merchandising are coordinated with your warehouses that will be receiving the updates.

The format for creating and updating inbound work orders is shown below.

Table 2-125 Work Order Header

Message Element	Included?	Notes
Work Order ID	Always	This field contains the unique identifier for work in process associated with an order or transfer.
Order Number	Always	This field contains the order number associated with the work order.
Work Order Detail	Always	Child node

Table 2-126 Work Order Detail

Message Element	Included ?	Notes
Warehouse	Always	The physical warehouse where the work order will be done.
Item	Always	The item on which the work order will be done.

Table 2-126 (Cont.) Work Order Detail

Message Element	Included ?	Notes
Location Type	Always	The location type of the final destination. Valid values are: S - Store W - Warehouse
Location	Always	The final destination for the item on the order. If the item is pre-distributed this location will be the store or warehouse on the allocation. If not, it will be the same warehouse as above.
Sequence Number	Always	The sequence number that is needed to ensure each record is unique, since the same work-in-progress code can be listed more than once on the same work order.
WIP Code	Always	The WIP code for the work to be done on the item. Valid values are in the codes table under the WWIP code type.
Instructions	No	Not used

Updated Work Order

When an existing work order for an approved order is updated, an update message is triggered to provide the details of the update. The update message, like create, will contain the full details in the message for all fields in the message.

Deleted Work Order

When a work order is deleted for a purchase order, this will also trigger a delete transaction to be sent to notify external systems. The delete message will include the work order ID, order number at the header level and the warehouse, item, location type, location, sequence number and WIP code at a detail level. It is also possible to delete just a detail from the work order.

The format for deleting inbound work orders or work order details is shown below.

Table 2-127 Work Order Header

Message Element	Included?	Notes
Work Order ID	Always	This field contains the unique identifier for work in process associated with an order or transfer.
Order Number	Always	This field contains the order number associated with the work order.
Work Order Detail	Optional	Child node

Table 2-128 Work Order Detail

Message Element	Included?	Notes
Warehouse	Always	The physical warehouse where the work order will be done.
Item	Always	The item on which the work order will be done.

Table 2-128 (Cont.) Work Order Detail

Message Element	Included?	Notes
Location Type	Always	The location type of the final destination. Valid values are: S - Store W - Warehouse
Location	Always	The final destination for the item on the order. If the item is pre-distributed this location will be the store or warehouse on the allocation. If not, it will be the same warehouse as above.
Sequence Number	Always	The sequence number that is needed to ensure each record is unique, since the same work-in-progress code can be listed more than once on the same work order.
WIP Code	Always	The WIP code for the work to be done on the item. Valid values are in the codes table under the WWIP code type.

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This would bring down the RIB adapter thus preventing any further messages from being processed until this is resolved.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
InBdWOCre	Work Order In Create	WOInDesc.xsd
InBdWOMod	Work Order In Modify	WOInDesc.xsd
InBdWODEl	Work Order In Delete	WOInRef.xsd

Work Orders Out Publication API

This section describes the Work Orders out Publication API.

Functional Area

Transfers

Business Overview

This publication API facilitates the transmission of outbound work orders for the finisher from Merchandising to external systems. Only transfers that pass through a finisher before reaching the final location can be associated with work orders. The work order provides instructions for one or more of the following tasks to be completed at the finisher location:

- Perform an activity on an item, such as monogramming or ticketing.
- Transform an item from one thing into another, such as dyeing a white t-shirt black.
- Combine bulk items into a pack or break down a pack into its component items.

Outbound work orders are not bundled with transfer messages, because multi-legged transfers can be routed to either internal finishers (held as virtual warehouses) or external finishers (held as partners).

All activities, transformations, and packing details are contained in the same message. Because Merchandising does not allow users to modify work order activities, transformation or packing information for an approved transfer (it must be brought back to Input status), separate detail-level messages of any type (create, delete, update) are never published.

Outbound work order delete messages are published when the second leg of a multi-legged transfer is unapproved. This can be accomplished through the un-approval of an entire multi-legged transfer or the un-approval of the second leg only. A two-leg transfer that has had the first leg shipped can be set back to In Progress status to make changes to the work order activities and the final location. When the action has occurred, only the second leg is really set back to in progress. The first leg remains in shipped status.

New Work Order

A new work order is created when a transfer with finishing is approved. This triggers a message notifying the external system. The message includes the following:

- Transfer work order ID
- Finisher ID
- Transfer number
- Transfer parent number
- Inventory type (available or unavailable)
- Work order detail information - including item, destination location, item's inventory status, and the work order activity information (costs, comments)
- Transformation details - including the from and to item IDs
- Packing information - including the from items and to items

The format for creating and updating outbound work orders is shown below.

Table 2-129 Work Order Header

Message Element	Included?	Notes
Work Order ID	Always	This field contains the unique identifier for the work in process associated with a transfer.
Destination ID	Always	This field contains the source location of the transfer.
Distro Number	Always	This field contains the first leg transfer number of the 2-legged transfer.
Work Order Detail	Always	Child node
Distro Parent Number	Optional	This field contains the second leg transfer number of the 2-legged transfer.
Work Order Transformation	Optional	Child node
Work Order Packing	Optional	Child node

Table 2-129 (Cont.) Work Order Header

Message Element	Included?	Notes
Inventory Type	Optional	This field contains the code to indicate whether the inventory on the transfer is available. Valid values are Available (A) and Unavailable (U).

Table 2-130 Work Order Detail

Message Element	Included?	Notes
Destination ID	Always	This field contains to the destination location for the transfer.
Item ID	Always	This field contains the item on which the work order will be performed.
Work In Process Sequence Number	No	Not used
Work In Process Code	No	Not used
Personalization	No	Not used
Instructions	No	Not Used
Order Line Number	No	Not Used
Auto Complete	No	Not Used
Work Order Activity	Optional	Child node
Inventory Status	Optional	This field contains the inventory status of the transfer detail.

Table 2-131 Work Order Activity

Message Element	Included?	Notes
Activity ID	Always	This field contains a code number indicating the activity that will be performed on the item.
Activity Cost	Optional	This field contains the per unit cost of the finishing activity.
Comments	Optional	This field contains any comments regarding this activity.

Table 2-132 Work Order Transformation

Message Element	Included?	Notes
From Item	Always	This field contains the original item on the transfer.
To Item	Always	This field contains the item which results from the finishing activity.

Table 2-133 Work Order Packing

Message Element	Included?	Notes
To Pack	Always	Child node

Table 2-133 (Cont.) Work Order Packing

Message Element	Included?	Notes
From Pack	Always	Child node

Table 2-134 Work Order Pack To

Message Element	Included?	Notes
To Item	Always	This contains the item resulting from the packing process.

Table 2-135 Work Order Pack From

Message Element	Included?	Notes
From Item	Always	This contains the component item used in the packing process.

Deleted Work Order

When a transfer with finishing is deleted or unapproved, this will trigger a delete transaction message to an external system. The delete message contains the work order ID to be deleted.

Table 2-136 Work Order Header

Message Element	Included?	Notes
Work Order ID	Always	This field contains the unique identifier of the work order being deleted.
Destination ID	Always	This field contains the source location of the transfer.
Distro Number	Always	This field contains the first leg transfer number of the 2-legged transfer.
Work Order Detail	Always	Child node
Distro Parent Number	Optional	This field contains the second leg transfer number of the 2-legged transfer.

Table 2-137 Work Order Detail

Message Element	Included?	Notes
Work Order ID	Always	This field contains the unique identifier of the work order being deleted.
Destination ID	Always	This field contains to the destination location for the transfer.
Item ID	Always	This field contains the item on which the work order will be performed.
Work In Process Sequence Number	No	Not used
Order Line Number	No	Not used

Error Handling

If any errors are encountered while publishing the message, a fatal error with status E (Error) is sent to RIB. This will bring down the RIB adapter, thus preventing any further messages from being processed until this is resolved.

Message XSD

Here are the filenames that correspond to each message type. Please consult RIB documentation for each message type to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
OutBdWoCre	Work Order Create Message	WOOOutDesc.xsd
OutBdWoDel	Work Order Delete Message	WOOOutRef.xsd

3

RIB Subscription Designs

This chapter provides an overview of the RIB subscription APIs used in by Merchandising.

Allocation Subscription API

This section describes the allocation subscription API.

Functional Area

Allocation

Business Overview

This API allows an external application to create, update, and delete allocations within Merchandising. The Oracle Retail Allocation Cloud Service does **not** use this API to interface allocations to Merchandising. Allocations created or updated using this API include those that are based on warehouse inventory, as well as those using inventory from another transaction, such as a purchase order, another allocation, a transfer, or a shipment (Vendor ASN or BOL) intended for a warehouse location. When allocating using something other than warehouse inventory, the ID of the transaction must also be included.

The receiving location for an allocation created through the API can be -

- Intra/Intercompany Stockholding store (both company owned and franchise)
- Intercompany Non-stockholding company store
- Intra/Intercompany Non-stockholding franchise store
- Stockholding warehouse location, except internal finishers.

If an allocation for a franchise store is received, Merchandising will also create a corresponding franchise order. This API supports multiple destination locations of the same or different type (warehouses as well as stores) as part of the detail section within the same message.

Allocation details can be created, edited, or deleted within the allocation message, including adding new locations to existing allocations, or modifying quantities for existing locations. If modifying an existing location, it assumes that the quantity that is passed in the message is an adjustment to the current quantity as opposed to an overwrite of the original value.

Detail lines can be individually removed from an allocation if the item on the detail line is not in-transit, received, or in progress. An entire allocation can be deleted if none of details are in-transit, have been received or is being processed by the source warehouse.

Allocation Creation

When new allocations are being created, this API first validates that all the required fields are present. After that, business level validations are performed on the input information. Detail line items must exist on an allocation header create message for an allocation to be created. New item location relationships will be created for allocation detail line items that did not previously exist.

Allocation Modification

Modifying an existing allocation will first validate the existence of the allocation header for modification. Only the allocation description, release date, and status can be modified at the header level.

Allocation Create or Update

Message Element	Included?	Notes
Allocation	Always	Contains the unique identifier of the allocation. This should fall within the range of Merchandising IDs already designated for allocations.
Allocation Description	Always	Contains the user defined description of the allocation.
Order No.	Optional	Contains the purchase order with which the allocation is associated. Only used if the source for the allocation is a purchase order.
Item	Always	Contains the transaction level item that is being allocated.
From Location	Always	Contains the location that is the source of the allocation. This must be a valid stockholding virtual warehouse.
Release Date	Optional	Contains the earliest date on which the warehouse should ship the allocation.
Origin Indicator	Optional	Indicates the source application that sends the allocation. Valid values are Oracle Retail Advanced Inventory Planning (AIP) and externally generated (EG). It will be defaulted to EG, if not provided.
Allocation Details	Optional	Child Node
Doc ID	Optional	Contains identification number for a transfer, another allocation, bill of landing number (BOL), or advanced shipping notice (ASN) number for a purchase order. This field is populated according to document type and indicates where the inventory for the allocation should be sourced. This should be null if the source for the allocation is warehouse inventory or a purchase order.
Documentation type	Optional	Contains the type of allocation product source. Valid values are ASN, Transfer (TSF), Bill of Lading (BOL), or Allocation (ALLOC). When this is passed in as null, the source is assumed to be a PO or warehouse inventory if no PO number is provided.
Context Type	Optional	Contains the actual purpose behind the creation of the allocation in the overall business setup. Valid codes need to be defined in the code type = 'CNTX'. For example, Promotion, Repairing, and so on.
Context Value	Optional	Contains the value related to the specified context type, wherever applicable. For example: Promotion ID when the context type is set as 'Promotion'.
Status	Optional	If specified, needs to be either 'R' (Reserved) or 'A' (Approved).

Allocation Details

New detail records can be added to an existing allocation. Modification of the existing details on the allocation is also supported by this API. If modifying an existing location, Merchandising assumes that the quantity passed in the message is an adjustment to the current quantity as opposed to an overwrite of the existing value. The API verifies the allocation is not in-transit, received, nor in progress prior to making any modification and also that the quantity does not fall to zero or below.

Allocation Details Create or Update

Message Element	Included?	Notes
To Location	Always	Contains the destination location of the allocation. This can be a stockholding store, a non-stockholding company store with a transfer entity different than that of the source warehouse, a non-stockholding franchise store, or a stockholding warehouse location, except internal finishers.
To location Type	Always	Contains the type of the destination location. Valid values are S (store) and W (warehouse)
Quantity Allocated	Always	Contains the allocated quantity of the item for the destination location. When the allocation is being created, this value must be a positive integer. If this value is being modified, it will contain the quantity that needs to be adjusted (positive or negative), rather than an override value
In Store Date	Optional	Contains the date the item is to be present in the destination store or warehouse. This date will be included in the Merchandising publication for communication to the warehouse.

Deleting Allocations

Allocation Header Delete

During an allocation header delete, on successfully validating the information in the message, the allocation header is updated to Cancelled (C) status.

Message Element	Included?	Notes
Allocation	Always	Contains the unique identifier of the allocation being deleted.
Allocation Detail	Optional	Child Node

Allocation Details Delete

The detail records are validated and deleted from the allocation on receipt of a detail delete message.

Message Element	Included?	Notes
To Location	Always	Contains the destination location being deleted.

Message Element	Included?	Notes
To Location Type	Always	Contains the location type of the destination location being deleted.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
XAllocCre	External Allocation Create via RIB	XAllocDesc.xsd
XAllocDel	External Allocation Delete via RIB	XAllocRef.xsd
XAllocDtlCre	External Allocation Detail Create via RIB	XAllocDesc.xsd
XAllocDtlDel	External Allocation Detail Delete via RIB	XAllocRef.xsd
XAllocDtlMod	External Allocation Detail Modification via RIB	XAllocDesc.xsd
XAllocMod	External Allocation Modification via RIB	XAllocDesc.xsd

Appointment Subscription API

This section describes the appointments subscription API.

Functional Area

Inventory

Business Overview

Merchandising subscribes to an appointment when merchandise arrives at a location. It also processes the appointment messages and attempts to receive against and close out the appointments. In addition, Merchandising attempts to close the document that is related to the appointment, when applicable. A document can be a purchase order, a transfer, or an allocation.

New Appointments

An appointment is created when shipment is about to arrive at a location. After performing the business level validation this message will update the status to Scheduled (SC). This message contains the location level information.

New Appointment Details

This message contains the item information associated with an appointment, including the ASN and the document number (PO, transfer, or allocation).

Updated Appointments

An update message updates the status of an existing appointment if already exists. Valid values for the status column include:

- SC–Scheduled
- MS–Modified Scheduled
- AR–Arrived
- AC–Closed

Updated Appointment Details

This message updates an appointment detail record that was previously sent, such as an update to the quantity for an item. If the record doesn't already exist, it is added.

Appointment Header

Message Element	Required?	Notes
Appointment Number	Always	This field contains the unique number generated by warehouse while creating an appointment.
From Location	Always	This field contains the location where the merchandise has been sent.
To Location	Optional	Not used.
Appointment Start	Optional	The date and time of the scheduled appointment; that is, when the goods are expected to arrive at the receiving location.
Appointment Action Status	Always	This field contains the status of the appointment. Valid values include: SC - Scheduled, MS - Modified Scheduled, AR - Arrived, and AC-Closed.
Appointment Details		
Appointment Receipt	Optional	The actual date and time upon which the trailer arrived at the receiving location for the appointment.

Appointment Detail Message

Message Element	Required?	Notes
Item	Always	This field contains the items shipped to the location.
Unit Quantity	Always	This field contains the quantity of the item slated to be sent to the location.

Message Element	Required?	Notes
PO Number	Always	This field contains the purchase order, transfer or allocation corresponding to the shipped merchandise.
Document type	Always	This field indicates the type of document corresponding to the shipped merchandise. Possible choices are Purchase Order (P), Transfer (T), or Allocation (A).
ASN Number	Optional	This field contains the advance shipping notice number associated with the appointment. It is populated only when the appointment is based on an ASN.

Delete Appointments

This message deletes any existing appointment and detail records.

Delete Detail Level Appointments

This API deletes a detail record for an existing appointment.

Appointment Header

Message Element	Required?	Notes
Appointment Number	Always	This field contains appointment number being deleted.
From Location	Always	This field contains the location from which the appointment is being deleted.

Appointment Detail Message

Message Element	Required?	Notes
Item	Always	This field contains the item being deleted from the appointment.
PO Number	Always	This field contains the purchase order, transfer or allocation containing the item being deleted from the appointment.
ASN Number	Optional	This field contains the advance shipping notice number associated with the item/appointment being deleted. It is populated only when the appointment is based on an ASN.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
Appointcre	Appointment Create Message	AppointDesc.xsd
Appointhdrmod	Appointment Header Modify Message	AppointDesc.xsd
Appointdel	Appointment Delete Message	AppointRef.xsd
Appointdtlcre	Appointment Detail Create Message	AppointDesc.xsd
Appointdtlmod	Appointment Detail Modify Message	AppointDesc.xsd
Appointdtldel	Appointment Detail Delete Message	AppointRef.xsd

ASN Outbound Subscription API

This section describes the ASNOUT subscription API.

Functional Area

Inventory

Business Overview

Merchandising receives advanced shipping notifications (ASNs), also known as a bill of lading (BOL) messages, from a warehouse management system (WMS), like Oracle Warehouse Management Cloud, or a store inventory system like Oracle Retail Store Inventory and Operations Cloud Service (SIOCS).

These ASNs are notifications to Merchandising that inventory is moving from one location to another. These notification messages contain data that is used by Merchandising to create or modify a shipment record. ASNs are received for:

- Pre-existing allocations.
- Pre-existing transfers.
- Externally generated transfers, created in the store or warehouse (created as transfer type of EG within Merchandising).

An ASN message may contain multiple transfers or allocations, and as a result, the shipment record in Merchandising will reflect these multiple movements of merchandise. A BOL number on the shipment record is a means of tracking one or more transfers and allocations back through the respective stock order records. Shipments for customer orders, franchise orders, and franchise returns are also managed through this API. If the receiving location is a non-stockholding location, like in the case of a warehouse shipment to a non-stockholding franchise store, or a warehouse shipment direct to a customer (that is processed through a non-stockholding store) then the shipment will be auto-received when processed by Merchandising.

Note

ASNs related to a purchase order from a supplier are classified as an Inbound ASNs. Details for those types of expected shipments are found in the ASN In Subscription API section.

Other Notes

- For customer order fulfillment, SIOCS will send an ASN Out message that does not include a ship-to location. Merchandising ignores these messages.
- Store to customer fulfillment request will not have associated transfer in Merchandising. When Oracle Retail Store Inventory and Operations Cloud Service (SIOCS) ships the customer order, SIOCS will generate an Outbound ASN message with an empty To Location or with Location Type as Customer (C). Since there are no associated transfers in Merchandising, Merchandising will not process these Outbound ASN messages. The reserved inventory will be backed out in Merchandising when Merchandising processes a SALES transaction backend.
- Messages consumed through this API can create new shipments or update existing shipments. A new shipment record will be created in Merchandising with Input status if the BOL number is not yet associated to any shipment record. If the BOL number is already associated to a shipment record, the shipment record will be updated accordingly.
- The Universal Identification Number (UIN) child node may be included in the message, but this information is not used by Merchandising. This is used by SIOCS.
- A Shipment and Carton Custom Flex Attribute child nodes can be included in the message, but this information is not used by Merchandising. This is used by SIOCS.

Shipment Message Details

When inventory is shipped from one location to another, Merchandising will be notified and will then create a shipment record based on the content of the message received. If the shipment already exists, the details of the existing shipment will be updated. The message includes the following:

ASNOUT Details

Message Element	Required?	Notes
Schedule Number	Optional	Not used by Merchandising.
Auto Receive Indicator	Optional	Not used by Merchandising.
To Location	Optional	This field contains the location where the shipment will be delivered.
To Location Type	Optional	This field contains the location type of the location where the shipment will be delivered to. Valid values are store (S), warehouse (W), or finisher (E).
To Store Type	Optional	Not used by Merchandising.
To Stockholding Indicator	Optional	Not used by Merchandising.
From Location	Always	This field contains the location from which the shipment was sourced. This applies to transfer and allocation shipments.

Message Element	Required?	Notes
From Location Type	Optional	This field contains the location type of the location from which the shipment was sourced. Valid values are store (S) or warehouse (W).
From Store Type	Optional	Not used by Merchandising.
From Stockholding Indicator	Optional	Not used by Merchandising.
ASN Number	Optional	This field contains the bill of lading number associated with the shipment.
ASN Type	Optional	Not used by Merchandising.
Container Quantity	Optional	This field contains the number of boxes associated with the shipment.
BOL Number	Always	This field contains the transaction sequence number from the transfer shipment confirmation process. This is the same as the ASN Number.
Shipment Date	Optional	This field contains the date the transfer and/or allocation was shipped.
Estimated Arrival Date	Optional	This field contains the estimated arrival date when the shipment is expected to arrive at the destination.
Shipment Address 1	Optional	Not used by Merchandising.
Shipment Address 2	Optional	Not used by Merchandising.
Shipment Address 3	Optional	Not used by Merchandising.
Shipment Address 4	Optional	Not used by Merchandising.
Shipment Address 5	Optional	Not used by Merchandising.
Shipment City	Optional	Not used by Merchandising.
Shipment State	Optional	Not used by Merchandising.
Shipment Zip Code	Optional	Not used by Merchandising.
Shipment Country ID	Optional	Not used by Merchandising.
Trailer Number	Optional	Not used by Merchandising.
Seal Number	Optional	Not used by Merchandising.
Transshipment Number	Optional	Not used by Merchandising.
Comments	Optional	This field contains any miscellaneous comments about the shipment.
Carrier Code	Optional	This field contains the courier that will deliver the shipment.
Carrier Service Code	Optional	This field contains the service level code for the courier that will deliver the shipment. Valid values are found in code type CSVC. Not used by Merchandising.
System Code	Optional	Not used by Merchandising.
From Location Virtual Warehouse	Optional	Not used by Merchandising.

Child Nodes

- ASNOUT Distro Details

ASNOUT Distro Details

This level of the message contains the details about the individual stock orders contained in the shipment.

Message Element	Required?	Notes
Distro Number	Always	This field contains the transfer number or allocation number associated with the shipment.
Distro Doc Type	Always	This field determines if the Distro Number specified is a Transfer (T) or an Allocation (A).
Customer Order Number	Optional	This contains the customer order number associated with the transfer on the shipment.
Fulfill Order Number	Optional	This contains the fulfillment order number associated with the customer order number for the transfer on the shipment.
Consumer Direct	Optional	Not used by Merchandising.
Comments	Optional	This field contains any comments about the stock orders contained in the shipment.

Child Nodes

- ASNOUT Carton Details

ASNOUT Carton Details

This section of the message contains details about the cartons for a distro on a shipment.

Message Element	Required?	Notes
Final Location	Optional	Not used by Merchandising.
Container ID	Always	This field contains the carton number for shipments originating from the ASN process as carton shipments. This field will be zero for all shipments that are not at a carton level.
Container Weight	Optional	Not used by Merchandising.
Container Length	Optional	Not used by Merchandising.
Container Width	Optional	Not used by Merchandising.
Container Height	Optional	Not used by Merchandising.
Container Cube	Optional	Not used by Merchandising.
Expedite Flag	Optional	Not used by Merchandising.
In Store Date	Optional	Not used by Merchandising.
Tracking Number	Optional	This field contains a unique tracking number that is used to track containers through a carrier's system. Not used by Merchandising.
Freight Charge	Optional	Not used by Merchandising.
Master Container ID	Optional	Not used by Merchandising.
Comments	Optional	This field contains any comments about the shipment container.

Message Element	Required?	Notes
Weight	Optional	This field contains the actual weight shipped for the container.
Weight UOM	Optional	This field contains the unit of measurement for weight (e.g., pounds, kilograms) that was shipped.
Carrier Shipment Number	Optional	Not used by Merchandising.
Original Item ID	Optional	Not used by Merchandising.

Child Nodes

ASNOUT Item Details

ASNOUT Item Details

This section outlines details about the items in the carton.

Message Element	Required?	Notes
Item ID	Always	This field contains the unique identifier for the item.
Unit Quantity	Always	This field contains the quantity of the item shipped in the carton for this shipment in the standard unit of measure.
Gross Cost	Optional	Not used by Merchandising.
Priority Level	Optional	Not used by Merchandising.
Order Line Number	Optional	This field is used to carry the customer order line number value for customer orders. This field is not used by Merchandising for non-customer orders.
Lot Number	Optional	Not used by Merchandising.
Final Location	Optional	Not used by Merchandising.
From Disposition	Optional	This value is used to determine if the quantity shipped is available or unavailable. Valid values for this field are inventory status codes (INV_STATUS_CODE) in the INV_STATUS_CODES table in Merchandising.
To Disposition	Optional	Not used by Merchandising.
Voucher Number	Optional	Not used by Merchandising.
Voucher Expiration Date	Optional	Not used by Merchandising.
Container Quantity	Optional	Not used by Merchandising.
Comments	Optional	Not used by Merchandising.
Unit Cost	Optional	This field contains the unit cost of the item in the shipment. This is used only for the Brazil Localization setup to calculate transaction code 74 (Recoverable Tax for Destination Location). In all other cases this should be NULL.
Base Cost	Optional	This value will be used for the Brazil Localization setup only to get the base cost (BC) from Fiscal Management for a transfer, which will flow into Merchandising. In all other cases this should be NULL.

Message Element	Required?	Notes
Weight	Optional	This field contains the actual weight shipped. This may be included for catch weight items. If not included Merchandising will use the average weight or nominal weight for the item at the shipping location.
Weight UOM	Optional	This field contains the unit of measurement for weight (for example, pounds, kilograms) shipped. Required if weight is included in the message.
Inventory Identifier Type	Optional	This field contains the inventory identifier type, which is passed during shipment of transfer/allocations. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, include Lot (L), Expiry Date (E), Import Document (D).
Inventory Id	Optional	This fields contains the inventory identifier ID, which is provided on shipment transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the *Oracle Retail Integration Guide* for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
asnoutcre	ASN Outbound Create Message	ASNOutDesc.xsd

Cost Change Subscription API

This section describes the cost change subscriptions.

Functional Area

Price and Cost

Business Overview

Merchandising exposes an API that will allow external systems to update unit cost within Merchandising. Cost changes can be performed at the item level, or at the following levels of the organization hierarchy: chain, area, region, district, and store. Unit costs are updated for all

stores within the location group. Because warehouses are not part of the organization hierarchy, they are only impacted by cost changes applied at the warehouse level.

All cost changes that are sent through this API are executed immediately. The cost change subscription creates both the cost change events with an effective date of the current date, as well as updates unit costs for item/locations that already exist in Merchandising. It does not create or delete item/locations in Merchandising.

In addition to RIB, Merchandising also exposes this API as a web service. The web service takes in a collection of cost changes and will return success and failure through the service response object. See the "[Cost Change Service](#)" section of this document in the "[Provider Services](#)" section of "[SOAP Web Services](#)" more information.

This API checks that the required fields are provided and checks the supplier's currency and the item status. If differentiator IDs are passed in, it verifies that they are valid for the passed in item. The API also retrieves the following:

- Transaction level items, if the passed in item is an item parent
- All locations based on the passed in hierarchy type and value, if provided.
- All item/location combinations where the passed in supplier/country is the primary supplier/country at an item location.
- All orderable buyer packs that the passed-in item or its children, if above transaction level
- All item/locations on approved (and worksheet) order, if the recalculate order indicator is set to Yes.

This API will perform the following actions:

- Create a cost change event in Executed status, with an effective date of the current date.
- Update the unit cost in Merchandising for all item/supplier/country and item/supplier/country/locations based on the information provided.
- Create price history for all item/locations that got updated as part of the cost change.
- If the recalculate order indicator is Yes, update all relevant order/item/locations unit cost in merchandising.

It is important to note that cost changes sent through this API do not include estimated landed costs. The cost updated here is the default purchase cost, before any deals, that will be used for purchase orders created in Merchandising, similar to cost changes initiated in Merchandising.

The format for creating and updating cost changes is shown below.

Table 3-1 Cost Change

Message Element	Required?	Notes
Item	Yes	This field contains the item to which the cost change applies. It can be a parent item, but its item level cannot be greater than its transaction level. This cannot be a buyer pack.
Supplier	Yes	This field contains the number of the supplier. The supplier must be a valid supplier in merchandising. This can be a primary or non-primary supplier.

Table 3-1 (Cont.) Cost Change

Message Element	Required?	Notes
Origin Country ID	Yes	This field contains the identifier of the origin country of the item/supplier to which the cost change will apply. This value must be a valid country in merchandising. This can be the primary or a non-primary country.
Diff ID	No	This field contains the identifier for a differentiator. This can be used with a parent or grandparent item. This value must be a valid differentiator ID in merchandising.
Unit Cost	Yes	This field contains the new unit cost of the item in the currency specified on the message.
Recalculate Order Indicator	Yes	This field indicates if orders in approved status for items on the cost change will be recalculated with the new cost. Valid values are Yes (Y) and No (N).
Currency Code	Yes	This field contains the currency code of the unit cost. This must be a valid currency code in merchandising.
Hierarchy Level	No	This field indicates the level of the organizational hierarchy to which the cost change applies. Valid values are: • CH - chain • AR - area • RE - region • DI - district • S - store • W - warehouse
Cost Change Hierarchy	No	Child node
Custom Flex Attributes	No	Child node
Purchase Rate	No	This field contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 3-2 Cost Change Hierarchy

Message Element	Required?	Notes
Hierarchy Value	Yes	This field contains the hierarchy value at the specified level of the hierarchy which encompasses the locations affected by the cost change. This value must exist as a valid chain, area, region, district, store or warehouse in Merchandising.

Custom Flex Attributes

If any custom flex attributes (CFAS) for the cost change have been added or modified, it will trigger an update message. The node of the integration that supports this will contain the name of the attribute as it is defined in the group set level view, the value of the custom attribute. If it is a date attribute, the date value is in a separate field. Flex attributes can only be added to or updated; they cannot be deleted.

Table 3-3 Flex Attributes

Message Element	Required?	Notes
Name	Yes	Holds the attribute name.
Value	Conditional	Holds the value of the attribute for number and character type attributes. Either the value or the value date field should be provided.
Value Date	Conditional	Holds the date for date type attributes. Either the value or the value date field should be provided.

Error Handling

This API ensures that the correct message type is passed in for cost change messages. If the message type is invalid, an error status is returned to the external system, along with the appropriate error message. This is to inform the external system that the message type is invalid.

The standard error handling functions of Merchandising are in place in this API and return messages as appropriate to the outcome.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
xcostchgmod	External Cost Change Modify	XCostChgDesc.xsd

Currency Exchange Rate Subscription API

This section describes the currency exchange rates subscription API.

Functional Area

Currency Exchange Rates

Business Overview

Currency exchange rates constitute financial information that is subscribed to by Merchandising. A currency exchange rate is the price of one country's currency expressed in another country's currency. This API assumes the currency codes are already present in Merchandising. This API supports creating new rates by date and updating existing rates for the same conversion date. Deleting previously created rates is not supported.

Table 3-4 Currency Exchange Rates

Message Element	Required?	Notes
From Currency	Yes	The source currency code of the currency exchange rate.
To Currency	Yes	The resultant currency code of the currency exchange rate.
Conversion Date	Yes	Contains the date on which the currency rate became or will become active.
Conversion Rate	Yes	The exchange rate between the two currency codes for the type and effective date.
Conversion Type	Yes	This field identifies the type of exchange rate the history exists for. Valid values are defined under code type EXTP: • Consolidation – C • Operational – O • Letter of Credit/Bank – L • Purchase Order – P • Customs Entry – U • Transportation - T

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
CurrRateCre	Currency Rate Create Message	CurrRateDesc.xsd
CurrRateCre	Currency Rate Modify Message	CurrRateDesc.xsd

Customer Order Fulfillment Subscription API

Functional Area

Customer Order Fulfillment

Business Overview

Merchandising provides an interface to process Customer Order Fulfillment requests from an external order management system (OMS). If the system option `OMS_IND = 'Y'`, then Merchandising expects to receive customer orders through this API. If the system option `PERSIST_CUSTOMER_DATA_IND = 'N'`, personal information will not be stored in the customer order table in Merchandising.

Merchandising supports two integration methods for processing Customer Order Fulfillment messages from OMS – either through RIB or Web Service. At implementation time, clients should decide on either one or the other integration method, but not both. The same core logic is used to validate and persist customer orders to Merchandising tables.

- In a RIB implementation, Merchandising subscribes to Customer Order Fulfillment messages. When a customer order is created, or partially or fully cancelled, the customer order information is sent from the Order Management System (OMS) to the RIB. Merchandising subscribes to the customer order information as published from the RIB and places the information onto Merchandising tables.
- In a web service implementation, Merchandising exposes a FulfillOrder web service to create or cancel a customer order in Merchandising. OMS will invoke the service with customer order details to place the information on Merchandising tables. See [Customer Order Services](#) in the [SOAP Web Services](#) section of this document for more details on this method.

The Customer Order Fulfillment message staged will go through a process of validation. Records that pass validation will create new customer order records. If any validation error occurs, transaction will be rolled back, and no customer orders will be created.

There are two scenarios where a customer order fulfillment request cannot be created in Merchandising:

1. Due to data validation errors (for example, invalid item).
2. Due to 'No Inventory' - There is not enough inventory available at the source location or item is not ranged or inactive at the source location, or item is not supplied by the supplier (in a PO scenario).

Other Notes

1. Non-stockholding franchise stores cannot be part of a fulfillment order, either as a sourcing location or as a fulfillment location.
2. Catch weight and transformable sellable items are not supported in this integration. To sell items that can vary by weight, like bananas through online channels, setup should be done as a regular (non-catch weight) item with a unit cost and standard UOM defined in items of eaches.
3. It is assumed that customer orders will be captured in the selling UOM in OMS, but that all transactions will be communicated to Merchandising in standard UOM.
4. If the same customer order fulfillment request is sent for a different item or for an existing item but with a different item line number, the existing PO or transfer will be updated.

The Customer Order Fulfillment messages contain information such as delivery type, source type, and destination type. Based on these, the system should proceed to create a Purchase Order, Transfer or Inventory Reservation. The table below shows the customer order scenarios for the combination of delivery type, source type and destination type:

Scenario #	Source Location	Fulfillment Location	Delivery Type	Transaction created
1	Warehouse	Store	Pickup in Store	Virtual WH to Physical Store Transfer + Reservation FulfilOrdDesc will contain: 1 st leg: source_loc_type = 'WH', fulfill_loc_type = 'S' 2 nd leg: source_loc_type = NULL, fulfill_loc_type = 'S'
2	Warehouse	Store	Ship to Customer	Virtual WH to Physical Store Transfer + Reservation FulfilOrdDesc will contain: 1 st leg: source_loc_type = 'WH', fulfill_loc_type = 'S' 2 nd leg: source_loc_type = NULL, fulfill_loc_type = 'S'
3	Store A	Store B	Pickup in Store	Physical Store to Physical Store Transfer + Reservation FulfilOrdDesc will contain: 1 st leg: source_loc_type = 'ST', fulfill_loc_type = 'S' 2 nd leg: source_loc_type = NULL, fulfill_loc_type = 'S'
4	Store A	Store B	Ship to Customer	Physical Store to Physical Store Transfer + Reservation FulfilOrdDesc will contain: 1 st leg: source_loc_type = 'ST', fulfill_loc_type = 'S' 2 nd leg: source_loc_type = NULL, fulfill_loc_type = 'S'
5	NULL	Store	Pickup in Store	Reservation FulfilOrdDesc will contain: Single-leg: source_loc_type = NULL, fulfill_loc_type = 'S'
6	NULL	Store	Ship to Customer	Reservation FulfilOrdDesc will contain: Single-leg: source_loc_type = NULL, fulfill_loc_type = 'S'
7	NULL	Warehouse	Ship to Customer	Virtual WH to Virtual Store Transfer FulfilOrdDesc will contain: Single-leg: source_loc_type = 'WH', fulfill_loc_type = 'V'
8	Vendor	Store	Pickup in Store	Purchase Order to Physical Store + Reservation FulfilOrdDesc will contain: 1 st leg: source_loc_type = 'SU', fulfill_loc_type = 'S' 2 nd leg: source_loc_type = NULL, fulfill_loc_type = 'S'

Scenario #	Source Location	Fulfillment Location	Delivery Type	Transaction created
9	Vendor	Store	Ship to Customer	Purchase Order to Physical Store+ Reservation FulfilOrdDesc will contain: 1 st leg: source_loc_type = 'SU', fulfill_loc_type = 'S' 2 nd leg: source_loc_type = NULL, fulfill_loc_type = 'S'
10	NULL	Vendor	Ship to Customer	Purchase Order to Virtual Store FulfilOrdDesc will contain: Single-leg: source_loc_type = 'SU', fulfill_loc_type = 'V'

The customer order subscription API supports create and cancel operations using the following message types belonging to the fulfilord message family:

- fulfilordapprdel – used by Merchandising to cancel customer orders.
- fulfilordreqdel – used by SIOCS to request a customer order cancellation. This message type is used only by SIOCS and is ignored by Merchandising.
- fulfilordpocre – used to create purchase orders because of customer order fulfillment requests.
- fulfilordtsfcre – used to create transfers because of customer order fulfillment requests.
- fulfilordstdlvcre – used to perform inventory reservation because of customer order fulfillment requests.

The format for creating customer order fulfillment requests is shown below.

Table 3-5 Create Customer Order Fulfillment

Message Element	Required?	Notes
Customer Order Number	Yes	This field contains the master customer order number from OMS.
Fulfillment Order Number	Yes	This field contains the unique number from OMS related to the fulfillment details. A single customer order can have one or more fulfillment orders.
Source Location Type	No	This field contains the location type where the fulfillment order will be sourced. This could be either SU for supplier, ST for store, or WH for warehouse. This would only be populated for vendor, warehouse, or multi-site fulfillment orders. Both source location type and source location ID must be populated, or both should be NULL.

Table 3-5 (Cont.) Create Customer Order Fulfillment

Message Element	Required?	Notes
Source Location ID	No	This field contains the store, supplier or warehouse number associated with sourcing the fulfillment order. This would only be populated for vendor, warehouse, or multi-site fulfillment orders. Both source location type and source location ID must be populated, or both should be NULL. If the source location type is supplier, this must be a valid supplier site in Merchandising. If the source location type is store, this should be a valid stockholding customer orderable company or franchise store. If the source location type is warehouse, it can be a valid physical or virtual warehouse.
Fulfillment Location Type	No	This field indicates the location type associated with fulfilling the fulfillment order. This would be either S (for physical store) or V (for virtual store).
Fulfillment Location ID	Yes	This field indicates the store number associated with fulfilling the fulfillment order. This should always be populated with a virtual or physical store number. The fulfillment location ID should be different from the source location ID.
Partial Delivery Indicator	Yes	This field indicates if the fulfillment order can be picked and shipped partially (N) or if it should be shipped only when complete (Y).
Delivery Type	No	This field indicates the fulfillment method - ship to customer or store pickup. Valid values are S (ship direct) and C (customer pickup).
Carrier Code	No	This field indicates the carrier the order is to be shipped with, if specified on the fulfillment order.
Carrier Service Code	No	This field indicates the method that was selected for shipping by the customer placing the order (e.g., Standard Shipping, Overnight, etc.). Valid values are from code type CSVN.
Consumer Delivery Date	Yes	This field indicates the desired date the delivery is required by the customer. This will be in GMT time.
Consumer Delivery Time	No	This field indicates the desired time the delivery is required by the customer. This will be in GMT time. Both Delivery Date and Delivery Time should be populated, or both should be NULL.
Delivery Charges	No	This field contains the delivery charges on drop ship. Used for Brazil implementations only. This value should be greater than 0.
Delivery Charges Currency	No	This field contains the currency of the delivery charges. It must be a valid currency code in Merchandising. Delivery Charges and Delivery Charges Currency must both be populated, or both should be NULL.
Comments	No	This field contains any comments sent by OMS about the order.
Customer Order Fulfillment Details	Yes	Child Node

Table 3-5 (Cont.) Create Customer Order Fulfillment

Message Element	Required?	Notes
Fulfillment Order Customer Details	Conditional	Child node
Custom Flex Attributes	No	Child node Not used in Merchandising
Order Placed Store	No	This field indicates the store number associated with the location that the customer order was placed. For on-line orders this would contain the virtual store number associated with the on-line store. For orders captured in a physical store this would contain the store number for the physical store.

Table 3-6 Create Customer Order Fulfillment Details

Message Element	Required?	Notes
Item	Yes	This field indicates the item ordered by the customer.
Reference Item	No	This field indicates the reference item (barcode) ordered by the customer. This is supported for vendor drop-ships orders only.
Order Quantity SUOM	Yes	This field contains the quantity that was ordered by the customer in item's standard unit of measure.
Standard UOM	No	This field contains the item's standard unit of measure.
Transaction UOM	No	This field indicates the original transaction unit of measure the order was placed in.
Substitute Indicator	Yes	This field indicates if substitutes are allowed on a fulfillment order. This will only be used by orders passed to SIOCS.
Unit Retail	No	This field contains the unit sales retail of item on the fulfillment order. This will only be used by Brazil orders in case of warehouse fulfillment or vendor sourced POs shipped directly to the customer. It is needed for sales nota fiscal generation.
Retail Currency	No	This field contains the currency for the unit retail. This will only be used by Brazil orders in case of warehouse fulfillment or vendor sourced POs shipped directly to the customer. It is needed for sales nota fiscal generation.
Comments	No	This field is used to indicate any special instructions for the item, such as services (monograms, engrave, etc.).
Item Line Number	No	This field contains the detail item line number on the fulfillment order.

Only one customer order detail record should exist for each fulfillment order. If the system options value Retain Customer Data is set to Y and the order is fulfilled from the supplier or warehouse with the fulfillment location type as virtual store, then this node is required. If Retain Customer Data is N, then only the customer number will be retained.

Table 3-7 Create Fulfillment Order Customer Details

Message Element	Required?	Notes
Customer Number	Yes	This field indicates the number that uniquely identifies the customer in OMS.
Delivery First Name	No	This field contains the first name of the contact person at the delivery address on the fulfillment order.
Delivery Phonetic First	No	This field contains the phonetic first name of the contact person at the delivery address on the fulfillment order.
Delivery Last Name	No	This field contains the last name of the contact person at the delivery address on the fulfillment order.
Delivery Phonetic Last	No	This field contains the phonetic last name of the contact person at the delivery address on the fulfillment order.
Delivery Preferred Name	No	This field contains the preferred name of the contact person at the delivery address on the fulfillment order.
Deliver Company Name	No	This field contains the company name of the contact person at the delivery address on the fulfillment order.
Deliver Address 1	No	This field contains the first line of the delivery address of the customer.
Delivery Address 2	No	This field contains the second line of the delivery address of the customer.
Delivery Address 3	No	This field contains the third line of the delivery address of the customer.
Delivery Country	No	This field contains the county portion of the delivery address.
Delivery City	No	This field contains the city portion of the delivery address.
Delivery State	No	This field contains the state portion of the delivery address.
Delivery Country ID	No	This field contains the country portion of the delivery address.
Delivery Post	No	This field contains the postal code portion of the delivery address.
Delivery Jurisdiction	No	This field identifies the jurisdiction code of the delivery country-state relationship.
Delivery Phone	No	This field contains the delivery phone number.
Delivery E-mail	No	This field contains the delivery email.
Bill First Name	No	This field contains the first name of the customer to be billed for this fulfillment order.
Bill Phonetic First	No	This field contains the phonetic first name of the customer to be billed for this fulfillment order.
Bill Last Name	No	This field contains the last name of the customer to be billed for this fulfillment order.

Table 3-7 (Cont.) Create Fulfillment Order Customer Details

Message Element	Required?	Notes
Bill Preferred Name	No	This field contains the preferred name of the customer to be billed for this fulfillment order.
Bill Company Name	No	This field contains the company name of the customer to be billed for this fulfillment order.
Bill Address 1	No	This field contains the first line of the billing address of the customer.
Bill Address 2	No	This field contains the second line of the billing address of the customer.
Bill Address 3	No	This field contains the third line of the billing address of the customer.
Bill Country	No	This field contains the county portion of the billing address.
Bill City	No	This field contains the city portion of the billing address.
Bill State	No	This field contains the state portion of the billing address.
Bill Country ID	No	This field contains the country portion of the billing address.
Bill Post	No	This field contains the postal code portion of the billing address.
Bill Jurisdiction	No	This field identifies the jurisdiction code for the billing country-state relationship.
Bill Phone	No	This field contains the billing phone number.
Bill E-mail	No	This field contains the billing e-mail address.

Table 3-8 Cancel Customer Order Fulfillment

Message Element	Required?	Notes
Customer Order Number	Yes	Holds the master customer order number from OMS.
Fulfillment Order Number	Yes	This field contains the unique number from OMS related to the fulfillment details. A single customer order can have one or more fulfillment orders.
Source Location Type	No	This field contains the original sourcing location type for the fulfillment order being cancelled. This would be either SU for supplier, ST for store, or WH for warehouse. This would only be populated for vendor, warehouse, or multi-site fulfillment orders.
Source Location ID	No	This field contains the store, supplier or warehouse number associated with sourcing the fulfillment order being cancelled. This would only be populated for vendor, warehouse, or multi-site fulfillment orders.
Fulfillment Location Type	No	This field indicates the location type associated with fulfilling the fulfillment order being cancelled. This would be either S (for physical store) or V (for virtual store).

Table 3-8 (Cont.) Cancel Customer Order Fulfillment

Message Element	Required?	Notes
Fulfillment Location ID	Yes	This field indicates the store number associated with fulfilling the fulfillment order being cancelled.
Fulfillment Order Detail	Yes	Child node

Table 3-9 Cancel Customer Order Fulfillment Detail

Message Element	Required?	Notes
Item	Yes	This field contains the item cancelled by the customer.
Reference Item	No	This field contains the reference item cancelled by the customer. This is supported for vendor drop-ships orders only.
Cancel Quantity SUOM	Yes	This field contains the quantity that should be cancelled from the order in item's standard unit of measure.
Standard UOM	No	This field contains the item's standard unit of measure.
Transaction UOM	No	This field indicates the original transaction unit of measure the order is placed in.
Item Line Number	No	This field indicates the detail item line number on the fulfillment order that is being cancelled.

In a RIB implementation, once fulfillment create messages are processed in Merchandising, Merchandising will publish to the RIB a customer order fulfillment confirmation message with a message type of `fulfillordcfmcre` via the customer order fulfillment confirmation publishing API. Confirmation messages will only be sent for customer order fulfillment create requests that result in creating purchase orders and transfers in Merchandising. It will not be sent for cancel requests, nor for customer order fulfillment requests that result in inventory reservation.

- If a customer order is partially fulfilled, a confirmation message with status 'P' will be sent with details of fulfilled order quantity.
- If a customer order is not fulfilled at all due to unavailable inventory, a confirmation message with status 'X' will be sent without any details.
- If a customer order is fulfilled completely due to available inventory, a confirmation message with status 'C' will be sent with details for the fulfilled order quantity

See [Customer Order Fulfillment Confirmation Publication API](#) for more details on the confirmation message sent.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
Fulfilordapprdel	Fulfilment Cancel Message	FulfilOrdRef.xsd
Fulfilordreqdel	Fulfilment Cancel Request Message	FulfilOrdRef.xsd
Fulfilordpocre	Fulfilment PO Create Message	FulfilOrdDesc.xsd
Fulfilordtsfcre	Fulfilment Transfer Create Message	FulfilOrdDesc.xsd
Fulfilordstdlvcre	Fulfilment Store Delivery Create Message	FulfilOrdDesc.xsd

Differentiator Group Subscription API

This section describes the Diff group subscription API.

Functional Area

Items

Business Overview

This API allows external systems to create, edit, and delete differentiator groups within Merchandising. Diff ID details can be added, edited, or deleted within the diff group message. When creating a new diff group, diff ID must be included, but they can also be passed in with their own specific message type. Diff ID detail create and modify messages must also include the diff group record.

Creating Diff Groups

When a new differentiator group is created, this API will first validate that all required fields are present in the message. When creating a new differentiator group at least one detail line must also be included in the message. After that, business level validation on the input information will be performed. The tables below summarize the validation.

Table 3-10 Diff Group Header

Message Element	Required?	Notes
Differentiator Group Identifier	Yes	This field contains the unique differentiator group identifier.
Differentiator Type	Yes	This field contains the differentiator type, such as C for color. Must exist as a valid diff type in Merchandising.
Differentiator Group Description	Yes	This field contains the description of the differentiator group.

Table 3-10 (Cont.) Diff Group Header

Message Element	Required?	Notes
Create Date Time	No	This field contains the date and time the differentiator group was created. If it is not populated on the subscription message it will be defaulted to the time of creation in Merchandising.
Differentiator Group Detail	Conditional	Child node This is required in a diff group detail create and modify message.

Table 3-11 Diff Group Detail

Message Element	Required?	Notes
Differentiator ID	Yes	This field contains the identifier of the differentiator contained within the differentiator group. This id must be unique within the diff group and must already exist in Merchandising.
Display Seq	No	This field contains the order in which the diff ID should appear within the differentiator group, when displayed on-line.
Create Date Time	No	This field contains the date/time the differentiator ID was added to the differentiator group. If it is not populated on the subscription message it will be defaulted to the time of creation in Merchandising.

Updating Diff Groups

When updating a differentiator group, the group ID must already be present in the Merchandising. Changes can be sent for header level updates or detail level updates. If the changes are at the header level, then all of the required header level information needs to be included in the update, similar to that described above for creating a new differentiator. However, the diff details should not be included in a header only update. Fields that can be updated at the header level using this API include:

- Differentiator type
- Differentiator group description

For updating the record, the diff group ID is required in the header level and diff ID is required at the detail level. Fields that can be updated at the detail level using this API include:

- Display seq

Deleting Differentiator Groups

If you are deleting a differentiator detail in the differentiator group or deleting the whole differentiator group, then the API will validate that the differentiator group is valid and that it is not associated with any items or diff ranges. If you are deleting the whole differentiator group, then no details should be included in the message. If you are deleting a detail record on the differentiator, then validation will be done to ensure that the diff id exists on the differentiator group.

Table 3-12 Diff Group Header

Message Element	Required?	Notes
Differentiator Group Identifier	Yes	This field contains the diff group to be deleted.
Differentiator Group Detail	Conditional	Child node This is required in a diff group detail delete message.

Table 3-13 Diff Group Detail

Message Element	Required?	Notes
Differentiator ID	Yes	This field contains the identifier of the differentiator that will be deleted, that is contained within the differentiator group. This id must be unique within the diff group and must already exist in Merchandising.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
XDiffGrpDes	Diff Group Create and Modify	XDiffGrpDesc.xsd
XDiffGrpDtl	Diff Group Detail Create and Modify	XDiffGrpDesc.xsd
XDiffGrpRef	Diff Group Delete	XDiffGrpRef.xsd
XDiffGrpDtlRef	Diff Group Detail Delete	XDiffGrpRef.xsd

Differentiator Subscription API

This section describes the Diff ID subscription API.

Functional Area

Items

Business Overview

This API subscribes to differentiators from external systems to create, update or delete differentiators in Merchandising. This subscription API provides a means to keep Merchandising in sync with an external system. These transactions are performed immediately upon message receipt so success or failure can be communicated to the calling application.

Creating Differentiators

When a new differentiator is created, this API will first validate that all required fields are present in the message. After that, business level validation on the input information will be performed. The business validation:

- verifies the diff id does not contain white space or underscores
- verifies if diff id is not already present as a diff id or diff group id
- verifies the diff type is a valid value on the code detail table under code type DIFF

If all the validations are met, the differentiator in the message data is created in Merchandising.

The format for creating and updating differentiators is shown below.

Differentiator

Message Element	Required?	Notes
Differentiator ID	Yes	This field contains the unique identifier of the differentiator. This must be a valid value under the code type DIFF.
Differentiator Type	Yes	This field contains the identifier of the differentiator type. This value must be a valid diff_type in merchandising.
Differentiator Description	Yes	This field contains the description of the differentiator.
Industry Code	No	This field contains the unique reference number which represents all possible combinations of sizes according to the National Retail Federation.
Industry Subgroup	No	This field contains a sub-grouping code used by industry standards to further identify the differentiator.
Create Date Time	No	This contains the date and time the differentiator was created. If this field is not populated on the message it will default to the time of creation in merchandising.

Updating Differentiators

When a differentiator is updated, this API will first validate that all required fields are present in the message. After that, business-level validation on the input information will be performed. If all the validations are met, the differentiator in the message data are updated in Merchandising. The message and validation are similar to that of the creating differentiators.

Deleting Differentiators

When a differentiator is deleted, this API will first validate that all required fields are present in the message. After that, business level validation on the input information will be performed. The business validation:

- verifies if diff id to be deleted exists in Merchandising

If all the validations are met, the differentiator in the message data is deleted from Merchandising.

Differentiator

Message Element	Required?	Notes
Differentiator ID	Yes	This field contains the unique identifier of the differentiator being deleted.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
xdiffidcre	Differentiator Create	XDiffIDDesc.xsd
xdiffidmod	Differentiator Modify	XDiffIDDesc.xsd
xdiffiddel	Differentiator Delete	XDiffIDRef.xsd

DSD Deals Subscription API

This section describes the DSD deals subscription API.

Functional Area

Deals

Business Overview

Direct store delivery (DSD) is the delivery of merchandise and/or services to a store without the benefit of a pre-approved purchase order, such as when the supplier drops off merchandise

directly in the retailer's store. This process is common in convenience and grocery stores, where suppliers routinely come to restock merchandise. In these cases, the invoice may be given to the store (as opposed to sent to corporate), and the invoice may or may not be paid for out of the register.

Merchandising subscribes to DSD messages from the RIB. These messages notify Merchandising of a direct store delivery transaction at a location so that it may record the purchase order and account for it in the store's inventory. Merchandising also subscribes to DSD deals messages for deals applicable to any DSD order and performs the following functionalities as necessary:

- Applies any deals to a DSD purchase order if the deals indicator in the message is set to Y
- Creates a shipment
- Receives a shipment.
- Creates an invoice

Table 3-14 DSD Deals

Message Element	Required?	Notes
Order Number	Yes	This field contains the order number.
Supplier	Yes	This field contains the supplier number for the deal being created for this order, not the supplier site.
Store	Yes	This field contains the location the shipment will be delivered to.
Department	No	The department in which all the items on the order belong.
Currency Code	Yes	This field contains a code identifying the currency the supplier uses for business transactions.
Paid Indicator	Yes	This field indicates if the invoice has already been paid. Valid values are Y (invoice has already been paid) or N (invoice should be paid in accounts payable system).
External Reference Number	No	If the invoice indicator is Y (invoice has been created), the external reference number, proof of delivery number, or payment reference number must be provided.
Proof of Delivery Number	No	This field contains the proof of delivery or service number given at the time of receipt at the store. This field will also be included when the invoice is interfaced through Sales Audit.
Payment Reference Number	No	This field contains the reference number attached to the invoice payment, used when the invoice is paid from the POS system and interfaced through Sales Audit.
Payment Date	No	This field contains the date when the invoice was paid from the POS system. This field will be populated when the invoice is interfaced through Sales Audit.
Deals Indicator	Yes	This field indicates whether deals need to be applied to the order or not. Valid values are Yes (Y) and No (N).
Shipment	Yes	This field contains the corresponding shipment for the order that was applied by the deal.
Invoice Id	Yes	This field contains the invoice number for the purchase order of this deal.

Table 3-14 (Cont.) DSD Deals

Message Element	Required?	Notes
Invoice Indicator	Yes	This field indicates whether an invoice was created for this receipt by the supplier. Valid values are Yes (Y) and No (N).
Receipt Date	Yes	This field contains the date of the receipt.
Quantity Sum	Yes	This field contains the total quantity for the invoice.
Cost Sum	Yes	This field contains the total merchandise cost for the invoice. This field will be held in the invoice currency.
External Receipt Number	Yes	This field holds the external transaction sequence number for the receipt.

Note

Invoices are not created if Invoice Matching is not running, if the invoice indicator or paid indicator from the message is N, or if paid indicator on the message is Y and Sales Audit is not running.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please see RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
dsddealscre	DSD Deals Create Message	DSDDealsDesc.xsd

DSD Receipt Subscription API

This section describes the DSD receipt subscription API.

Functional Area

Purchase Orders

Business Overview

Direct store delivery (DSD) is the delivery of merchandise to and/or the performance of services in a store without the benefit of a pre-approved purchase order. When the delivery or service occurs, the store inventory system (for example, SIOCS) informs Merchandising of the receipt, which also generates the purchase order at the same time. The receipt can include both the merchandise item as well as the non-merchandise information associated with the order. This information works in conjunction with payment details sent through Sales Audit if payment was made a part of the delivery or service at the store.

Table 3-15 DSD Receipt Header

Message Element	Required?	Notes
Supplier	Yes	This field contains the unique identifying number for a supplier for the receipt.
Origin Country ID	Yes	This field contains the identifier of the country from which the item is being sourced.
Store	Yes	This field contains the location where the items were delivered.
Department	Conditional	The department in which all the items on the order belong.
Currency Code	Yes	This field contains a code identifying the currency the supplier uses for business transactions.
Paid Indicator	Yes	This field indicates if the invoice has already been paid. Valid values are Y (invoice has already been paid) or N (invoice should be paid in accounts payable system).
External Reference Number	No	If the invoice indicator is Y (invoice has been created), the external reference number, proof of delivery number, or payment reference number must be provided.
Proof of Delivery Number	No	This field contains the proof of delivery or service number given at the time of receipt at the store. This field will also be included when the invoice is interfaced through Sales Audit.
Payment Reference Number	No	This field contains the reference number attached to the invoice payment, used when the invoice is paid from the POS system and interfaced through Sales Audit.
Payment Date	No	This field contains the date when the invoice was paid from the POS system. This field will be populated when the invoice is interfaced through Sales Audit.
Invoice Indicator	Yes	This field indicates whether an invoice was created for this receipt by the supplier. Valid values are Yes (Y) and No (N).
Deals Indicator	Yes	This field indicates whether deals need to be applied to the order or not. Valid values are Yes (Y) and No (N).
DSD Detail	No	Child node
DSD Non-Merchandise	No	Child node
External Receipt Number	No	This field holds the external transaction sequence number for the receipt.
Receipt Date	No	This field contains the date of the receipt.

Table 3-16 DSD Receipt Detail

Message Element	Required?	Notes
Item	Yes	This field contains the item in the receipt. The item must be an approved item that is at transaction level or above.
Quantity Received	Yes	This field contains the number of items received for the item/shipment combination. The value must be greater than 0.
Unit Cost	No	This field contains the cost of the item from the supplier/ origin country in the supplier's currency.
Weight	No	This field contains the weight of the item in the receipt, if it is a catch weight item.
Weight UOM	No	This field contains the unit of measure of the received weight.
DSD Detail UIN	No	Child node; not used in Merchandising
Inventory Identifier Type	Optional	This fields contains the inventory identifier type, which is passed during DSD receipt message. Valid values are found under the Inventory Identifier Types (IIDT) code type, for e.g. Lot (L), Expiry Date (E), Import Document (D).
Inventory Id	Optional	This fields contains the inventory identifier ID, which is provided on DSD receipt transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Table 3-17 DSD Non-Merchandise

Message Element	Required?	Notes
Non-Merchandise Code	Yes	This field contains the non-merchandising code that will be added to an invoice for services or other non-merchandise costs associated with the order.
Non-Merchandise Amount	Yes	This field contains the amount of the non-merchandise cost that was invoiced. This field will be held in the invoice currency.
VAT Code	No	This field contains the code identifying the VAT rate that should be applied to the non-merchandise amount entered.
Service Performed Indicator	Yes	This field indicates if a service was performed for the non-merchandise cost. Valid values are yes (Y) or no (N).

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
dsdreceiptcre	DSD Receipt Create Message	DSDReceiptDesc.xsd
dsdreceiptmod	DSD Receipt Modify Message	DSDReceiptDesc.xsd

Freight Term Subscription API

This section describes the freight terms subscription API.

Functional Area

Foundation Data

Business Overview

Freight terms are financial arrangement information related to shipping that can be subscribed to by Merchandising from a financial system. Freight terms are the terms for shipping - for example, the freight terms could be a certain percentage of the total cost; a flat fee per order, and so on. After confirming the validity of the records enclosed within the message, Merchandising is updated with the information.

Creating/Updating Freight Terms

When a new freight term is created, this API will first validate that all required fields are present in the message. Required fields are terms, description, enabled flag, and start and end dates. After required field validation, the freight term record in the message will be inserted if the term does not exist. If the freight term exists, then the dates and enabled flag will be updated.

Message Element	Required?	Notes
Freight Terms	Yes	Contains a number that uniquely identifies the freight terms.
Terms Description	Yes	Contains a description of the freight terms used in the system.
Enabled Flag	Yes	Indicates whether the freight terms are valid or invalid within the respective application. Valid values are Y and N.
Active Start Date	Yes	Indicates the date for assigning an active date to the Freight Terms.
Active End Date	Yes	Indicates the date for assigning an inactive date to the Freight Terms.

Error Handling

If an error occurs in this procedure, a call will be placed to a function to build a complete error message. This message together with a status of E is returned to the external system. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
FrtTermCre	Freight Term Create Message	FrtTermDesc.xsd

General Ledger Chart of Accounts Subscription API

This section describes the General Ledger chart of accounts subscription API.

Functional Area

Financial Integration

Business Overview

This API is used when Retail Financial Integration (RFI) is not being used and integration with a financial system is a custom connections or connects with a non-Oracle financials system. When using RFI with an Oracle Financial system, the Chart of Accounts Information in Merchandising and Sales Audit is created through a mapping and COA validation process.

Before Merchandising can publish stock ledger data to an external financial application, it must receive the General Ledger chart of accounts (GLCOA) structure. The chart of accounts is the financial application's debit and credit account segments (for example, company, cost center, account, and so on). These are mapped to the transactions, locations, and product hierarchy in Merchandising when stock ledger data and Sales Audit totals are sent to the General Ledger. In some financial applications, these are known as code combination chart fields. There is also a primary account in some systems known as a CCID that uniquely identifies the combination of segment or chart field values. Upon receipt of GLCOA message data, Merchandising populates the data to the FIF_GL_ACCT table.

Create and Update Chart of Accounts

This message is used to create new chart of account entries, as well as update existing entries. The message payload details are shown below.

Table 3-18 Chart of Accounts Create

Message Element	Required?	Notes
Primary Account	Always	This field denotes the primary account for a chart of accounts.
Attribute 1	Optional	Secondary account information. A value is required if description 1 is supplied.
Attribute 2	Optional	Secondary account information. A value is required if description 2 is supplied.
Attribute 3	Optional	Secondary account information. A value is required if description 3 is supplied.
Attribute 4	Optional	Secondary account information. A value is required if description 4 is supplied.
Attribute 5	Optional	Secondary account information. A value is required if description 5 is supplied.
Attribute 6	Optional	Secondary account information. A value is required if description 6 is supplied.
Attribute 7	Optional	Secondary account information. A value is required if description 7 is supplied.
Attribute 8	Optional	Secondary account information. A value is required if description 8 is supplied.
Attribute 9	Optional	Secondary account information. A value is required if description 9 is supplied.
Attribute 10	Optional	Secondary account information. A value is required if description 10 is supplied.
Attribute 11	Optional	Secondary account information. A value is required if description 11 is supplied.
Attribute 12	Optional	Secondary account information. A value is required if description 12 is supplied.
Attribute 13	Optional	Secondary account information. A value is required if description 13 is supplied.
Attribute 14	Optional	Secondary account information. A value is required if description 14 is supplied.
Attribute 15	Optional	Secondary account information. A value is required if description 15 is supplied.
Description 1	Optional	Description of the attribute 1 field. Required if attribute 1 is supplied.
Description 2	Optional	Description of the attribute 2 field. Required if attribute 2 is supplied.
Description 3	Optional	Description of the attribute 3 field. Required if attribute 3 is supplied.
Description 4	Optional	Description of the attribute 4 field. Required if attribute 4 is supplied.
Description 5	Optional	Description of the attribute 5 field. Required if attribute 5 is supplied.
Description 6	Optional	Description of the attribute 6 field. Required if attribute 6 is supplied.
Description 7	Optional	Description of the attribute 7 field. Required if attribute 7 is supplied.

Table 3-18 (Cont.) Chart of Accounts Create

Message Element	Required?	Notes
Description 8	Optional	Description of the attribute 8 field. Required if attribute 8 is supplied.
Description 9	Optional	Description of the attribute 9 field. Required if attribute 9 is supplied.
Description 10	Optional	Description of the attribute 10 field. Required if attribute 10 is supplied.
Description 11	Optional	Description of the attribute 11 field. Required if attribute 11 is supplied.
Description 12	Optional	Description of the attribute 12 field. Required if attribute 12 is supplied.
Description 13	Optional	Description of the attribute 13 field. Required if attribute 13 is supplied.
Description 14	Optional	Description of the attribute 14 field. Required if attribute 14 is supplied.
Description 15	Optional	Description of the attribute 15 field. Required if attribute 15 is supplied.
Set of Books	Always	Indicates the set of books that these accounts apply to.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
Glcoacre	GL COA Create Message	GLCOADesc.xsd

GL Chart of Account Validation Service

This section describes the GL Chart of Accounts Subscription API.

Functional Area

Financial Integration

Overview

When using Oracle Retail Financials Integration (RFI) to manage General Ledger integration an Oracle financial solution, a validation service is used to ensure that the segment combinations mapped to by Merchandising and Sales Audit users are valid combinations in the General Ledger. This validation is called from Merchandising and Sales Audit when creating General Ledger cross-reference mappings.

Note

This validation is also used by Oracle Retail Invoice Matching.

As part of your implementation, you will need to configure the URL for the service call in the RETAIL_SERVICE_REPORT_URL table for code RAV. For cloud service implementations, configuration of this service call should be done in coordination with the Oracle Cloud Operations team by logging an SR. For more information, see the *RFI Implementation Guide*.

Inbound ASN Subscription API

This section describes the ASNIN subscription API.

Functional Area

Shipping and Receiving

Business Overview

A supplier or consolidator sends an advanced shipping notice (ASN) to Merchandising. Merchandising subscribes to the ASN information and places the information into Merchandising tables depending upon the validity of the records enclosed within the ASN message.

An ASN message consists of a header record, a series of order records, carton records, and item records. For each message, the header, order, and item records are required. The carton portion of the record is optional. If a carton record is present, however, then that carton record must contain items.

The header record contains information about the shipment, such as where it is being shipped and when it is estimated to arrive. The order records identify which purchase orders are associated with the shipment. If the shipment is packed in cartons, carton records will identify which items are in which cartons. The item records contain the items on the shipments, along with the quantity shipped.

The location contained in the ASN represents the expected receiving location for the order. If the location is a non-stockholding store in Merchandising, then the shipment is also automatically received when the ASN is processed. Two types of non-stockholding stores orders are supported in this integration – franchise stores and drop ship customer orders.

Once the ship quantity is matched, an invoice is generated for Invoice Matching.

Note

This message can also be used by stores and warehouses for inbound transfer and allocation shipments. However, for Merchandising, those shipments are all processed as an Outbound ASN.

Creating/Updating ASN

When a new ASN is created or updated, this API will first validate that all required fields are present in the message. Additionally, when creating or updating an ASN at least one detail line must also be included in the message. After that, business-level validation on the input information will be performed. The tables below summarize these two types of validation.

ASN Header

Message Element	Required?	Notes
Schedule Number	Optional	Not used in Merchandising.
Auto Receive	Optional	Not used in Merchandising.
Destination Location	Optional	Contains the location that the shipment will be delivered to. For purchase orders this will always be either a store or a physical warehouse.
Destination Location Type	Optional	This column contains the destination location type of the destination location field. Valid values are 'S' = store, and 'W' = warehouse.
Destination Store Type	Optional	Not used in Merchandising
To Stockholding Indicator	Optional	Not used in Merchandising
Source Location	Optional	Not used in Merchandising
Source Location Type	Optional	Not used in Merchandising
Source Location Store Type	Optional	Not used in Merchandising
From Stockholding Indicator	Optional	Not used in Merchandising
Advance Shipment Number	Always	This column contains the advance shipping notice number associated with the shipment.
Shipment Type	Always	This column is used to determine the ship origin. If it is C, that means it is a carton shipment and the shipment origin in Merchandising is set to 6 (ASN UCC-128). Otherwise, the Merchandising shipment origin will be defaulted to 0 (ASN Shipment).
Container Quantity	Optional	Not used in Merchandising.
Bill of Lading Number	Optional	This column holds the bill of lading number associated with a shipment from the PO receiving process.
Shipment Date	Always	This column contains the date the PO was shipped.
Estimated Arrival Date	Optional	This field contains the estimated arrival date of a vendor PO shipment.
Ship Address 1	Optional	Not used in Merchandising.
Ship Address 2	Optional	Not used in Merchandising.

Message Element	Required?	Notes
Ship Address 3	Optional	Not used in Merchandising.
Ship Address 4	Optional	Not used in Merchandising.
Ship Address 5	Optional	Not used in Merchandising.
Ship City	Optional	Not used in Merchandising.
Ship State	Optional	Not used in Merchandising.
Ship Postal Code	Optional	Not used in Merchandising.
Ship Country	Optional	Not used in Merchandising.
Trailer Number	Optional	Not used in Merchandising.
Seal Number	Optional	Not used in Merchandising.
Carrier Code	Optional	This column contains a code that indicates the carrier that is involved in the shipment.
Carrier Service Code	Optional	Not used in Merchandising.
Vendor Number	Always	This column contains the supplier who will provide the merchandise specified in the order.
Ship payment method	Optional	This column indicates the payment terms for freight charges associated with the order. Valid values are found in code type SHMT.
Comments	Optional	Contains any comments about the shipment.

ASN PO

Message Element	Required?	Notes
Order Number	Always	Identifies the order number which relates to the goods delivered in the shipment.
Document Type	Optional	Not used in Merchandising.
Not After Date	Optional	This column contains the last date that delivery of the order will be accepted.
Comments	Optional	This column contains any comments about the shipment.

ASN Carton Detail

This is an optional node inside the PO details. If the shipment type is C, then the carton details and the items within the carton are validated.

Message Element	Required?	Notes
Final Location	Always	This will be the final destination of the carton. For a cross-dock order this will be the allocation location, otherwise it will be the direct to order location.
Container ID	Always	This column holds the UCC-128 carton number.
Container Weight	Optional	Not used in Merchandising.
Container Length	Optional	Not used in Merchandising.
Container Width	Optional	Not used in Merchandising.
Container Height	Optional	Not used in Merchandising.

Message Element	Required?	Notes
Container Cube	Optional	Not used in Merchandising.
Expedite Flag	Optional	Not used in Merchandising.
In Store Date	Optional	Not used in Merchandising.
Carrier Shipment Number	Optional	Not used in Merchandising.
Tracking Number	Optional	Not used in Merchandising.
Freight Charge	Optional	Not used in Merchandising.
Master Container ID	Optional	Not used in Merchandising.

ASN Item Details

This is a mandatory node inside the PO and the optional Carton Node. For each message, the header, order, and item records are required. The carton portion of the record is optional. If a carton record is present, then that carton record must contain items in it.

Message Element	Required?	Notes
Final Location	Conditional	This column contains the final location for the order. If it is a pre-allocated order that has been pre-marked, then this is required and would contain the allocation location.
Item Number	Optional	This column contains unique identifier for the item. Either VPN, item number, or reference item must be specified in the message.
Unit Quantity	Always	This column contains the quantity of the item that is expected to be received.
Priority Level	Optional	Not used in Merchandising.
VPN	Optional	This column contains the vendor product number used to find the item number. Either VPN, item number, or reference item must be specified in the message.
Order Line Number	Optional	Not used in Merchandising.
Lot Number	Optional	Not used in Merchandising.
Reference Item Number	Optional	The column contains a bar code or reference item. Either VPN, item number, or reference item must be specified in the message.
Distro Number	Optional	Not used in Merchandising.
Consumer Direct Order	Optional	Not used in Merchandising.
Customer Order Number	Optional	Not used in Merchandising.
Fulfill Order Number	Optional	Not used in Merchandising.
Distro Document Type	Optional	Not used in Merchandising.
Container Quantity	Optional	Not used in Merchandising.
Comments	Optional	Not used in Merchandising.

Note

The Universal Identification Number (UIN) Details and Flex Attribute is not used by Merchandising.

Delete ASN

When deleting an ASN, Merchandising validates that the shipment is not in “Received” status. If the shipment is in a valid status, the shipment will be deleted. When deleting the ASN records, if Invoice Matching is integrated, it will populate the staging table for Invoice Matching to know which shipments have been purged.

Message Element	Required?	Notes
Destination Location	Optional	Contains the location that the shipment will be delivered to. For purchase orders this will always be either a store or a physical warehouse.
Destination Location Type	Optional	This column contains the destination location type of the destination location field. Valid values are 'S' = store, and 'W' = warehouse.
Destination Store Type	Optional	Not used in Merchandising
To Stockholding Indicator	Optional	Not used in Merchandising
Source Location	Optional	Not used in Merchandising
Source Location Type	Optional	Not used in Merchandising
Source Store Type	Optional	Not used in Merchandising
Source Stockholding Indicator	Optional	Not used in Merchandising
ASN Number	Always	This column contains the advance shipping notice number associated with the shipment. The delete happens based on the ASN number where the origin of shipping is through ASN shipment or through ASN UCC-128.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
asnincre	ASN in Create	ASNInDesc.xsd
asninmod	ASN in Modification	ASNInDesc.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
asnindel	ASN in Delete	ASNInRef.xsd

Inventory Adjustment Subscription API

This section describes the Inventory Adjustment Subscription API.

Functional Area

Inventory

Business Overview

Merchandising receives requests for inventory adjustments from an integration subsystem through the Inventory Adjustment Subscription API. The requests contain information about the item and location whose inventory is being adjusted, the quantity to adjust, a from and to disposition code, and the reason for the adjustment. Merchandising uses information in these requests to:

- Adjust overall quantities of stock on hand for an item at a location.
- Adjust the availability of item-location quantities based on status.

After initial processing and validation, Merchandising performs the following tasks:

- The item/location is ranged if it does not already exist.
- For total stock on hand adjustments:
 - Stock on hand is updated for the item at the location, for total stock on hand adjustments.
 - Stock adjustment is recorded to the Merchandising transaction level stock ledger
- For status-based adjustments:
 - Quantities by inventory status are adjusted for the item/location combination.
 - Non-sellable quantity is updated for the item/location.
- For both types, an audit trail is created for the inventory adjustment by item, location, inventory status and reason.

Note

An adjustment can impact both total stock on hand and inventory status at the same time.

Inventory Adjustment Transaction Codes

Whenever the status or quantity of inventory changes, Merchandising writes transaction codes to adjust inventory values in the stock ledger. The types of inventory adjustment transaction codes are:

- Tran code 22 - Adjustments where positive and negative adjustments are made to total stock on hand using a reason code with the COGS indicator = N. In this case, a transaction is inserted to the transaction level stock ledger for both the retail and cost value of the adjustment.
- Tran code 23 - Adjustments where positive and negative adjustments are made to total stock on hand using a reason code with the COGS indicator = Y. In this case, a transaction is inserted to the transaction level stock ledger for both the retail and cost value of the adjustment.
- Tran code 25 - Adjustments to inventory status, where inventory is moved to or from an unavailable or non-sellable status.

Other Notes

- One or both of the to disposition and from disposition fields must have values. Both cannot be empty.
- The item must be inventoried and approved.
- If the item is a simple pack catch weight item, then the weight, weight UOM, and Weight Type should all be either defined or NULL. Weight UOM must be of the type Mass.
- The item is a transaction-level or a reference item. When a reference item is passed in, its parent item (the transaction level item) has its inventory adjusted.
- If adjusting a pack at a warehouse, the pack item must have its inventory tracked at the pack level (receive as type = Pack for the item/warehouse).
- If the location is a warehouse, then either a virtual or physical warehouse can be supported. If it is a virtual warehouse, it must be a stockholding warehouse. If it is a physical warehouse, then the adjusted quantity is distributed among the virtual locations of the physical location.

The table below contains the details of the message as well as the validations.

Table 3-19 Inventory Adjustment Header

Message Element	Required?	Notes
Destination ID	Yes	This field contains the location where the inventory adjustment is being made.
Inventory Adjustment Detail	Yes	Child node

Table 3-20 Inventory Adjustment Detail

Message Element	Required?	Notes
Item	Yes	This contains the item for which stock is being adjusted. The item should be an approved inventory item.
Adjustment Reason Code	No	This field contains the reason for inventory adjustment. This field will only have a value for sellable inventory types.

Table 3-20 (Cont.) Inventory Adjustment Detail

Message Element	Required?	Notes
Unit Quantity	Conditional	This field contains the number of units to be added or reduced. The value should not be 0 and should be a whole number if the standard UOM of the item is EA. The value should be null for Weight-only adjustments which is applicable for Catch weight simple packs and catch weight type 3 and 4 component items having standard UOM as EA.
Transshipment Number	No	Not used
From Disposition	Conditional	From Disposition and To Disposition should both have values or both should be NULL.
To Disposition	Conditional	From Disposition and To Disposition should both have values or both should be NULL.
From Trouble Code	No	Not used
To Trouble Code	No	Not used
From WIP Code	No	Not used
To WIP Code	No	Not used
Transaction Code	No	Not used
User ID	Yes	This field contains the name of the user who created the inventory adjustment.
Create Date Time	Yes	This field contains the date and time the inventory adjustment was made.
PO Number	No	This field contains either a PO, Allocation, BOL or Transfer number associated to the inventory adjustment based on the doc_type. The value should exist in Merchandising.
Document Type	No	This field indicates the type of document where the inventory adjustment originated from. Valid values are: • P - Purchase Order • T - Transfer • A - Allocation
Auxiliary Reason Code	No	Not used
Weight	Conditional	This contains the weight of the catch-weight item. Weight, Weight UOM and Weight Type should all have values, or all should be NULL.
Weight UOM	Conditional	This is the unit of measurement for weight. Weight, Weight UOM and Weight Type should all have values, or all should be NULL. This value should belong to the uom_class, MASS.
Unit Cost	No	This field contains the unit cost of the item.
Inventory Adjustment UIN	No	Child node
Weight Type	Conditional	Indicates whether the weight is the Total weight (T) or the Weight per unit (WPU). This field is applicable only for Catch weight simple packs and catch weight type 3 and 4 component items having standard UOM as EA.
Weight Only Adjustment	Conditional	Indicates whether the adjustment is only for the weight (Y) or for both stock and weight (N).

Table 3-20 (Cont.) Inventory Adjustment Detail

Message Element	Required?	Notes
Inventory Identifier Type	Optional	This field contains the inventory identifier type, which is passed during inventory adjustment message. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, include Lot (L), Expiry Date (E), Import Document (D).
Inventory Id	Optional	This fields contains the inventory identifier ID provided on inventory adjustment transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.
comments	Optional	Holds any comments associated with the inventory adjustment.

Table 3-21 Inventory Adjustment UIN

Message Element	Required?	Notes
Unique Identification Number	Yes	This field contains the Universal Identification Number of the item at the location.
Status	Yes	This field contains the status code of the UIN.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
invadjustcre	Inventory Adjustment Create Message	InvAdjustDesc.xsd

Inventory Request Subscription API

This section describes the inventory request subscription API.

Functional Area

Inventory Request Subscription

Business Overview

Merchandising receives requests for inventory using the Inventory Request API, which allows for items to be ordered by the store and fulfilled by the Merchandising. Unlike store order replenishment, Merchandising fulfills inventory requests from the store regardless of replenishment review cycles, delivery dates, and any other factors that may restrict a request from being fulfilled.

For item/store combinations that are on the Store Order type of replenishment in Merchandising, orders will be placed using this API and then the replenishment process builds the recommended order quantity (ROQ) based on the store's requests. Requests that will not be reviewed prior to the date requested by the store are fulfilled through a one-off process (executed real-time through this API) that creates warehouse transfers and/or purchase orders to fulfill the requested quantities.

For item/location combinations that are currently using other methods of replenishment in Merchandising, the store requested quantities will be added on top of the calculated recommended order quantities to increase the overall replenishment. It can also be used for item/store combinations not on replenishment in Merchandising. In these cases, the one-off process described above will be used to create a POs or transfers utilizing attributes defined for the item/location.

Other validation notes:

- Order quantities will be rounded using the store order multiple when an order is created for a warehouse or to the case size if ordering from the supplier.
- Upcharges will always be applied to a transfer, when they can be defaulted.
- Merchandising will validate that all items belong to the same department when department level ordering (supplier) or department level transfers (warehouse) are being used.
- The store must be open for ordering.

Creating Inventory Requests

The table below summarizes these validations applicable for this API.

Message Element	Required?	Notes
Item	Always	The item must be approved, orderable, and inventoried item; it must also be ranged to the location in the inventory request and must be active at that location.
Unit of purchase	Always	Unit of purchase must either be eaches (EA), case (CA), or pallet (PA).
Need date	Always	This is the date that the store needs the item by.
Need quantity	Always	This is the quantity being requested in standard UOM.
Delivery slot	Optional	Valid delivery slots are in the delivery_slot table.
Store	Always	The store must exist as a valid stockholding store in Merchandising.
Request type	Optional	Store order (SO) or Inventory Request (IR) If the request type is SO or blank, then replenishment method should be store order. If the request type is IR, delivery slot should be provided.

Error Handling

If an error occurs in this procedure, a call will be placed to a function to build a complete error message. This message together with a status of E is returned to the external system. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database. This API supports non-fatal error processing. If an error is encountered in one inventory request detail, it will log and return the error to the RIB.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
InvReqCre	Inventory Request Create Message	InvReqDesc.xsd

Item Subscription API

This section describes the item subscription API.

Functional Area

Items

Business Overview

This API subscribes to items from external systems to create, update, or delete items in Merchandising. This subscription API provides a means to not only create, update, and delete items but also create, update, and delete other details of the item structure.

Item messages include the detail nodes for the supplier and supplier/country. If the item is not a non-sellable pack, the item/zone/price node is also required. Optional nodes can be included in the message for supplier/country, supplier/country/location, pack components, item/vat, dimension, UDA, season, image, translations, HTS, assessments, conditional tariff treatments, expenses, related items, required documents, and upcharges.

This API supports the insertion of data into the Item Induction staging tables based on the data sent through Oracle Retail Integration Bus (RIB), without requiring that the data sent to be functionally complete. If a complete set of item data is sent by the external system through this API, then it is possible to automatically trigger the upload of the data directly into Merchandising by setting a flag in the RIB message, such that item information is uploaded without further user interaction.

Items must be created and maintained following a logical hierarchy as outlined by the referential integrity of the item database tables: Item parents before child items; item components before items that are packs; items before item-suppliers; item/suppliers before item/supplier/countries; items before item/locations (a separate API), and so on. Failing to do so results in message failure.

The auto-creation of item children using differentiator records attached to an item parent, as currently occurs using RMS online processes, is not supported in this API.

These transactions are performed immediately upon message receipt so that success or failure can be communicated to the calling application.

Creating Items

When a new item is created, this API validates that all required fields are present in the message. After that, business-level validation on the input information is performed. If all the validations are met, the item in the message data is created in Merchandising. The tables below summarize these two types of validation.

Table 3-22 Header Level Validation

Message Element	Required?	Notes
Item	Always	The unique identifier of the main item on the message.
Item parent	Optional	Must be an existing item in Merchandising.
Item grandparent	Optional	Must be an existing item in Merchandising.
Pack indicator	Optional	Valid values are Y and N. The value Y indicates this item is a pack item.
Item level	Optional	Valid values are 1 (single level or pack item), 2 (child item), and 3 (grandchild item).
Transaction level	Optional	Valid values are 1, 2, and 3.
Differentiator 1	Optional	Must be an existing differentiator or differentiator group in Merchandising. If the item on the message is a parent, this field may contain a differentiator group ID.
Differentiator 2	Optional	same as Differentiator 1.
Differentiator 3	Optional	same as Differentiator 1.
Differentiator 4	Optional	same as Differentiator 1.
Department	Always	Must be an existing department in Merchandising.
Class	Optional	Must be an existing class in the dept in Merchandising.
Subclass	Optional	Must be an existing subclass in the dept/class in Merchandising.
Item description	Always	Contains the complete, long description for the item being created.
Item supplier country location hierarchy level	Optional	Must be a valid organization hierarchy level. Valid values are: A - Area AI - All Internal Finishers AS - All Stores AW - All Warehouses D - District DW - Default Warehouse I - Internal Finisher L - Location Trait PW - Physical Warehouse R - Region S - Store T - Transfer Zone W - Warehouse
Item zone price hierarchy level	Optional	Must be a valid organization hierarchy level. Valid values are chain (C), area (A), region (R), district (D), store (S) or warehouse (W). Must be null if the item is a non-sellable pack.

Table 3-22 (Cont.) Header Level Validation

Message Element	Required?	Notes
Short Description	Optional	Contains a shortened description of the item being created. Generally, this description would be used by other systems where it is problematic to display the full long description (e.g. a POS system, or customer receipt).
Cost zone group id	Optional	Must be an existing cost zone in Merchandising. Must be null if landed cost is not being used in the system or the item is an orderable buyer pack.
Standard UOM	Optional	Must be a valid UOM. Defaulted to Eaches (EA), if not populated in the message.
Store order multiple	Optional	Valid values are Cases (C), Inners (I), and Eaches (E).
Forecast indicator	Optional	Indicates if this item will have a sales forecast created for it by a forecasting module. Valid values are Y and N. Must be N if the item is a pack.
Simple pack indicator	Conditional	Valid values are Y and N. This field is required if the item is a pack.
Contains inner indicator	Always	Valid values are Y and N. This should be Y if the item is a pack and at least one component item is a pack. Otherwise, it should be N.
Sellable indicator	Optional	Indicates if this item will be made available for sale to end consumers. Valid values are Y and N. It should be Y for a non-pack item.
Orderable indicator	Optional	Indicates if this item will be orderable. Valid values are Y and N. It should be Y for a non-pack item.
Pack type	Optional	Valid values are buyer (B) and vendor (V). Must be Null for a non-pack item.
Order as type	Optional	Must be a valid code from code type PARC if populated: E - Eaches P - Pack Must be Null for a non-pack item.
Comments	Optional	Free form text
Create datetime	Optional	Date/time stamp of when the item was created. This value is populated on insert with SYSDATE and cannot be updated.
Status	Optional	Valid values are worksheet (W), submitted (S), approved (A) and deleted (D).
UOM conversion factor	Optional	Conversion factor between an Each and the standard_uom when the standard_uom is not in the quantity class.

Table 3-22 (Cont.) Header Level Validation

Message Element	Required?	Notes
Package size	Optional	Holds the size of the product printed on any packaging. Should be NULL for below transaction level items.
Handling temperature	Optional	Must be a valid code from code type HTMP if populated.
Handling sensitivity	Optional	Must be a valid code from code type HSEN if populated.
Manufacturer's recommended retail	Optional	Manufacturers recommended retail price for the item.
Waste type	Optional	Must contain either: SL - Sales Wastage SP - Spoilage Wastage
Waste percentage	Optional	Must be greater than 0 or less than 100 if populated.
Item number type	Optional	Must be a valid code from code type UPCT if populated (note: new codes for this code type are not supported):EAN13 - EAN/UCC-13EAN13S - EAN/UCC-13 with SupplementEAN8 - EAN/UCC-8ISBN10 - ISBN-10ISBN13 - ISBN-13ITEM - Oracle Retail Item NumberMANL - ManualNDC - NDC/NHRIC - National Drug CodePLU - PLUSCC - SSCC Shipper CartonUCC14 - EAN/UCC-14UPC-A - UCC12UPC-AS - UCC12 with SupplementUPC-E - UCC8UPC-ES - UCC8 with SupplementVPLU - Variable Weight PLU
Catch weight indicator	Optional	Indicates if this item should be weighed upon receipt. Valid values are Y and N.
Constant dimension indicator	Optional	Valid values are Y and N.
Gift wrap indicator	Optional	Valid values are Y and N.
Ship alone indicator	Optional	Valid values are Y and N.
External source system	Optional	Not used by Merchandising.
Size group 1	Optional	Not used by Merchandising.
Size group 2	Optional	Not used by Merchandising.
Size 1	Optional	Not used by Merchandising.
Size 2	Optional	Not used by Merchandising.
Color	Optional	Not used by Merchandising.
System indicator	Optional	Not used by Merchandising.
UPC supplement	Optional	Not used by Merchandising.
UPC type	Optional	Not used by Merchandising.
Primary UPC indicator	Optional	Not used by Merchandising.
Primary replenishment indicator	Optional	Not used by Merchandising.

Table 3-22 (Cont.) Header Level Validation

Message Element	Required?	Notes
Item aggregate indicator	Optional	Used only for a parent level item. Used in conjunction with the Diff 1-4 aggregate indicators to define which diff values should be used in aggregations along with the parent item, for example, to specify a style (parent item) /color (diff) aggregation. Valid values are Y and N.
Differentiator 1 aggregate indicator	Optional	Valid values are Y and N.
Differentiator 2 aggregate indicator	Optional	Valid values are Y and N.
Differentiator 3 aggregate indicator	Optional	Valid values are Y and N.
Differentiator 4 aggregate indicator	Optional	Valid values are Y and N.
Perishable indicator	Optional	Valid values are Y and N.
Notional pack indicator	Optional	Valid values are Y and N.
Stock on Hand inquiry at pack indicator	Optional	Valid values are Y and N.
AIP case type	Optional	Must be one of the following if populated: F - Formal I - Informal
Order type	Optional	Must be one of the following if populated: F - Fixed Weight V - Variable Weight
Sale type	Optional	Must be one of the following if populated: L - Loose Weight V - Variable Weight Each
Catch weight UOM	Optional	UOM for Catchweight Items.
Deposit item type	Optional	Valid values are: E - Contents A - Container Z - Crate T - Returned Item (Empty bottle)
Inventory indicator	Optional	Indicates if inventory will be managed for this item. Valid values are Y and N.
Item transformation indicator	Optional	Valid values are Y and N.
Container item	Conditional	Required if the Deposit item type is E.
Package UOM	Optional	The unit of measure associated with the package size.
Format ID	Optional	Required for Variable PLU item types.
Prefix	Optional	Prefix for variable weight UPCs.
Brand	Optional	Must be an existing brand in Merchandising if populated.

Table 3-22 (Cont.) Header Level Validation

Message Element	Required?	Notes
Product classification	Optional	Must be a valid code from code type PCLA if populated.
Item description secondary	Optional	Secondary descriptions of the item.
Description uppercase	Optional	All uppercase description of the item (same as upper(item_desc)).
Merchandise indicator	Optional	Valid values are Y and N.
Original retail	Optional	The original retail price of the item per unit.
Retail label type	Optional	Must be a valid code from code type RTLT if populated.
Retail label value	Optional	The value associated with the retail label type.
Default waste percentage	Always	Must be NULL for below transactional level items. Must be greater than 0 and less than 100, if populated. For transactional level or above items, this field must be populated if wastage type is SP.
Item service level	Optional	Holds a value that restricts the type of shipment methods that RCOM can select for an item.
Check UDA indicator	Optional	Valid values are Y and N.
Deposit in price per UOM	Optional	Required if deposit item type is E. Valid values are: E - Exclusive of Deposit Amount I - Inclusive of Deposit Amount
Attempt RMS load	Always	Valid values are STG and RMS. The value STG indicates the item will be loaded into induction staging environment as an incomplete item. The value RMS indicates the item is complete and an attempt to create the item in Merchandising will be made. In the case of RMS, the item will be subject to all validations for a completed item.
Long Description	Optional	Contains a long description of the item being created.

A pack item may be created by providing the component item information while creating the item. This detail is required if item is a pack item. Below are the validations:

Table 3-23 Pack Item Validation

Message Element	Required?	Notes
Component item	Always	Must be an existing item in Merchandising.
Pack quantity	Always	Must be an integer if standard UOM is 'EA'. Must be greater than 0.

Additionally, item zone prices may be updated together with the creation of a new item. Below are the validations:

Table 3-24 Item Zone Price Validation

Message Element	Required?	Notes
Hierarchy ID	Always	
Base retail Indicator	Optional	Not used by Merchandising.
Selling unit retail	Optional	Unit retail price in the selling unit of measure.
Selling UOM	Optional	Unit of Measure for the selling unit retail. UOM must be from either the QTY, MASS, VOL, AREA or DIMEN class.
Multi selling UOM	Optional	Unit of Measure for the multi-selling units and retail. UOM must be from either the QTY, MASS, VOL, AREA or DIMEN class.
Country ID	Optional	Valid Country code.
Currency code	Optional	Not used by Merchandising.
Multi units	Optional	Must be populated only if multi-unit retail is populated. Must be greater than 1 if populated.
Multi-unit retail	Optional	Must be greater than 0.

If the above validations pass, then the item is created with the status defined in the message. If the status in the message is approved, then the item is also subjected to a series of approval checks. For an item to be successfully approved, mandatory information, such as supplier, supplier country, component item information (if item is a pack), is required to be passed as part of the item message. If the item cannot be approved, it is not created.

You can also include following information for the item: expenses, conditional tariff treatments, HTS and assessments, tickets, UDA, VAT (for SVAT tax type), Upcharges, Images, Seasons, Related Items, Required documents, and Item supplier country locations. If included, these are created simultaneously with the creation of the item.

Updating Items

Updates can be made to the items that are in Worksheet, Submitted or Approved status. For the update messages, the API validates that the item number included in the message already exists in Merchandising.

Header Level Updates

Only header-level fields need to be provided for header level updates. Any item details included in the message are ignored for a header-level update message. There are certain fields that are not allowed to be updated at the header level, depending on the status and whether these are still provided in the message; appropriate error message will be returned.

Deleting Items

If you are deleting an item, the API will first validate that the item number is valid. The item number is the only required field for a header delete message. If the item does not exist in Merchandising a reject message is returned.

Optionally the item detail records can be deleted by integrating as part of the item delete message.

Table 3-25 Item Delete

Message Element	Required?	Notes
Item	Always	Must be an existing item in Merchandising.
Hierarchy Level	Optional	Not used
Item Country	Optional	Child Node
Item Supplier	Optional	Child Node
Item VAT	Optional	Child Node
Item Image	Optional	Child Node
Item Season	Optional	Child Node
Item UDA	Optional	Child Node
Item Pack	Optional	Not used.
System Indicator	Optional	Not used.
UPC Supplement	Optional	Not used.
Item Header Translation	Optional	Child Node
Item HTS	Optional	Child Node
Item Expenses	Optional	Child Node
Item Ticket	Optional	Child Node
Item Up-Charges	Optional	Child Node
Item Conditional Tariff Treatment	Optional	Child Node
Related Items	Optional	Child Node
Required Documents	Optional	Child Node

Creating an Item Supplier

An item supplier may be created together with the creation of a new item or added to an existing item. Below are the message details and validations:

Message Element	Required?	Notes
Supplier	Always	Must be an existing supplier in Merchandising.
Primary supp indicator	Always	Indicates if this supplier is the primary supplier for the item. An item can only have one primary supplier. Valid values are Y and N.
VPN	Optional	Vendor Part/Product Number. Must not contain the new line, pipe, linefeed or semicolon characters.
Supplier label	Optional	The supplier label for an item (Parent/Child)
Consignment rate	Optional	Used only for consignment items, to specify the rate of consignment (the percentage of the retail price that will be remitted to the supplier).

Message Element	Required?	Notes
Supp discontinue date	Optional	Must be greater than or equal vdate.
Direct ship indicator	Optional	Indicates if this item can be directly shipped to end consumers from this supplier. Valid values are Y and N.
Pallet name	Optional	Must be a valid code from code type PALN if populated: FLA - Flat PAL – Pallet Additional codes can be added to the code type.
Case name	Optional	Must be a valid code from code type CASN if populated: BA - Barrel BBL - Barrel BE - Bundle BG - Bag BI - Bin BJ - Bucket BK - Basket BX - Box CA - Can CON - Container CR - Crate CS - Case CT - Carton PACK - Pack PO – Pot Additional codes can be added to the code type.
Inner name	Always	Must be a valid code from code type INRN if populated: EA - Eaches INR - Inner SCS - Sub-Case SPACK - Sub-Pack Additional codes can be added to the code type.
Primary case size	Conditional	Required for an orderable item that has been configured for Informal (I) case types.
Supplier differentiator 1	Optional	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
Supplier differentiator 2	Optional	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.

Message Element	Required?	Notes
Supplier differentiator 3	Optional	This field contains the third supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
Supplier differentiator 4	Optional	This field contains the fourth supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
Concession rate	Always	Required for items defined as concession items.
Default expense profiles indicator	Optional	Valid values are Y or N. Indicates whether expenses should be defaulted from the Expense Profile or not.

Updating an Item Supplier

To update an item's supplier information, the supplier must exist for the item in Merchandising; otherwise, an error will be returned. All fields identified in the create section above are updateable and will go through the same validation as in the creation of an item supplier.

Deleting an Item Supplier

To update an item's supplier information, the supplier must exist for the item in Merchandising; otherwise, an error will be returned. All fields identified in the create section above are updateable and will go through the same validation as in the creation of an item supplier. Optionally, the item supplier detail records can be deleted by integrating as part of the item supplier delete message.

Table 3-26 Item Supplier Delete

Message Element	Required?	Notes
Supplier	Always	Must be an existing supplier for the item in Merchandising.
Delete Children Indicator	Always	Indicates whether the deletion should be applied to all associated item/supplier records below the item level of the item specified on the deletion message. Valid values are Y and N.
Item Supplier Country	Optional	Child node.
Item Supplier Country of Manufacture	Optional	Child node.
Item Supplier translation	Optional	Child node.

Creating an Item Supplier Country

An item supplier country may be created together with the creation of a new item or added to an existing item supplier. Below are the validations:

Message Element	Required?	Notes
Origin country ID	Always	Must be an existing country in Merchandising.
Primary country indicator	Always	Indicates the primary origin country for the item / supplier. An item/supplier can only have one primary origin country. Valid values are Y and N.
Unit cost	Optional	Contains the unit cost for the item/supplier/origin country. This must be provided if the item is orderable.
Lead time	Optional	Must be between 0 and 9999.
Pickup lead time	Optional	Must be between 0 and 9999.
Minimum order quantity	Optional	Must be greater than 0.
Maximum order quantity	Optional	Must be greater than 0.
Supplier hierarchy level 1	Optional	Can be used to identify the first level of a supplier hierarchy, such as the manufacturer.
Supplier hierarchy level 2	Optional	Can be used to identify the second level of a supplier hierarchy, such as the distributor.
Supplier hierarchy level 3	Optional	Can be used to identify the third level of a supplier hierarchy, such as the wholesaler.
Default UOP	Always	Contains the default Unit of Purchase for the item at the supplier/origin country.
Supplier pack size	Always	Must be greater than 0. Must not be a decimal if Cost UOM class is Qty
Inner pack size	Always	Must be less than supplier pack size and supplier pack size must be a multiple of inner pack size if populated. Must not be decimal if Cost UOM class is Qty
Ti	Always	Contain the number of cased in a tier on a pallet. Must be greater than 0. The pallet size for an item is the supplier pack size x Ti x Hi.
Hi	Always	Contains the number of tiers high for a pallet. Must be greater than 0. The pallet size for an item is the supplier pack size x Ti x Hi.
Cost UOM	Always	Must be an existing Unit of measure in Merchandising.If the UOM Class is QTY, inner pack size and supp pack size cannot be a decimal.
Tolerance type	Conditional	Must be one of the following:A - ActualP - PercentageRequired for a variable weight simple pack catchweight item.

Message Element	Required?	Notes
Minimum tolerance	Conditional	Required if the item is a variable weight simple pack catchweight item. Must be greater than 0. Must be less than 100 for Percentage tolerance type. Must be less than net weight for Actual tolerance type. Must be less than max tolerance.
Maximum tolerance	Optional	Required if the item is a variable weight simple pack catchweight item. Must be greater than 0. Must be less than 100 for Percentage tolerance type. Must be greater than net weight for Actual tolerance type. Must be greater than max tolerance.
Supplier hierarchy type 1	Optional	Used in conjunction with the Supplier Hierarchy Level 1 to define the type, for example Manufacturer.
Supplier hierarchy type 2	Optional	Used in conjunction with the Supplier Hierarchy Level 1 to define the type, for example Distributor.
Supplier hierarchy type 3	Optional	Used in conjunction with the Supplier Hierarchy Level 1 to define the type, for example Wholesaler.
Round level	Optional	Must be one of the following: C - CaseCL - Case/LayerCLP - Case/Layer/PalletL - LayerLP - Layer/PalletP - Pallet Used in conjunction with the rounding percentages to determine when to round up or down to the nearest inner, case, layer, or pallet.
Round to inner percentage	Always	Must be between 0 and 100.
Round to case percentage	Always	Must be between 0 and 100.
Round to layer percentage	Always	Must be between 0 and 100.
Round to pallet percentage	Always	Must be between 0 and 100.
Packing method	Optional	Must be a valid code from code type PKMT if populated: • FLAT - Flat • HANG - Hanging
Default expense profiles indicator	Optional	Valid values are Y or N. Indicates whether expenses should be defaulted from the Expense Profile or not.
Purchase type	Conditional	Required if consignment concession flag is Y • .0 – Owned (default) • 1 - Consignment • 2 - Concession
Calculation basis	Conditional	Required if consignment concession flag is Y and purchase type is either 1 or 2. • C - Cost per Unit • P - Purchase Rate

Message Element	Required?	Notes
Purchase rate	Conditional	Required if purchase type is either 1 or 2 and calculation basis is P. Must be between 0 and 100.
Pricing Control	Optional	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: • R - Retailer • S - Supplier If no value is provided for consignment/concession items, this will be defaulted based on the default pricing control at supplier site level.

Updating an Item Supplier Country

To update item supplier country information, the supplier country must exist for the item in Merchandising; otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of an item supplier country.

Deleting an Item Supplier Country

To delete an item supplier country, the supplier country must exist for the item in Merchandising; otherwise, an error will be returned. Optionally, the item supplier country detail records can be deleted by integrating as part of the item supplier country delete message.

Table 3-27 Item Supplier Country Delete

Message Element	Required?	Notes
Origin Country Id	Always	Must be an existing supplier country for the item in Merchandising.
Item Supplier Country Location	Optional	Child Node
Item Supplier Country Dimension	Optional	Child Node

Creating Item Supplier Manufacturing Country

An item supplier manufacturing country may be created together with the creation of a new item or added to an existing item supplier. Below are the validations:

Message Element	Required?	Notes
Manufacturer country ID	Always	Must be an existing country in Merchandising.
Primary manufacturer country indicator	Optional	Valid values are Y and N.

Updating Item Supplier Manufacturing Country

To update item supplier country information, the supplier manufacturing country must exist for the item in Merchandising; otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of an item supplier manufacturing country.

Deleting Item Supplier Manufacturing Country

To delete an item supplier manufacturing country, the supplier manufacturing country must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-28 Item Supplier Country of Manufacture Delete

Message Element	Required?	Notes
Manufacturer Country Id	Always	Must be an existing supplier manufacturer country for the item in Merchandising.

Creating Item Supplier Country Location

Item supplier country location can be created together with the creation of a new item or added to an existing item supplier country. Records are not required at this level for an item. If provided, the values override those defined at the item/supplier/country level for the specified location(s). Below are the validations:

Message Element	Required?	Notes
Hierarchy ID	Always	Must be NULL when item-supplier-country-location hierarchy level is either All Stores (AS), All Warehouses (AW) or All Internal Finishers (AI). Required if item-supplier-country-location hierarchy level is either Store (S), Default Warehouse (DW), Warehouse (W), Physical Warehouse (PW), Internal Finisher (I), District (D), Region (R), Area (A), Transfer Zone (T) or Location Trait (L). Contains the identifier for the specified location hierarchy level. For example, if the location hierarchy level is S this will contain a store ID.
Unit cost	Optional	Required if item is an owned item or a consignment/concession item with calculation basis as C, when consignment concession flag is set to Y. Required if consignment concession flag is set to either N or D. Must be blank for buyer packs. Must be greater than 0 and less than 9999999999999999.9999 if populated.
Negotiated item cost	Optional	The supplier negotiated item cost based on the location of the item.

Message Element	Required?	Notes
Primary location indicator	Optional	Indicates if the location is the primary for the item/supplier/origin country. Valid values are Y and N.
Pickup lead time	Optional	Must be between 0 and 9999.
Round level	Optional	Must be one of the following: C - CaseCL - Case/LayerCLP - Case/Layer/PalletL - LayerLP - Layer/PalletP - Pallet Used in conjunction with the rounding percentages to determine when to round up or down to the nearest inner, case, layer, or pallet.
Round to case percentage	Optional	Must be between 0 and 100.
Round to layer percentage	Optional	Must be between 0 and 100.
Round to pallet percentage	Optional	Must be between 0 and 100.
Round to inner percentage	Optional	Must be between 0 and 100.
Supplier hierarchy level 1	Optional	Can be used to identify the first level of a supplier hierarchy, such as the manufacturer.
Supplier hierarchy level 2	Optional	Can be used to identify the second level of a supplier hierarchy, such as the distributor.
Supplier hierarchy level 3	Optional	Can be used to identify the third level of a supplier hierarchy, such as the wholesaler.
Cost UOM	Always	Must be an existing Unit of measure in Merchandising.
Purchase type	Conditional	Must be populated when consignment concession flag is Y. 0 - Owned 1 - Consignment 2 - Concession
Calculation basis	Conditional	Must be provided when consignment concession flag is Y and purchase type is either 1 or 2. Must be one of the following: - C - Cost per Unit - P - Purchase Rate
Purchase rate	Conditional	Must be provided if purchase type is either 1 or 2 and calculation basis is P. Must be between 0 and 100.

Message Element	Required?	Notes
Pricing Control	Optional	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: • R - Retailer • S - Supplier If no value is provided for consignment/concession purchase types, this will be defaulted from item/supplier/country of sourcing if purchase type is Consignment/Concession at item/supplier/country of sourcing level. If purchase type is Owned at item/supplier/country of sourcing level, this will be defaulted based on Default Pricing Control at supplier site level.

Updating Item Supplier Country Locations

To update item supplier country location information, the supplier country location must exist for the item in Merchandising; otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of an item supplier country location.

Deleting Item Supplier Country Locations

To delete an item supplier country location, the supplier country location must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-29 Item Supplier Country Location Delete

Message Element	Required?	Notes
Hierarchy Id	Always	Must be an existing location for the item supplier country in Merchandising.

Creating Item Supplier Country Dimensions

Item supplier country dimensions can be created together with the creation of a new item or added to an existing item supplier country. Dimensions are not required for an item. However, if non-standard units of measure will be used for the item there must be a case-type dimension provided. For example, for a selling unit retail to be defined in ounces there must be a dimension defined containing the liquid and volume to be used when converting between a unit and an ounce. Below are the validations:

Message Element	Required?	Notes
Dimension object	Always	Must be one of the following: CA - Case EA - Each IN - Inner PA - Pallet
Tare weight	Optional	Must be greater than zero and less than weight (gross weight).

Message Element	Required?	Notes
Tare type	Optional	Must be one of the following:D - DryW - Wet
Length width and height UOM	Conditional	Must be a valid UOM of the UOM Class DIMEN, if populated.
Length	Optional	Must be provided if dimension object is populated and UOM class is DIMEN.Must be greater than 0.
Width	Optional	Must be provided if dimension object is populated and UOM class is DIMEN.Must be greater than 0.
Dim height	Optional	Height of dim_object measured in units specified in lwh_uom.
Liquid volume	Optional	Required if dimension object is populated and it's UOM class is LVOL.Must be greater than 0, if populated.
Liquid volume UOM	Optional	Must be a valid UOM of the UOM Class LVOL, if populated.
Statistical cube	Optional	Must be greater than 0, if populated.
Weight UOM	Optional	Must be a valid UOM of the UOM Class MASS, if populated.
Weight	Optional	Required if tare type and net weight are populated.Must be less than tare weight, if populated.Must be greater than 0, if populated.
Net weight	Optional	Minimum tolerance value must be less than net weight for an actual tolerance type.Maximum tolerance value must be greater than net weight.Must be greater than 0, if populated.Must be less than or equal to weight (gross weight).
Presentation method	Optional	Must be a valid code from code type PCKT, if populated.

Updating Item Supplier Country Dimensions

To update item supplier country dimension information, the supplier country dimension must exist for the item in Merchandising, otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of an item supplier country dimension.

Deleting Item Supplier Country Dimensions

To delete an item supplier country dimension, the supplier country dimension must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-30 Item Supplier Country Dimension Delete

Message Element	Required?	Notes
Dimension Object	Always	Must be an existing supplier country dimension for the item in Merchandising.

Creating Item VAT

Item VAT (value added tax) may be created together with the creation of a new item or added to an existing item when default tax type for the system is SVAT (Simple VAT). If not populated, item VAT defaults to the corresponding department's VAT information. Below are the validations:

Message Element	Required?	Notes
VAT Type	Always	Valid values for are Cost (C), Retail (R) and Both (B).
VAT Region	Always	Must be an existing VAT Region in Merchandising, if populated
VAT Code	Always	Must be an existing VAT code in Merchandising, if populated
Active Date	Always	Must not be earlier than or equal to vdate.
Reverse VAT Indicator	Always	Valid values are Y and N.

Deleting Item VAT

In order to delete an item VAT, the VAT information must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-31 Item VAT Delete

Message Element	Required?	Notes
VAT Type	Always	Must be an existing VAT type for the VAT Item in Merchandising.
VAT Region	Always	Must be an existing VAT region for the VAT Item in Merchandising.
VAT Code	Always	Must be an existing VAT code for the VAT Item in Merchandising.
Active Date	Always	Must be an existing active date for the VAT Item in Merchandising.

Creating Item UDA

Item UDA (user defined attribute) of type date/freeform text/list of value can be created together with the creation of a new item or added to an existing item. Below are the validations:

Message Element	Required?	Notes
UDA ID	Always	Must be an existing UDA in Merchandising for the given display type.
Display type	Always	Must contain one of the following to indicate the type of UDA: • DT – Date • LV – List of Values • FF – Free Form
UDA date	Optional	Must be populated for 'DT' display type.
UDA value	Optional	Must be populated for 'LV' display type.
UDA text	Optional	Must be populated for 'FF' display type.
New UDA date	Optional	Must not already be present in Merchandising for the given item UDA combination. Must not be same as the existing value for the given item UDA combination.
New UDA value	Optional	Must not already be present in Merchandising for the given item UDA combination. Must not be same as the existing value for the given item UDA combination.
New UDA text	Optional	Must not already be present in Merchandising for the given item UDA combination. Must not be same as the existing value for the given item UDA combination.
UDA Free Form Text Translations	Optional	Child Node

Updating Item UDA

To update an item UDA of type date/freeform text/list of values, the UDA must exist for the item in Merchandising; otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of an item UDA.

Deleting Item UDA

To delete an item UDA, the UDA must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-32 Item UDA Delete

Message Element	Required?	Notes
UDA Id	Always	Must be an existing UDA for the Item in Merchandising.
Display Type	Always	Display Type of the UDA. Valid values are: - LV - List of Values - FF - Freeform - DT - Date

Table 3-32 (Cont.) Item UDA Delete

Message Element	Required?	Notes
UDA Value	Optional	This can either be a UDA value (for LV display type) or a UDA date (for DT display type) or a freeform text (for FF display type). This must be an existing value for the Item UDA in Merchandising.
UDA Free Form Text Translations	Optional	Child Node

Creating Item Season

Item Seasons can be created together with the creation of a new item or added to an existing item. Below are the validations:

Message Element	Required?	Notes
Season ID	Always	Must be an existing season in Merchandising.
Phase ID	Always	Must be an existing phase for the given season in Merchandising.
Item season sequence number	Optional	Used to ensure uniqueness. Each season/phase combination for an item should have a unique sequence number.
Diff ID	Optional	Can be used to differentiate an item/season by a diff, for example if the season/phase applies to only a specific color for a style.
Create datetime	Optional	Holds the date time stamp the item/season/phase was created.
Last update datetime	Optional	Holds the date time stamp of the most recent update by the Last update ID.
Last update ID	Optional	Holds the ID of the user who most recently updated this record.
color	Optional	Not used by Merchandising.

Deleting Item Season

In order to delete an item season, the season-phase ID must exist for the item in Merchandising, otherwise, an error will be returned.

Table 3-33 Item Season Delete

Message Element	Required?	Notes
Season ID	Always	Must be an existing season for the Item in Merchandising.
Phase ID	Always	This must be an existing phase for the item season in Merchandising.
Differentiator ID	Optional	This must be a valid group/differentiator ID for the item in Merchandising.

Table 3-33 (Cont.) Item Season Delete

Message Element	Required?	Notes
Color	Optional	Not used.

Creating Item Images

Item images can be created together with the creation of a new item or added to an existing item. Below are the validations:

Message Element	Required?	Notes
Image name	Always	Contains the image file name (e.g. image.jpg).
Image address	Optional	Contains the image URL path. If not provided the system default will be used.
Image description	Optional	Contains a name for the image that is logical for a user to be able to identify the image.
Create datetime	Optional	May contain the date timestamp the image was added for the item.
Last update datetime	Optional	May contain the date timestamp for when the image was last updated.
Last update ID	Optional	May contain the ID for the user that made the last update.
Image type	Always	Must be a valid code from code type = IITD if populated: • H - High • L - Low • M - Medium • T - Thumbnail
Primary indicator	Always	Valid values are Y and N.
Display priority	Always	Must be greater than 0.

Updating Item Images

To update an item image, the image name must exist for the item in Merchandising; otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of an item image.

Deleting Item Images

To delete an item image, the image must exist for the item in Merchandising; otherwise, an error will be returned.

Optionally the item image translation can be deleted by integrating as part of the item image delete message.

Table 3-34 Item Image Delete

Message Element	Required?	Notes
Image Name	Always	Must be an existing image for the item in Merchandising.
Image Translations	Optional	Child Node

Creating HTS and Assessments

HTS and Assessments may be created together with the creation of a new item or added to an existing item. Below are the validations:

Table 3-35 HTS Validation

Message Element	Required?	Notes
HTS	Always	Contains the HTS code. The specified code must exist in Merchandising.
Import country ID	Always	Contains the country ID where the item is being imported into and the HTS code applies.
Origin country ID	Always	Contains the country ID where the item originated from.
Effect from Date	Always	Contains the first date the HTS code is effective for the item.
Effect to Date	Always	Contains the last date the HTS code is effective for the item.
Clearing zone ID	Optional	Must be an existing clearing zone in Merchandising and the clearing zone must be associated with the import country.
Status	Optional	Valid values are worksheet (W) and approved (A).

Table 3-36 HTS Assessment Validation

Message Element	Required?	Notes
Component ID	Always	Component ID must exist on the ELC components table.
CVB code	Optional	Required if the component rate calculation basis is value (V), otherwise this will be defaulted to NULL. Must be a valid CVB code in Merchandising.
Component rate	Optional	Must be greater than or equal to 0 is populated. Required if CVB Code is populated.
Per count	Optional	Required if CVB Code is populated. Defaulted to NULL if calculation basis is V, else it must be greater than or equal to 0.0001.
Per count UOM	Optional	Required if calculation basis is V - Value. Must be an existing UOM in Merchandising.
Estimated assessment Value	Optional	The calculated value of the components.

Table 3-36 (Cont.) HTS Assessment Validation

Message Element	Required?	Notes
Nominal flag 1	Optional	Valid values are '+', '-' and 'N', if populated. Defaulted to N if not populated.
Nominal flag 2	Optional	Valid values are '+', '-' and 'N', if populated. Defaulted to N if not populated.
Nominal flag 3	Optional	Valid values are '+', '-' and 'N', if populated. Defaulted to N if not populated.
Nominal flag 4	Optional	Valid values are '+', '-' and 'N', if populated. Defaulted to N if not populated.
Nominal flag 5	Optional	Valid values are '+', '-' and 'N', if populated. Defaulted to N if not populated.

Updating HTS and Assessments

To update HTS and assessments, the record to be updated must exist in Merchandising; otherwise, an error will be returned. Status and origin country ID can be updated at the HTS level. For assessments, all fields identified in the create section above except for component ID are updateable and will go through the same validation as in the creation of assessments.

Deleting HTS and Assessments

To delete HTS and assessments, the record to be deleted must exist in Merchandising; otherwise, an error will be returned.

Table 3-37 Item HTS Delete

Message Element	Required?	Notes
HTS	Always	The specified HTS code must exist for the item in Merchandising.
Import country ID	Always	The specified import country Id must exist for the item/HTS in Merchandising.
Origin country ID	Always	The specified HTS import Id must exist for the item/HTS in Merchandising.
Effect from Date	Always	The specified effective from date must exist for the item/HTS in Merchandising.
Effect to Date	Always	The specified effective to date must exist for the item/HTS in Merchandising.
Item HTS Assessments	Optional	Child Node

Table 3-38 Item HTS Assessments Delete

Message Element	Required?	Notes
Component Id	Always	The specified component must exist for the item HTS assessment in Merchandising.

Creating Expenses

Expenses may be created together with the creation of a new item or added to an existing item that has suppliers defined. Below are the validations:

Message Element	Required?	Notes
Supplier	Always	Item supplier relationship must exist in Merchandising.
Component ID	Always	Component ID must exist on the ELC components table of type Expense (E).
Discharge port	Always	Must be an existing outside location of type 'DP' in Merchandising, if populated.
Origin country ID	Optional	Must be an existing country in Merchandising, if populated.
Lading port	Optional	Must be an existing outside location of type 'LP' in Merchandising, if populated.
Zone ID	Optional	Must be an existing cost zone in Merchandising, if populated.
Zone group ID	Optional	Must be an existing cost zone group in Merchandising, if populated.
Base Expense Indicator	Optional	Valid values are Y and N.
CVB code	Optional	Must be a valid CVB code in Merchandising.
Component rate	Optional	Required if component ID is populated. Must be greater than or equal to 0 if populated.
Per count	Optional	Required if calculation basis is S (Specific). Must be greater than 0, if populated. Defaulted to NULL for Value calculation basis.
Per count UOM	Optional	Required if calculation basis is Specific (S). Must be an existing UOM in Merchandising, if populated. Defaulted to NULL for Value (V) calculation basis.
Component currency	Optional	Must be an existing currency in Merchandising if populated.
Update orders Indicator	Optional	Valid values are Y and N, if populated.
Nominal flag 1	Optional	Valid values are '+', '-' and 'N', if populated.
Nominal flag 2	Optional	Valid values are '+', '-' and 'N', if populated.
Nominal flag 3	Optional	Valid values are '+', '-' and 'N', if populated.
Nominal flag 4	Optional	Valid values are '+', '-' and 'N', if populated.
Nominal flag 5	Optional	Valid values are '+', '-' and 'N', if populated.

Updating Expenses

To update expenses, the item/supplier/component ID must exist in Merchandising; otherwise, an error will be returned. All fields identified in the create section above except for supplier/component ID are updateable and will go through the same validation as in the creation of expenses.

Deleting Expenses

To delete expenses, the item/supplier/component ID/discharge port must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-39 Item Expense Delete

Message Element	Required?	Notes
Supplier	Always	Must be an existing supplier expense for the item in Merchandising.
Component Id	Always	Must be an existing component Id for the item supplier expense in Merchandising.
Discharge port	Always	Must be an existing discharge port location for the item supplier expense in Merchandising.
Origin country ID	Optional	Must be an existing country for item supplier expense in Merchandising, if populated.
Lading port	Optional	Must be an existing lading port location for the item supplier expense in Merchandising, if populated.
Zone ID	Optional	Must be an existing cost zone for item supplier expense in Merchandising, if populated.

Creating Tickets

Tickets may be created together with the creation of a new item or added to an existing item. Below are the validations:

Message Element	Required?	Notes
Ticket type ID	Always	Must be an existing ticket type in Merchandising.
PO print type	Optional	Must be one of the following: - A - On Approval - R - On Receipt This is used to indicate when ticket requests will be created for the item.
Print on price change Indicator	Always	Valid values are Y and N.
Ticket over percentage	Optional	Must be greater than 0, if populated.

Updating Tickets

To update tickets, the item/ticket type ID must exist in Merchandising; otherwise, an error will be returned. All fields identified in the create section above except for ticket type ID are updateable and will go through the same validation as in the creation of item tickets.

Deleting Tickets

To delete item tickets, the item/ticket type ID must exist in Merchandising; otherwise, an error will be returned.

Table 3-40 Item Ticket Delete

Message Element	Required?	Notes
Ticket type ID	Always	Must be an existing ticket type for the item in Merchandising.

Creating Up-charges

Upcharges may be created together with the creation of a new item or added to an existing item. Below are the validations:

Table 3-41 Header Level Validation

Message Element	Required?	Notes
From location type	Always	Valid values are store (S), warehouse (W), physical warehouse (PW), area (A), region (R), country (C).
From location	Optional	Valid location or hierarchy ID.
To location type	Always	Valid values are store (S), warehouse (W), physical warehouse (PW), area (A), region (R), country (C).
To location	Optional	Valid location or hierarchy ID.

Table 3-42 Detail Level Validation

Message Element	Required?	Notes
Component ID	Always	Component ID must exist on the ELC components table.
Component rate	Always	Percentage rate to be applied for the component.
Per count	Optional	Must be greater than 0.
Per count UOM	Optional	Must be an existing UOM in Merchandising.
Up charge group	Always	Must be an existing Upcharge Group in Merchandising.
Component currency	Always	Must be an existing currency in Merchandising.
Transfer allocation default indicator	Optional	Indicator to specify if the component is to be cascaded to transfers and allocations.
Computation Value Base	Optional	Must be an existing CVB code in Merchandising. One and only one from CVB and Cost basis must be provided, if the component ID is of VALUE type

Table 3-42 (Cont.) Detail Level Validation

Message Element	Required?	Notes
Cost Basis	Optional	Must be a valid code from code type UCCB. One and only one from CVB and Cost basis must be provided, if the component ID is of VALUE type
Include in total upcharge indicator	Always	Valid values are 'Y' and 'N'.

Deleting Up-charges

To delete item upcharges, the upcharge or upcharge/component ID must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-43 Item Up-charges Header Delete

Message Element	Required?	Notes
From location type	Always	Valid values are store (S), warehouse (W), physical warehouse (PW), area (A), region (R), country (C).
From location	Optional	Location or hierarchy ID from which upcharges are being deleted.
To location type	Always	Valid values are store (S), warehouse (W), physical warehouse (PW), area (A), region (R), country (C).
To location	Optional	Location or hierarchy ID to which upcharges are being deleted.
Detail Up-charges	Optional	Child node

Table 3-44 Item Up-charges Header Delete

Message Element	Required?	Notes
Component ID	Always	Component ID must exist on the item up-charges in Merchandising.
Transfer allocation default indicator	Optional	Indicator to specify if the component is to be cascaded to transfers and allocations

Updating Up-charges Detail

To update an item upcharge detail, the upcharge/component ID must exist for the item in Merchandising; otherwise, an error will be returned.

Creating Conditional Tariff treatments

Conditional Tariff Treatments can be created with the creation of a new item or added to an existing item. Below are the validations:

Table 3-45 Conditional Tariff Treatment Validation

Message Element	Required?	Notes
Tariff Treatment	Always	Unique code representing a conditional tariff treatment or special program used to determine rates of duty to be paid when importing the associated item. Valid values are present in the Tariff Treatments table (TARIFF_TREATMENT) with the conditional indicator set to Y.
Recalculate Orders	Optional	This would indicate whether orders need to be recalculated or not when adding or removing tariff treatments from an item.

Deleting Conditional Tariff Treatments

In order to delete Conditional tariff treatments, the record to be deleted need to exist in Merchandising, otherwise, no error will be returned.

Table 3-46 Conditional Tariff treatment Delete

Message Element	Required?	Notes
Tariff Treatment	Always	Unique code representing a conditional tariff treatment or special program used to determine rates of duty to be paid when importing the associated item. Valid values are present in the Tariff Treatments table (TARIFF_TREATMENT) with the conditional indicator set to 'Y'.
Recalculate Orders	Optional	This would indicate whether orders need to be recalculated or not when adding or removing tariff treatments from an item.

Creating Item Country

For GTAX tax type, item country can be created together with the creation of a new item or added to an existing item. If not provided, country is set to the default value. Below are the validations:

Message Element	Required?	Notes
Country ID	Always	Must be an existing country in Merchandising.

Deleting Item Country

To delete an item country, the item country must exist in Merchandising; otherwise, an error will be returned.

Table 3-47 Item Country Delete

Message Element	Required?	Notes
Country ID	Always	Must be an existing country for the item in Merchandising.

Creating Related Items

Related Items may be added to the main Item, together with the creation of the item or added to an existing item. Below are the validations:

Table 3-48 Header Level Validation

Message Element	Required?	Notes
Relationship ID	Optional	Unique identifier for each relationship header.
Relationship Name	Always	Name given to the relationship.
Relationship Type	Always	Describes the type of relationship. Values are configured in code_detail table under code_type IREL.
Mandatory Indicator	Optional	Indicates whether the relationship is mandatory.
Related Item Translation	Optional	Child node

Table 3-49 Detail Level Validation

Message Element	Required?	Notes
Related Item	Always	Item ID of the related item.
Priority	Optional	Applicable only with relationship type SUBS. For multiple related substitute items, this column could be used (optionally) to define relative priority.
Start Date	Optional	From this date, related item can be used on transactions.
End Date	Optional	Until this date, the related item can be used on transactions. A value of null means that it is effective forever.

Deleting Related Items

In order to delete related Item, the relationship ID or relationship ID/related Item must exist for the item in Merchandising, otherwise, an error will be returned.

Table 3-50 Related Item Header Delete

Message Element	Required?	Notes
Relationship ID	Always	Must exist in Merchandising.

Table 3-51 Related Item Detail Delete

Message Element	Required?	Notes
Relationship ID	Always	Must exist in Merchandising.
Related Item	Always	Must exist in Merchandising under the given relationship ID.

Updating Related Items

In order to update a related Item relationship, the relationship ID (for header update) or the relationship ID/related item combination (for detail update) must exist for the item in Merchandising; otherwise, an error will be returned. All the updateable fields will go through the same validation as in the creation of related item.

Creating Required Documents Items

Required documents of an Item may be added to the main Item, together with the creation of the item or added to an existing item. Below are the validations:

Table 3-52 Header Level Validation

Message Element	Required?	Notes
Document ID	Always	Unique number used to distinguish the different documents.

Deleting Required Documents Items

In order to delete required document of an Item, the required document and Item must exist for the item in Merchandising; otherwise, an error will be returned.

Table 3-53 Related Item Header Delete

Message Element	Required?	Notes
Document Id	Always	Must exist in Merchandising.

Flex Attributes

If custom flex attributes (CFAS) have been defined for items, at item or item/supplier or item/supplier/country or item/supplier/country/location level, then they can be integrated as part of this API. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute. Flex attributes can only be added to or updated on an item at header and detail levels, but cannot be deleted.

Message Element	Required?	Notes
Name	Always	Holds the name of the attribute.
Value	Optional	Holds the value of the attribute for non-date attributes.
Value Date	Optional	Holds the value of the attribute for date attributes.

Translated Language

If translations have been defined for items, at item or item/image or item/supplier level, then they can be integrated as part of this API. The node of the integration that supports this will accept the language and the description specified in the language. Translations can be added, updated or deleted.

Table 3-54 Item Header Translation Create or Update

Message Element	Required?	Notes
Language	Always	Contains the language in which the translated text is maintained. Must be a valid language in Merchandising.
Short Description	Optional	Contains the translated text for the item short description.
Item Description	Always	Contains the translated text for the item description.
Item Description Secondary	Optional	Contains the translated text for the secondary description of the item.
Long Description	Optional	Contains a long description of the item being created.

Table 3-55 Item Header Translation Delete

Message Element	Required?	Notes
Language	Always	Contains the language of the translated text that needs to be deleted.

Table 3-56 Item Supplier Translation Create or Update

Message Element	Required?	Notes
Language	Always	Contains the language in which the translated text is maintained. Must be a valid language in Merchandising.
Supplier Differentiator 1	Optional	Contains the translated text for supplier differentiator 1.
Supplier Differentiator 2	Optional	Contains the translated text for supplier differentiator 2.
Supplier Differentiator 3	Optional	Contains the translated text for supplier differentiator 3.
Supplier Differentiator 4	Optional	Contains the translated text for supplier differentiator 4.
Supplier Label	Optional	Contains the translated text for the supplier label.

Table 3-57 Item Supplier Translation Delete

Message Element	Required?	Notes
Language	Always	Contains the language of the translated text that needs to be deleted.

Table 3-58 Item Image Translation Create or Update

Message Element	Required?	Notes
Language	Always	Contains the language in which the translated text is maintained. Must be a valid language in Merchandising.
Image Description	Always	Contains the translated text for the item image description.

Table 3-59 Item Image Translation Delete

Message Element	Required?	Notes
Language	Always	Contains the language of the translated text that needs to be deleted.

Table 3-60 Item UDA Free Form Text Translation Create or Update

Message Element	Required?	Notes
Language	Always	This field represents the language in which the translated text is maintained. The language should be from the installed language pack.
UDA Text Desc	Always	This contains the translated value for the UDA text in the selected language.

Table 3-61 Item UDA Free Form Text Translation Delete

Message Element	Required?	Notes
Language	Always	Contains the language of the translated text that needs to be deleted.

Table 3-62 Related Item Translation Create or Update

Message Element	Required?	Notes
Language	Always	Contains the language in which the translated text is maintained. The language should be from the installed language pack.
Relationship Name	Always	Contains the translated value for the relationship name in the selected language.

Table 3-63 Related Item Translation Delete

Message Element	Required?	Notes
Language	Always	Contains the language of the translated text that needs to be deleted.

Table 3-63 (Cont.) Related Item Translation Delete

Message Element	Required?	Notes
Relationship Name	Optional	Contains the translated value for the relationship name in the selected language.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
xitemcre	Item Create Message	XItemDesc.xsd
xitemmod	Item Modify Message	XItemDesc.xsd
xitemdel	Item Delete Message	XItemRef.xsd
xitemsupcre	Item/Supplier Create Message	XItemDesc.xsd
xitemsupmod	Item/Supplier Modify Message	XItemDesc.xsd
xitemsupdel	Item/Supplier Delete Message	XItemRef.xsd
xitemsupctycre	Item/Supplier/Country Create Message	XItemDesc.xsd
xitemsupctymod	Item/Supplier/Country Modify Message	XItemDesc.xsd
xitemsupctydel	Item/Supplier/Country Delete Message	XItemRef.xsd
xiscmfrcre	Item/Supplier/Country of Manufacture Create Message	XItemDesc.xsd
xiscmfmod	Item/Supplier/ Country of Manufacture Modify Message	XItemDesc.xsd
xiscmfrcdel	Item/Supplier/ Country of Manufacture Delete Message	XItemRef.xsd
xiscdimcre	Item/Supplier/Country/Dimension Create Message	XItemDesc.xsd
xiscdimmod	Item/Supplier/Country/Dimension Modify Message	XItemDesc.xsd
xiscdimdel	Item/Supplier/Country/Dimension Delete Message	XItemRef.xsd
xitemvatcre	Item/Vat Create Message	XItemDesc.xsd
xitemvatdel	Item/Vat Delete Message	XItemRef.xsd
xitemctrycre	Item/Country Create Message	XItemCtryDesc.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
xitemctrydel	Item/Country Delete Message	XItemCtryRef.xsd
xitemudacre	Item/UDA Create Message	XItemDesc.xsd
xitemudadel	Item/UDA Delete Message	XItemRef.xsd
xitemimagecre	Item/Image Create Message	XItemDesc.xsd
xitemimagemod	Item/Image Modify Message	XItemDesc.xsd
xitemimagedel	Item/Image Delete Message	XItemRef.xsd
xitemtlcre	Item Master translated language Create Message	XItemDesc.xsd
xitemtlmod	Item Master translated language Modify Message	XItemDesc.xsd
xitemtlidel	Item Master translated language Delete Message	XItemRef.xsd
xitemsuptlcre	Item/Supplier translated language Create Message	XItemSupDesc.xsd
xitemsuptlmod	Item/Supplier translated language Modify Message	XItemSupDesc.xsd
xitemsuptlidel	Item/Supplier translated language Delete Message	XItemSupRef.xsd
xitemimagetlcre	Item/Image translated language Create Message	XItemImageDesc.xsd
xitemimagetlmod	Item/Image translated language Modify Message	XItemImageDesc.xsd
xitemimagetlidel	Item/Image translated language Delete Message	XItemImageRef.xsd
xitemhtscre	Item/HTS create message	XItemDesc.xsd
xitemhtsmod	Item/HTS modify message	XItemDesc.xsd
xitemhtsdel	Item/HTS delete message	XItemRef.xsd
xitemhtsassesscre	Item/HTS assess create message	XItemDesc.xsd
xitemhtsassessmod	Item/HTS assess modify message	XItemDesc.xsd
xitemhtsassessdel	Item/HTS assess delete message	XItemRef.xsd
xitemexpensescre	Item/Expenses create message	XItemDesc.xsd
xitemexpensesmod	Item/Expenses modify message	XItemDesc.xsd
xitemexpensesdel	Item/Expenses delete message	XItemRef.xsd
xitemticketcre	Item/Ticket create message	XItemDesc.xsd
xitemticketmod	Item/Ticket modify message	XItemDesc.xsd
xitemticketdel	Item/Ticket delete message	XItemRef.xsd
xitemseasoncre	Item/Seasons create message	XItemDesc.xsd
xitemseasondel	Item/Seasons delete message	XItemRef.xsd
xitemchgcre	Item up charge create message	XItemDesc.xsd
xitemchgdtlmod	Item up charge detail modify message	XItemDesc.xsd
xitemchgdel	Item up charge delete message	XItemRef.xsd
xitemcttcre	Item Conditional Tariff treatment create message	XItemDesc.xsd
xitemcttdel	Item Conditional Tariff treatment delete message	XItemRef.xsd
xritemscre	Related Items create message	XItemDesc.xsd
xritemsmod	Related Items modify message	XItemDesc.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
xitemsdcl	Related Items delete message	XItemRef.xsd
xitemudafftlcre	Item UDA Free Form Text Translations Create message	XItemDesc.xsd
xitemudafftlmod	Item UDA Free Form Text Translations Update message	XItemDesc.xsd
xitemudafftlcl	Item UDA Free Form Text Translations Delete message	XItemRef.xsd
xitemreqdoccre	Item required documents of an Item create message	XItemDesc.xsd
xitemreqdocdel	Item required documents of an Item delete message	XItemRef.xsd
xritemtlcre	Related Item Translations Create message	XItemDesc.xsd
xritemtlmod	Related Item Translations Update message	XItemDesc.xsd
xritemtlcl	Related Item Translations Delete message	XItemRef.xsd

Item Location Subscription API

This section describes the item location subscription API.

Functional Area

Items

Business Overview

This API subscribes to item location from external systems to create or modify item location combinations in Merchandising. Item/location relationships can be created for an item and a single location or using one of the levels of the organizational hierarchy.

Creating Locations

When a new item location is created, this API will first validate that all required fields are present in the message. Additionally, when creating a new item location at least one detail line must also be included in the message. After that, business level validation on the input information will be performed. The tables below summarize these two types of validation.

Updating Item Location

When updating an item location, this API will first validate that all required fields are present in the message. Additionally, when updating an item location at least one detail line must also be included in the message. After that, business level validation on the input information will be performed. The tables below summarize these two types of validation.

Table 3-64 Item Location Create and Update

Message Element	Required?	Notes
Item	Yes	Must be an existing item in Merchandising.

Table 3-64 (Cont.) Item Location Create and Update

Message Element	Required?	Notes
Hier Level	Yes	Must be a valid organization hierarchy level. Valid values are chain (CH), area (AR), region (RE), district (DI), store (S) or warehouse (W).

Table 3-65 Item Location Detail Create and Update

Message Element	Required?	Notes
Hier Value	Yes	Must be valid ID for a chain, area, region, district, store or warehouse given the Hier Level.
Status	Yes	For new item/location relationships, this must be active (A), inactive (I), discontinued (C). If it is an update to an existing item/location, valid values are active (A), inactive (I), discontinued (C), or delete (D).
Store Ord Mult	Yes	This contains the multiple in which the item needs to be shipped from a warehouse to the location. Valid values are found in ORML.
Source Method	Conditional	This field determines the primary sourcing for this item/location. Valid values are Supplier (S) or Warehouse (W). Must either be supplier site or virtual warehouse. Additionally, valid value is dependent on the Hier Level: For Hier Level of store (S): - Source Method is required to be warehouse (W) for wholesale or franchise stores. For Hier Level of warehouse (W) - Source Method is required to be supplier (S) Source Method must be W if the Source Warehouse is populated.
Source Warehouse	Conditional	This determines which warehouse is the sourcing location for this item/location. This value is required if the sourcing method is Warehouse. Must be a valid virtual warehouse. Additionally, item should already be ranged to the Source Warehouse.
Receive as Type	Conditional	This determines whether the stock on hand for a pack component item or the buyer pack itself will be updated when a buyer pack is received at a warehouse. Valid values are Each (E) or Pack (P). Must be blank if Item is not a pack or Hier Level is not warehouse. Must be eaches (E), when Hier Level is warehouse and Hier Value is an internal finisher. Must be pack (P), when Item is a vendor pack. Must be pack (P), when Item order as type is pack and warehouse break pack indicator is N.
Taxable Indicator	No	This field determines if the item is taxable at the location. Valid values are Yes (Y) or No (N).
UIN Type	No	This contains the type of unique identification number (UIN) used to identify the instances of the item at the location. Valid values are found in code type UINT.

Table 3-65 (Cont.) Item Location Detail Create and Update

Message Element	Required?	Notes
UIN Label	Conditional	This contains the label for the unique identification number (UIN) when displayed. Valid values are found in code type ULBL. UIN Label is required if UIN Type is populated.
Ext UIN Indicator	No	This indicates if unique identification number (UIN) is being generated in the external system. Valid values are Yes (Y) or No (N).
Primary Supplier	No	This contains the primary supplier site for the item/location.
Primary Country	No	This contains the primary country of sourcing for the item/location.
Local Item Description	No	Could be used to have an alternate description for the item at this language, such as in another language.
Ti	No	This determines the number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of cases for a pallet.
Hi	No	This determines the number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of cases for a pallet.
Daily Waste Percent	No	This defines the average percentage lost from inventory on a daily basis due to natural wastage.
Local Short Description	No	Could be used to provide an alternative short description for the item at this location, such as in another language.
UIN Type	No	This contains the type unique identification number (UIN) used to identify the instances of the item at the location. Valid values are found in code type UINT.
UIN Label	No	This contains the label for the unique identification number (UIN) when displayed. Valid values are found in code type ULBL.
Capture Time	No	This determines when the unique identification number (UIN) should be captured for an item during transaction processing. Valid values are found in code type CPTM.
Unit Cost	No	This contains the current unit cost of the item based on the primary supplier/country for the location in local currency.
Cost UOM	No	This is used to allow costs to be managed in a different UOM than the standard UOM.
Purchase Type	No	This defines whether the item is owned, consignment stock, or a concession item at the location. Valid values are: • 0 - Owned • 1 - Consignment • 2 - Concession
Calculation Basis	No	This determines if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are: • C - Cost per Unit • P - Purchase Rate

Table 3-65 (Cont.) Item Location Detail Create and Update

Message Element	Required?	Notes
Purchase Rate	No	This contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item, if the calculation basis for the item/location is Purchase Rate.
Promotable Indicator	No	This determines whether the retailer is allowed to specify if the item is promotable or not. Valid values are Yes (Y) or No (N).
Pricing Control	No	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: • R - Retailer • S - Supplier If no value is provided for consignment/concession purchase types, this will be defaulted from item/supplier/country of sourcing/location if purchase type is Consignment/Concession at item/supplier/country of sourcing/location level. If purchase type is Owned at item/supplier/country of sourcing/location level, then this will be defaulted from item/supplier/country of sourcing if purchase type is Consignment/Concession at item/supplier/country of sourcing level. If purchase type is Owned at item/supplier/country of sourcing level, then this will be defaulted based on Default Pricing Control at supplier site level.
Selling Phase Start Date	No	Used to determine the timing of the generation of markdown recommendation, which will occur x weeks after the start of this phase.
Update Sell Date on Store Order Receipt	No	This indicates whether or not to update the Selling Phase Start Date for the item/location when a purchase order of this item is received at the store.
Update Sell Date on Warehouse to Store Receipt	No	This indicates whether or not to update the Selling Phase Start Date for the item/location when a warehouse to store transfer/allocation of this item is received at the store.
Update Sell Date on Store to Store Receipt	No	This indicates whether or not to update the Selling Phase Start Date for the item/location when a store-to-store transfer of this item is received at the store.
Minimum Units for Sell Date Reset	No	This field contains the minimum shipped quantity of an item on a transfer/allocation that will trigger the resetting of the Selling Phase Start Date. This value is only used if either of the Set Selling Start Date upon Receipt of a Warehouse to Store Transfer/Allocation or Set Selling Start Date upon Receipt of a Store to Store Transfer are set to Yes (Y).
Manufacturer Suggested Retail Tax Inclusive	No	This field holds the tax-inclusive Manufacturer Suggested Retail in local currency and in an item's standard UOM. It is used when default_tax_type is GTS or SVAT.

Table 3-65 (Cont.) Item Location Detail Create and Update

Message Element	Required?	Notes
Manufacturer Suggested Retail Tax Exclusive	No	This field holds the tax-exclusive Manufacturer Suggested Retail in local currency and in an item's standard UOM. It is used when default_tax_type is SALES.

Table 3-66 Item Location Trait Create or Update

Message Element	Required?	Notes
Launch Date	No	This holds the date when the item is initially sold at the location.
Quantity Key Options	No	This field determines whether the quantity key on a POS can be used for this item at the location. Valid values are found in code type RPO.
Manual Price Entry	No	This field determines whether the price for the item/location can be entered manually on POS. Valid values are found in code type RPO.
Deposit Code	No	This determines whether a deposit is associated with this item at the location. Valid values are found in code DEPO.
Food Stamp	No	This determines whether the item is approved for food stamps at the location. Valid values are Yes (Y) or No (N).
WIC	No	This determines whether the item is approved for WIC at the location. Valid values are Yes (Y) or No (N).
Proportional Tare Percent	No	This is the proportion of the total weight of a unit of an item that is the packaging. For example, if the tare item is bulk candy, this is the proportional of the total weight of one piece of candy that is the candy wrapper.
Fixed Tare Value	No	This is the tare of the packaging. For example, if the tare item is bulk candy, this is weight of the bag and twist tie.
Fixed Tare UOM	No	This contains the unit of measure value associated with the tare value.
Reward Eligible	No	This determines whether the item is legally valid for various types of bonus point/award programs at the location. Valid values are Yes (Y) or No (N).
National Brand Competitor Item	No	This contains the nationally branded item to which it will be compared to. Will contain a valid item ID.
Return Policy	No	This determines the return policy for the item at the location. Valid values are found in code type RETP.
Stop Sale	No	This defines if the sale of the item should be stopped immediately at the location (i.e., in case of recall etc.). Valid values are Yes (Y) or No (N).
Electronic Market Clubs	No	This contains the code that represents the marketing clubs to which the item belongs to at the location. Valid values are found in code type MKTC.
Report Code	No	This determines to which reports the location should run. Valid values are found in code type REPC.

Table 3-66 (Cont.) Item Location Trait Create or Update

Message Element	Required?	Notes
Shelf Life on Selection	No	This contains the required shelf life for an item on selection in days.
Shelf Life on Receipt	No	This contains the required shelf life for an item on receipt in days.
Investment Buy Shelf Life	No	Indicates the shelf life in days for this item at this location, which is used by the Investment Buy replenishment processing.
Store Reorderable	No	This determines whether the store may re-order the item. Valid values are Yes (Y) or No (N).
Rack Size	No	This determines the rack size that should be used for the item.
Full Pallet Item	No	This determines whether a store must reorder an item in full pallets only. Valid values are Yes (Y) or No (N).
In Store Market Basket	No	This contains the in-store market basket code for the item/location combination. Valid values are found in code type STMB.
Storage Location	No	This contains the current storage location or bin number for the item at the location.
Alternate Storage Location	No	This contains the preferred alternate storage location or bin number for the item at the location.
Returnable	No	This determines if the item can be returned at this location. Valid values are Y or N.
Refundable	No	This determines if the item is refundable at the location or not. Valid values are Yes (Y) or No (N).
Back Order	No	This determines if the item can be back ordered to the location. Valid values are Yes (Y) or No (N).

Table 3-67 Flex Attributes

Message Element	Required?	Notes
Name	Yes	Hold the attribute name.
Value	No	Holds the value of the attribute for number and character type attributes
Value Date	No	Holds the date for date type attributes.

Flex Attributes

Table 3-68 Flex Attributes

Message Element	Required?	Notes
Name	Yes	Holds the attribute name.
Value	No	Holds the value of the attribute for number and character type attributes
Value Date	No	Holds the date for date type attributes.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S) is returned to the external system, indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
xitemloccre	External item locations create	XItemlocDesc.xsd
xitemlocmod	External item locations modification	XItemlocDesc.xsd

Item Reclassification Subscription API

This section describes the item reclassification subscription API.

Functional Area

Merchandise Hierarchy

Business Overview

Merchandising subscribes to item reclassification messages, which update the department, class, and/or subclass for the item, that are published by an external system. This subscription is necessary in order to keep Merchandising in sync with the external system. This API allows external systems to create and delete item reclassification events within Merchandising.

Only the following item types can be interfaced using this API:

- Transaction level items without a parent
- Parent items, whose child items are the transaction level, such as with a fashion style (parent) and its SKUs (children)
- Complex pack items - but the reclassification cannot include the component items in the pack

The following item types cannot be reclassified:

- Child items with a parent - these are reclassified when the parent is updated
- Reference items - these below transaction level items are automatically reclassified with the transaction level item or its parent, whichever applies
- Simple Packs - these are reclassified when the component item is reclassified

This API allows a reclassification event to be created for a department/class/subclass combination that does not yet exist. This is valid as long as the merchandise hierarchy is scheduled to be created on or prior to the reclassification taking effect.

New/Updated Reclassifications

When a reclassification is created, both a reclassification header and detail are required. For an update, either to update the header or add a detail, all fields in both header and detail nodes are required.

Reclassification Header

Message Element	Required?	Notes
Reclass Number	Always	This is the unique number which identifies the reclassification event. For a detail create message, the reclassification number should already exist in Merchandising.
Reclass Description	Always	This is the description of the reclassification event.
Reclass Date	Always	The date on which the reclassification event is scheduled to take place. The date must be at least a day after the current business day.
To Department	Always	The department to which the item will belong after the reclassification event. The dept/class/subclass combination must already exist or will be created before the reclassification date.
To Class	Always	The class to which the item will belong after the reclassification event. The dept/class/subclass combination must already exist or will be created before the reclassification date.
To Subclass	Always	The subclass to which the item will belong after the reclassification event. The dept/class/subclass combination must already exist or will be created before the reclassification date.
Item Reclassification Detail	Always	Child node

Reclassification Detail

Message Element	Required?	Notes
Item	Always	This is the item that will be reclassified. Following are the validations that will be done on upload: Must be a level 1 item If the item is a pack, it should not be a simple pack. Must not be on any approved order Must not be an orderable buyer pack that can be received as component items. Must not be on an existing reclassification

Deleting Reclassifications

Reclassifications can be deleted by:

- Deleting a single reclassification event
- Deleting specific items on a reclassification event
- Deleting all reclassification events on a particular event date
- Deleting all reclassification events

Reclassification Header

Message Element	Required?	Notes
Reclass Number	Conditional	The reclassification event that will be deleted. If this is populated in a header delete message, then the specific reclassification event will be deleted. This is required in a header delete message if Reclass Date is NULL and the Purge All indicator is N or NULL. If Reclass Number is provided, Reclass Date should be NULL.

Message Element	Required?	Notes
Reclass Date	Conditional	The date of the reclassification events to be deleted. If this is populated in a header delete message, then all reclassification events occurring on the given date will be deleted. This is required in a header delete message if the Reclass Number is NULL and the Purge All indicator is N or NULL. If Reclass Date is provided, Reclass Number should be NULL.
Purge All	Conditional	If this field is Y in a header delete message, then all item reclassification events will be deleted. When the purge indicator is Y, both Reclass Number and Reclass date should be NULL. If the Purge All indicator is NULL or N, only the Reclass Number or the Reclass date should contain a value.
Reclassification Detail	Optional	Child Node

Reclassification Detail

Message Element	Required?	Notes
Item	Conditional	This is the item within a reclassification event that will be deleted. This is the only field required for detail delete messages as an item can only exist in one reclassification event. The entire reclassification event will be deleted when no item detail exists after detail deletion.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
xitemrclscre	External item reclassification create	XItemRclsDesc.xsd
xitemrclsdtlcre	External item reclassification detail create	XItemRclsDesc.xsd
Xitemrclsdel	External item reclassification delete	XitemRclsRef.xsd
Xitemrclsdtldele	External item reclassification detail delete	XItemRclsRef.xsd

Location Trait Subscription API

This section describes the location trait subscription API.

Functional Area

Foundation Data

Business Overview

The Location Trait Subscription API processes incoming data from an external system to create, edit and delete location traits in Merchandising. This data is processed immediately upon message receipt so success or failure can be communicated to the external application.

The table below contains the details of the message as well as the validations.

Table 3-69 Location Traits

Message Element	Required?	Notes
Trait ID	Yes	This field contains the unique id number of the location trait.
Trait Description	Yes	This field contains the description of the location trait.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
xloctrtcre	External Location Trait Create	XLocTrtDesc.xsd
xloctrtdel	External Location Trait Delete	XLocTrtRef.xsd
xloctrmod	External Location Trait Modification	XLocTrtDesc.xsd

Required fields are shown in RIB documentation.

Merchandise Hierarchy Reclassification Subscription API

This section describes the merchandise hierarchy reclassification subscription API.

Functional Area

Merchandise Hierarchy

Business Overview

This API allows Merchandising to subscribe merchandise hierarchy reclassification messages, that are published by an external system. It is intended to be used by retailers who manage their hierarchies in a system outside Merchandising. This API allows for pending merchandise hierarchy reclassification events to be created, modified or deleted. A separate batch process will read the information off the pending merchandise hierarchy table and creates or modifies the merchandise hierarchy information in Merchandising once the effective date arrives.

Creating Merchandise Hierarchy Reclassifications

When a new merchandise hierarchy reclassification is created, the API will first validate that all required fields are present in the message. Certain of the fields are required regardless of hierarchy level, while others are dependent on other hierarchy configurations. After that, business level validation on the input information will be performed. The tables below summarize the validation.

Table 3-70 Header Level Validation

Message Element	Required?	Notes
Merchandise Hierarchy Level	Always	Indicates the level of merchandise hierarchy. Valid values are V (division), G (group), D (department), C (class), and S (subclass).
Merchandise Hierarchy ID	Always	Holds the merchandise hierarchy ID, of the merchandise hierarchy component being created or updated.

Table 3-70 (Cont.) Header Level Validation

Message Element	Required?	Notes
Merchandise Hierarchy parent ID	Always	This field will hold the parent of the hierarchy identified in the Merchandise Hierarchy ID field. This column will only be populated if the Merchandise Hierarchy level is class or subclass.
Merchandise Hierarchy Grandparent ID	Conditional	This field will hold the grandparent ID of the hierarchy identified in the Merchandise Hierarchy ID field. This column will only be populated if the Merchandise Hierarchy Level is subclass.
Merchandise Hierarchy Name	Always	The name of the hierarchy value.
Effective Date	Always	The date the hierarchy change will become effective. The effective date must be greater than or equal to the current date.
Action Type	Conditional	Indicates if this field is an addition (A) or modification (M). It is required on a create message and should not be populated on a modify message.
Buyer	Conditional, Optional	The number of the buyer associated with the entity. This value must be predefined in Merchandising. This field should only hold a value if the hierarchy level indicates division, group, or department.
Purchase Type	Conditional	The code indicates whether items in the department will be created by default as normal merchandise (0), consignment (1), or concession (2). This field is required if the hierarchy level indicates department, otherwise it should be null. Additionally, if the Consignment/Concession system option is set to N, this should always be 0.
Total Market Amount	Optional	This field stores total market amount that is expected for the entity. This field will only be used if the hierarchy value indicates division or department.
Merchandiser	Conditional, Optional	This field indicates the number of the merchandiser associated with the entity. This value must be predefined in Merchandising. This field should hold a value only if the hierarchy level indicates division, group, or department.
Budgeted Markup Percentage	Conditional	This field stores the markup percent of cost. Budgeted Markup Percentage or Budgeted Intake Percentage cannot be both null or both have values. This field is required if the hierarchy level indicates department, otherwise it should be null.
Profit Calculation Type	Conditional	Indicates whether profit will be calculated by direct cost (1) or retail inventory (2). This field is required for a new department, otherwise it should be null. A Department cannot be set up as profit calculation type of Direct Cost and purchase type of Consignment Stock.
Markup Calculation Type	Conditional	Indicates how markup is calculated in the department. Valid values are for a new department, otherwise it should be null.

Table 3-70 (Cont.) Header Level Validation

Message Element	Required?	Notes
OTB Calculation Type	Conditional	Indicates how open to buy is calculated in the department. Valid values are cost (C) and retail (R). This field is required for a new department, otherwise it should be null.
Maximum Average Counter	Conditional	The maximum count of days with acceptable data to include in an average for items with the department. This field is required for a new department, otherwise it should be null. The value cannot be a negative.
Average Tolerance Percentage	Conditional	The tolerance percentage value used in averaging for items within a department. This field will only be used for a new department. The value cannot be a negative.
Budgeted Intake Percentage	Conditional	Indicates the markup percent of retail to use as a default for the department. Budgeted Markup Percentage or Budgeted Intake Percentage cannot be both null or both have values. This field is required for a new department, otherwise it should be null.
Department VAT Inclusive Indicator	Conditional	Indicates the default value for the class VAT indicator. When classes are initially set up, they will inherit this value. This field will only be populated when the hierarchy level indicates department.
Class VAT Indicator	Conditional	Indicates whether retail is displayed and held with or without VAT for items within a class. Valid values are Y (yes) and N (no). This field is required if the hierarchy level indicates department and you are configured for Simple VAT or Global Tax and must be set to Y in those cases. If you are configured for US Sales Tax, then it must be N.

Updating Merchandise Hierarchy Reclassifications

For updating a previously created reclassification, the hierarchy type must be already present in Merchandising. For updates, the validation is similar to that described above for creating a new reclassification. If updating the effective date of a reclassification that has an Add action type, there should not be any child hierarchy with earlier effective date. For example, if you are adding a department, there cannot be a reclassification for adding a class in the department with an earlier effective date.

Deleting Merchandise Hierarchy Reclassifications

To delete a previously created reclassification event, the below validations will be executed to ensure there are no conflicts, along with checks for the existence of child reclassification records, before deleting the record.

Message Element	Required?	Notes
Merchandise Hierarchy Level	Always	Indicates the level of merchandise hierarchy. Valid values are V (division), G (group), D (department), C (class), and S (subclass).

Message Element	Required?	Notes
Merchandise Hierarchy ID	Always	Holds the merchandise hierarchy ID for the selected level.
Merchandise Hierarchy parent ID	Optional	This field will hold the parent of the hierarchy identified in the Merchandise Hierarchy ID field. This column will only be populated if the Merchandise Hierarchy Level is class or subclass.
Merchandise Hierarchy Grandparent ID	Optional	This field will hold the grandparent ID of the hierarchy identified in the Merchandise Hierarchy ID field. This column will only be populated if the Merchandise Hierarchy Level is subclass.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
xmrchhrclscre	Create Merchandise Hierarchy Reclassification	XMrchHrRclsDesc.xsd
xmrchhrclsmod	Modify Merchandise Hierarchy Reclassification	XMrchHrRclsDesc.xsd
xmrchhrclsdel	Delete Merchandise Hierarchy Reclassification	XMrchHrRclsRef.xsd

Merchandise Hierarchy Subscription API

This section describes the merchandise hierarchy subscription API.

Functional Area

Merchandise Hierarchy

Business Overview

The merchandise hierarchy allows the retailer to create the relationships that are necessary to support the product management structure of a company. This hierarchy reflects a classification of merchandise into multi-level descriptive categorizations to facilitate the planning, tracking, reporting, and management of merchandise within the company. If Merchandising is not the system of record for merchandise hierarchy information, then this API

may be used to create, update or delete elements of the merchandise hierarchy, including division, group, department, class, and subclass, based on an external system.

Deleting a division or group will process immediately upon receipt of the message, assuming there are no dependent levels below them. However, departments, classes, and subclasses will not actually be deleted from the system upon receipt of the message. Instead, they will be added to a table, where a background process (Daily Purge of Foundation Data) will occur to ensure the records can be deleted as part of the delete process. For more on this batch process, see the *Retail Merchandising System Operations Guide, Volume 1 - Batch Overviews and Designs*.

If you are implementing Merchandising with Simple VAT as the default tax type, then department-level VAT records can be created and edited within the department message (VAT records are not deleted). VAT creates can be passed in with a department create message, or they can be passed in with their own specific message type. VAT region and VAT codes records must exist prior to creating department VAT records. Also, when passing in a new VAT region to an existing department with attached items, the VAT information will default to all items.

The merchandise hierarchy must be created from the highest level down. Conversely, the hierarchy must be deleted from the lowest level up. Each lower level references a parent level. This means a department is associated with a group; a class is associated with a department; and a subclass is associated with department/class combination because classes are not unique across departments.

Creating a Company

When a new company is created, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed, which checks that the company doesn't already exist. If the validation is met, the company in the message data is created. Only one company can exist in Merchandising.

Updating a Company

When a company is updated, this API will first validate that all required fields are present in the message. Business level validation on the input information will be performed, which verifies if the company ID to be updated already exists. If the company already exists, the company's details in the message data are updated.

The format and validation for creating and updating the company is shown in the table below:

Table 3-71 Company

Message Element	Required?	Notes
Company	Yes	This field contains the unique number which identifies the company for which the system is running.
Company Name	Yes	This field contains the name of the company for which the system is running.
Add_1	Yes	This value contains the first line of the company headquarters address.
Add_2	No	This value contains the second line of the company headquarters address.
Add_3	No	This value contains the third line of the company headquarters address.
City	Yes	This field contains the city of the company headquarters.

Table 3-71 (Cont.) Company

Message Element	Required?	Notes
State	No	This field contains the state code of the company headquarters. This value must be a valid state in Merchandising.
Country Code	Yes	This field contains the country code of the company headquarters. This value must be a valid country in Merchandising.
Postal Code	No	This field contains the postal code of the company headquarters.

Creating Divisions

When a new division is created, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed. The business validation:

- Verifies division is not already present.
- Verifies that, if total market amount is received, then it should be at least 1000.

If all the validations are met, the division in the message data is created.

Updating Divisions

When a division is updated, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed. The business validation:

- Verifies division is present.
- Verifies that, if total market amount is received, then it should be at least 1000.

If all the validations are met, the division's details are updated.

The format and validation for creating and updating divisions is shown in the table below:

Table 3-72 Division

Message Element	Required?	Notes
Division	Yes	This field contains the unique identifier of the division.
Division Name	Yes	This field contains the name of the division.
Merchant	No	This field contains the number of the merchant associated with the division. This value must be a valid merchant in Merchandising.
Buyer	No	This field contains the number of the buyer associated with the division. This value must be a valid buyer in Merchandising.
Total Market Amount	No	This field contains the total market amount that is expected for the division. If this field is not null it must be at least 1000.

Deleting Divisions

When a division is deleted, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed to:

- Verify the division already exists.
- Verify the division is not associated to a diff group.
- Verify the division is not associated to a season id.
- Verify the division is not associated to a ticket type.
- Verify the division is not associated to a UDA.

If the information passes these validations, the division will be added to a purging staging table for processing in the Daily Purge of Foundation Data process.

The format and validation for deleting division is shown in the table below:

Table 3-73 Division

Message Element	Required?	Notes
Division	Yes	This field contains the unique identifier of the division being deleted.

Creating Groups

When a new group is created, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed, which checks whether the group already exists. If it does not exist, the group in the message data is created.

Updating Groups

When a group is updated, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed, which verifies whether the group to be updated already exists. If group already exists, the group details are updated.

The format and validation for creating and updating divisions is shown in the table below:

Table 3-74 Groups

Message Element	Required?	Notes
Group Number	Yes	This field contains the number which uniquely identifies the group.
Group Name	Yes	This field contains the name of the group.
Division	Yes	This field contains the identifier of the division of which the group is a member. This value must be a valid division in Merchandising.
Merchant	No	This field contains the number of the merchant associated with the division. This value must be a valid merchant in Merchandising.

Table 3-74 (Cont.) Groups

Message Element	Required?	Notes
Buyer	No	This field contains the number of the buyer associated with the division. This value must be a valid buyer in Merchandising.

Deleting Groups

When a group is deleted, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed to:

- Verify the group already exists.
- Verify the group is not associated to a diff group.
- Verify the group is not associated to a season id.
- Verify the group is not associated to a ticket type.
- Verify the group is not associated to a UDA.

If the information passes these validations, the group will be added to a purging staging table for processing in the Daily Purge of Foundation Data process.

The format and validation for deleting groups is shown in the table below:

Table 3-75 Groups

Message Element	Required?	Notes
Group Number	Yes	This field contains the unique identifier of the group being deleted.

Creating Departments

When a new department is created, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed. The business validation:

- Verifies the department is not already present.
- Verifies if total market amount is received then it should be at least 1000.
- Verifies the child messages, if included, have their required fields present. The child messages contain the VAT and upcharge details for a department.

If all the validations are met, the department in the message data is created. Custom flex attributes can also be created for the department through this API, if they are active for the department.

Updating Departments

When a department is updated, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed. The business validation:

- Verifies if the department is present.

- Verifies if total market amount is received then it should be at least 1000.
- Verifies the child messages, if included, contain all required fields. The child messages contain VAT and upcharge details for a department.

If all the validations are met, the department in the message data is created. Like with create, custom flex attributes can also be updated for the department, if active.

The format and validation for creating and updating departments is shown in the table below:

Table 3-76 Department

Message Element	Required?	Notes
Department	Yes	This contains the number which uniquely identifies the department.
Department Name	Yes	This contains the name of the department being created or updated.
Buyer	No	This field contains the number of the buyer associated to the department. This value must be a valid buyer in Merchandising.
Purchase Type	Yes	This field contains the code which indicates whether items in this department are normal merchandise (0) consignment stock (1) and concession stock (2).
Total Market Amount	No	This field contains the total market amount that is expected for the department. This value cannot be less than 1000.
Merchandiser	No	This field contains the number of the merchandiser that is associated to the department. This value must be a valid merchant in Merchandising.
Group Number	Yes	This field contains the number of the group to which the department belongs. This value must be a valid group in Merchandising
Budgeted Markup	No	This field contains the budgeted markup percentage, the markup percent of cost. If this value is not populated on the message, it will be calculated to be the inverse of the budgeted intake percentage. This column will hold 70% as 70, not .70. This field must have a value if Budgeted Intake is NULL.
Profit Calc Type	Yes	This field contains the number which indicates whether profit will be calculated by direct cost (1) or retail inventory (2). It determines the accounting method to be used in the stock ledger for this department.
Markup Calc Type	Yes	This field contains the code that indicates how markup is calculated in this department. Valid values are cost (C) and retail (R).
OTB Calc Type	Yes	This field contains the code which indicates how open to buy (OTB) is calculated for this department. Valid values are cost (C) and retail (R).
Maximum Average Counter	Yes	This field contains the maximum count of days with acceptable data to include in an average for items within the department. The value should be greater than zero.
Average Tolerance Percentage	Yes	This field contains the tolerance percentage value used in averaging for items within this value. This column will hold 70% as 70, not .70. The value should be greater than zero.

Table 3-76 (Cont.) Department

Message Element	Required?	Notes
Budgeted Intake	No	This field contains the budgeted intake percentage, which is the percent of the total take that is income. If this field is not populated on the message, it will be calculated as the inverse of the budgeted markup percentage. This column will hold 70% as 70, not .70. This field must have a value if Budgeted Markup is NULL.
Department VAT	No	Child node
Custom Flex Attributes	No	Child node
Department Up-Charges	No	Child node
Inventory Layer Identifier Type	No	This field contains the type of identifier that will be used across systems to manage and identify inventory layers at department level.
Apply Inventory Identifier Type	No	This field will determine if the values at the class level should be overwritten with department-level values.

Deleting Departments

When a department is deleted, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed to:

- Verify the department already exists.
- Verify the department is not associated to a diff group.
- Verify the department is not associated to a season id.
- Verify the department is not associated to a ticket type.
- Verify the department is not associated to a UDA.

If the information passes these validations, the department will be added to a purging staging table for processing in the Daily Purge of Foundation Data process. Flex attributes are deleted when the department is deleted.

The format and validation for deleting departments is shown in the table below:

Table 3-77 Department

Message Element	Required?	Notes
Department	Yes	This contains the number of the department being deleted.
Department Up-Charges	No	Child node

Creating VAT Information for Departments

If you are configured to run Merchandising using Simple VAT (SVAT) for your default tax type, then you can set the VAT rates by region for the department using this API. If included, this API will check for all required fields in the message and create the VAT record for a department. When adding new VAT region to an existing department with attached items, the VAT information will default to all items.

Updating VAT Information for Departments

If you are configured to run Merchandising using Simple VAT (SVAT) for your default tax type, then you can update the VAT rates by region for the department using this API. This API will check for all required fields in the message and updates the VAT information for a department. When updating VAT details for a department with attached items, the VAT information will default to all items.

The format and validation for creating and updating VAT information is shown below:

Table 3-78 Department VAT

Message Element	Required?	Notes
VAT Region	Yes	This contains the number of the VAT region to which this department is associated. This value must be a valid VAT Region in Merchandising.
VAT Code	Yes	This field contains the alphanumeric identifier of the VAT code. This value must be a valid VAT Code in Merchandising.
VAT Type	Yes	This field indicates if the VAT rate is used for purchasing (C), selling (R), or both (B).
Reverse VAT Indicator	No	This field indicates if items in the department are subject to reverse charge VAT at the region. Valid values are Yes (Y) and No (N).

Creating Up-charges for Departments

When a message contains up-charge details, first it validates for the required fields, including "from" locations and "to" locations in the message. If no up-charge record is found, this message creates the up-charge for a department and from/to location combination. As part of the addition, you can indicate in the message if you want to have the up-charges added to existing items or only added for new items. Similarly, there is a flag in the message to indicate whether the new upcharges will be cascaded to transfers and allocations which are unshipped and not in closed or deleted status. The department upcharges will be created as soon as the message is consumed, however the new upcharges will be cascaded to items, transfers, and allocations via batches which runs at the end of every day.

Updating Up-charges for Departments

When a message contains up-charge details, first it validates for the required field in the message. If an up-charge record exists for a department and the from/to location combination in the message, then the up-charge details are updated for the department. As part of the update there is also an option to have the up-charges updated for items in the department, or unshipped transfers and allocations for items in the department. The department upcharges will be updated as soon as the message is consumed, however the updates will be cascaded to items, transfers, and allocations through batches which runs at the end of every day.

The format and validation for creating and updating up-charge information is shown in the table below:

Table 3-79 Department Up-Charge Header

Message Element	Required?	Notes
From Location	Conditional	This field contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is Country (C), Area (A), or Region (R). It will be a store, virtual warehouse, or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either All Stores (AS) or All Warehouses (AW).
To Location	Conditional	This field contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is Country (C), Area (A), or Region (R). It will be a store, virtual warehouse, or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either All Stores (AS) or All Warehouses (AW).
From Location Type	Yes	This field contains the type of source location from which goods will be transferred. Valid values are: • C - Country • A - Area • R - Region • S - store • W - Virtual Warehouse • PW - Physical Warehouse • AS - All Stores • AW - All Warehouses
To Location Type	Yes	This field contains the type of destination location to which goods will be transferred. Valid values are: • C - Country • A - Area • R - Region • S - store • W - Virtual Warehouse • PW - Physical Warehouse • AS - All Stores • AW - All Warehouses
Department Up-Charge Detail	No	Child node

Table 3-80 Department Up-Charge Detail

Message Element	Required?	Notes
Component ID	Yes	This field contains the unique identifier of the Up Charge component. This should be a valid cost component in merchandising.
Component Rate	Yes	This field contains the rate to be charged against the cost of the Item/To Location combinations within the department.

Table 3-80 (Cont.) Department Up-Charge Detail

Message Element	Required?	Notes
Per Count	Yes	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies.
Per Count UOM	Yes	This field contains the unit of measure in which the Per Count is specified. This should be a valid unit of measure in merchandising.
Up-Charge Group	Yes	This field contains the group to which the component ID belongs. Valid values can be found on the codes table.
Component Currency	Yes	This field contains the currency of the Up Charge component. This must be a valid currency in Merchandising.
Effective Date	No	This field contains the date from which the new values are effective in the system.
Item Default Indicator	No	This field indicates if component rate information is updated or not for existing items under the department. Valid values are Yes (Y) and No (No).
Transfer Allocation Default Indicator	No	This field indicates if component rate information is updated or not for existing transfers and allocations under the department. Valid values are Yes (Y) and No (N).
Computation Value Base (CVB)	No	This field contains the Computation Value Basis for the Up Charge component at department level.
Cost Basis	No	This field contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
Include in Total Up Charge Indicator	No	This field contains the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level. Valid values are Y or N.

Deleting Up-charges for Departments

When a message contains an up-charge, first it validates for the required field in the message. If up-charge record exists for a department, this message deletes the upcharge details for a department. Deleting up-charges from a department does not automatically remove them from the items or transfers and allocations for items in the department.

The format and validation for deleting up charges is shown in the table below:

Table 3-81 Department Up-Charge Detail

Message Element	Required?	Notes
Component ID	Yes	This field contains the unique identifier of the Up Charge component being deleted.

Creating Classes

When a new class is created, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed, which checks whether the class already exists. If the class does not exist, the class in the message data is created. Custom flex attributes can also be created for the class through this API, if they are active for the class.

Updating Classes

When a class is updated, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed, which verifies if the class to be updated already exists. If class already exists, the class details are updated. Like with create, custom flex attributes can also be updated for the class, if active.

The format and validation for creating and updating classes is shown in the table below:

Table 3-82 Class

Message Element	Required?	Notes
Class	Yes	This field contains the unique number of the class.
Class Name	Yes	This field contains the name of the class.
Class VAT Indicator	Yes	This field indicates whether retail is displayed and held with or without VAT for items within the class. This field is available when VAT is on in the system and defined at the class level. This field is always defaulted to No (N).
Department	Yes	This field contains the number of the department which contains the class. This value must be a valid department in Merchandising.
Custom Flex Attributes	No	Child node
Inventory Layer Identifier Type	No	This field contains the type of identifier that will be used across systems to manage and identify inventory layers at class level.

Deleting Classes

When a class is deleted, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed to verify:

- The class exists.
- It is not associated to a diff group.
- It is not associated to a season.
- It is not associated to a ticket type.
- It is not associated to a UDA.

If these validations pass, the class will be added to a purging staging table for processing in the Daily Purge of Foundation Data process. Flex attributes are deleted when the class is deleted.

The format and validation for deleting classes is shown in the table below:

Table 3-83 Class

Message Element	Required?	Notes
Class	Yes	This field contains the unique number of the class.
Department	Yes	The number of the department which contains the class. This value must be a valid department in Merchandising.

Creating Subclasses

When a new subclass is created, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed, which checks whether the subclass already exists. If the subclass does not exist, then it is created. Custom flex attributes can also be created for the subclass through this API, if they are active for the subclass. Item Location Selling Phase Attributes like - Selling Phase Start Date, Direct to Store Orders, Warehouse to Store Transfers / Allocations, Store to Store Transfers, Minimum Transfer / Allocation Units for Update can also be created for the subclass.

Updating Subclasses

When a subclass is updated, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed which verifies if the subclass to be updated already exists. If subclass already exists, it is updated. Like with create, custom flex attributes can also be updated for the subclass, if active. Item Location Selling Phase Attributes like - Selling Phase Start Date, Direct to Store Orders, Warehouse to Store Transfers / Allocations, Store to Store Transfers, Minimum Transfer / Allocation Units for Update can also be updated for the subclass.

The format and validation for creating and updating subclasses is shown in the table below:

Table 3-84 Subclass

Message Element	Required?	Notes
Subclass	Yes	This field contains the unique number of the subclass.
Subclass Name	Yes	This field contains the name of the subclass.
Class	Yes	This field contains the number of the class which contains the subclass. This value must be a valid class in Merchandising.
Department	Yes	This field contains the number of the department which contains the class. This value must be a valid department in Merchandising.
Custom Flex Attributes	No	Child node
Default Intercompany Transfer Price	Yes	Must be a valid code from code type ICTP for subclass under a department having cost based accounting. Must be 'R' for subclass under a department having retail based accounting.
Update Sell Date on Store Order Receipt	No	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a purchase order of this item is received at the store.

Table 3-84 (Cont.) Subclass

Message Element	Required?	Notes
Update Sell Date on Warehouse to Store Receipt	No	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a warehouse to store transfer/allocation of this item is received at the store.
Update Sell Date on Store to Store Receipt	No	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a store to store transfer of this item is received at the store.
Minimum Units for Sell Date Reset	No	This field contains the minimum shipped quantity of an item on a transfer/allocation that will trigger the resetting of the Selling Phase Start Date. This value is only used if either of the Set Selling Start Date upon Receipt of a Warehouse to Store Transfer/Allocation or Set Selling Start Date upon Receipt of a Store to Store Transfer are set to Yes (Y).

Deleting Subclasses

When a subclass is deleted, this API will first validate that all required fields are present in the message. Business-level validation on the input information will be performed to:

- Verify the subclass already exists.
- Verify the subclass is not associated to a diff group.
- Verify the subclass is not associated to a season id.
- Verify the subclass is not associated to a ticket type.
- Verify the subclass is not associated to a UDA.

If these validations pass, the subclass will be added to a purging staging table for processing in the Daily Purge of Foundation Data process. Flex attributes are deleted when the subclass is deleted.

The format and validation for deleting classes is shown in the table below:

Table 3-85 Subclass

Message Element	Required?	Notes
Subclass	Yes	This field contains the unique number of the subclass being deleted
Class	Yes	This field contains the unique number of the class.
Department	Yes	The number of the department which contains the class/subclass. This value must be a valid department in Merchandising.

Flex Attributes

If any custom flex attributes (CFAS) for the department, class, or subclass were added or modified, it will trigger an update message. The node of the integration that supports this will contain the name of the attribute as it is defined in the group set level view, the value of the

custom attribute. If it is a date attribute, the date value is in a separate field. Flex attributes can only be added to or updated; they cannot be deleted.

Table 3-86 Flex Attributes

Message Element	Required?	Notes
Name	Yes	Holds the attribute name.
Value	No	Holds the value of the attribute for number and character type attributes
Value Date	No	Holds the date for date type attributes.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
xmrchhrcompcr	External Create Company	XMrchHrCompDesc.xsd
xmrchhrcompmod	External Modify Company	XMrchHrCompDesc.xsd
xmrchhrdivcre	External Create Division	XMrchHrDivDesc.xsd
xmrchhrdivmod	External Modify Division	XMrchHrDivDesc.xsd
xmrchhrdivdel	External Delete Division	XMrchHrDivRef.xsd
xmrchhrgrpcr	External Create Group	XMrchHrGrpDesc.xsd
xmrchhrgrpmod	External Modify Group	XMrchHrGrpDesc.xsd
xmrchhrgrpdel	External Delete Group	XMrchHrGrpRef.xsd
xmrchhrdeptcre	External Create Department	XMrchHrDeptDesc.xsd
xmrchhrdeptmod	External Modify Department	XMrchHrDeptDesc.xsd
xmrchhrdeptdel	External Delete Department	XMrchHrDeptRef.xsd
Xmrchhrvatcre	External Merch Hierarchy VAT create	XMrchHrDeptDesc.xsd
xmrchhrvatmod	External Merch Hierarchy VAT modify	XMrchHrDeptDesc.xsd
xmrchhrdeptchrgcre	External Merch Hier Dept Up-Charge create	XMrchHrDeptDesc.xsd
xmrchhrdeptchrgmod	External Merch Hier Dept Up-Charge modify	XMrchHrDeptDesc.xsd
xmrchhrdeptchrgdel	External Merch Hier Dept Up-Charge delete	XMrchHrDeptRef.xsd
xmrchhrclscre	External Create Class	XMrchHrClsDesc.xsd
xmrchhrclsmod	External Modify Class	XMrchHrClsDesc.xsd
xmrchhrclsdel	External Delete Class	XMrchHrClsRef.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
xmrchhrsclsdel	External Create Subclass	XMrchHrScIsDesc.xsd
xmrchhrsclsmod	External Modify Subclass	XMrchHrScIsDesc.xsd
xmrchhrsclsdel	External Delete Subclass	XMrchHrScIsRef.xsd

Order Subscription API

This section describes the PO subscription API.

Functional Area

Purchase Orders

Business Overview

This subscription API is used to keep Merchandising in sync with an external system that is responsible for maintaining purchase orders. It is assumed that the source of orders sent in this API is not the supplier, as vendor managed inventory (VMI) POs can be sent using the Upload Purchase Order and Purchase Order Change Acknowledgements from Suppliers to Merchandising (ediupack) batch upload. It also does not support creating customer order POs or contract POs. Customer order POs are assumed to be sent using the Customer Order Fulfillment Subscription API and contract orders are created using replenishment processes in Merchandising or manually using the UI.

POs can be created, modified or deleted at the header or the detail level. This API also creates, edits, and deletes other data associated with a purchase order, including letter of credit, expenses, harmonized tariff schedules (HTS) and assessments, and custom flex attributes (CFAS). It also will apply rounding rules, default inventory management parameters, apply bracket costs, update open to buy buckets, and insert a record into the deals queue for deals to be applied to the order, if applicable. These transactions are performed immediately upon receipt of the message so that success or failure can be sent back to the calling application.

If the location on a purchase order is a franchise store, a corresponding franchise order is also created along with the PO.

Creating Purchase Orders

New purchase order messages pass through a series of validations such as required field and valid value validations for each field as well as business validations. The tables below summarize these validations.

Table 3-87 Header Level Validation

Message Element	Required?	Notes
Order Number	Always	Must be a unique order number not used by any existing purchase orders in Merchandising.

Table 3-87 (Cont.) Header Level Validation

Message Element	Required?	Notes
Supplier	Always	Must be a valid, active supplier in Merchandising. The supplier and locations on the order must belong to the same org unit. If the EDI PO indicator is set to Y, the supplier of the order must also be an EDI supplier.
Currency Code	Optional	Must be a valid currency code. If not provided, this defaults to the currency code used by the supplier.
Terms	Optional	Must be a valid payment terms in Merchandising. If terms is not provided in the message, the API will default this to supplier terms.
Not Before Date	Optional	Must be equal to or after the current date and before the Not After Date, if provided. If the date is not provided, the API will default the value to the current date + default supplier lead time if there are not items attached to the order. If items are already added to the order, it will be defaulted to current date + minimum lead time + minimum pickup lead time among all items in the order.
Not After Date	Optional	Must be equal to or after the current date and after the Not Before Date, if provided. If the date is not provided, the API will default the value to the current date + default supplier lead time if there are not items attached to the order. If items are already added to the order, it will be defaulted to current date + maximum lead time + maximum pickup lead time among all items in the order.
OTB End of Week Date	Optional	Must be a valid end of week date and equal to or after the current date. If the date is not provided, the API will default the value to the last day of the week that the not after date value falls in.
Department	Optional	Must be a valid department in Merchandising. The department field should not be populated if items belonging to different departments are present in the order. Department is required if the Department Level PO system option is Y.
Status	Optional	Valid statuses are Worksheet (W), Submitted (S), or Approved. If not provided, the status will be defaulted to W. A status of closed (C) is also allowed, but only for order updates.
Exchange Rate	Optional	The exchange rate should be greater than zero. If not provided, and the currency code is provided, the exchange rate will be based on the given currency code and primary currency. If the currency code is not provided, then the exchange rate will default based on the supplier's currency code and primary currency.
Include On Order Indicator	Optional	Valid values are Y and N. If not provided in the message, it will be defaulted to Y.
Written Date	Optional	If not provided in the message, it will be defaulted to the current date.

Table 3-87 (Cont.) Header Level Validation

Message Element	Required?	Notes
Origin Indicator	Optional	Valid values for origin indicator are: 0 – Merchandising generated PO (from replenishment) 1 – Other system generated PO 2 – Manual purchase order 3 – Buyer worksheet PO 4 – Consignment sales generated PO 5 – Vendor generated PO 6 – AIP generated PO 7 – SIM generated PO 8 – Allocation generated PO 9 – Consignment transfer generated PO 10 – Consignment ownership change generated PO 11 - Consignment Inventory Adjustment 12 - Consignment Vendor Order The expected values for this field in purchase order subscription are 1, 2, 6, 7, 8, and 12. If it is not provided in the message, it will be defaulted to 2.
EDI PO Indicator	Optional	Valid values are Y and N. If not provided in the message, it will be defaulted to the supplier's EDI PO indicator.
Pre Mark Indicator	Optional	Valid values are Y and N. If not provided in the message, it will be defaulted to N. If Y, then the order must be pre-allocated before it is approved.
User ID	Optional	If not passed into the message, then a value will be defaulted for auditing purposes.
Comment	Optional	
Attempt RMS Load	Optional	If not passed into the message, then it will be defaulted to RMS, which means that the information will persist to the Merchandising tables as opposed to the staging tables. Valid values are RMS and STG.
Master PO number	Optional	This is can be used for linking multiple orders together for multiple delivery date orders.
Lading Port	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid lading port in Merchandising.
Discharge Port	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid discharge port in Merchandising.
Factory	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid, active factory in Merchandising.
Agent	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid, active agent in Merchandising.

Table 3-87 (Cont.) Header Level Validation

Message Element	Required?	Notes
Ship Method	Optional	If not provided, this will be defaulted based on supplier attributes. This must be a valid ship method in Merchandising, which are stored in the codes table under the code type SHPM.
Partner Type 1	Optional	This should be provided if partner 1 is given. Valid values are S1, S2 and S3. These are stored under the codes table under the code type SUHL.
Partner 1	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid, active partner in Merchandising for the given partner type. Partner and partner type should be provided, or both should be null.
Partner Type 2	Optional	This should be provided if partner 2 is given. Valid values are S1, S2 and S3. These are stored under the codes table under the code type SUHL.
Partner 2	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid, active partner in Merchandising for the given partner type. Partner and partner type should be provided, or both should be null.
Partner Type 3	Optional	This should be provided if partner 3 is given. Valid values are S1, S2 and S3. These are stored under the codes table under the code type SUHL.
Partner 3	Optional	If not provided, this will be defaulted based on supplier import attributes. If provided, it should be a valid, active partner in Merchandising for the given partner type. Partner and partner type should be provided, or both should be null.
Payment Method	Optional	Valid values are stored under the code type PYMT in the codes table. If not provided, this will default to the supplier payment method.
Purchase Type	Optional	Valid values are stored under the code type PURT in the codes table. If not provided, this will default to the purchase type defined at the supplier inventory management level.
FOB Title Pass	Optional	This is required for import orders with payment method of Letter of Credit. Valid values are stored under the code type FGBT in the codes table.
FOB Title Pass Desc	Optional	This is required for import orders with payment method of Letter of Credit.
PO Type	Optional	This should be a valid PO Type in Merchandising. Valid PO Types are found in the PO_TYPE table.
Import Country ID	Optional	This is required for import orders. The value should be a valid country in Merchandising. If the value is not provided in the message, it will default to the country ID of the primary address of the location in the order.
Order Type	Optional	Indicates the type of order and which Open To Buy bucket will be updated. Valid values are found against code type ORDO

Table 3-87 (Cont.) Header Level Validation

Message Element	Required?	Notes
Buyer	Optional	This field contains the ID of the buyer associated with the PO. Validation is performed against the Buyer table.
Location Type	Optional	This field contains the type of location in the location field. Valid values are S (Store) or W (Warehouse).
Location	Optional	This field contains the location where all items on the order will be delivered to. If populated, it will mean a single location order.
Promotion	Optional	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
QC Indicator	Optional	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.
Freight Terms	Optional	This refers to the freight terms related to the PO.
Backhaul Type	Optional	This field contains the type of backhaul allowance that will be applied to the order.
Backhaul Allowance	Optional	This field will contain the backhaul allowance value.
Ship Pay Method	Optional	This indicates the payment terms for freight charges associated with the order. The code type SHMT holds the valid values.
FOB Trans Responsibility	Optional	Contains the code indicating the type of the location that is responsible for the transportation of the order.
FOB Trans Responsibility Description	Optional	This field describes the code for the location responsible for the transportation of the order.
Vendor Order No	Optional	This field contains the vendor's unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to an Oracle Retail order when the order is created on-line.
Freight Contract No	Optional	This field contains the number of the contract with a shipper that will give specific freight rates.
Pickup Location	Optional	This field contains the location at which the order will be picked up, if the order is a Pickup order.
Pickup No	Optional	This field contains the reference number of the Pickup order.
Pickup Date	Optional	This field contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.
Appointment Date/Time	Optional	This column will hold the date and time of the receiving appointment at the warehouse.
Import Type	Optional	This is the default importer/exporter assigned to the supplier of the Purchase order. Valid values are stored against code type BTLT.
Invoice Location	Optional	This identifies the importer/exporter assigned to the supplier. It should reference the WH.WH column.
Clearing Zone	Optional	This column will hold the clearing zone ID.

Table 3-87 (Cont.) Header Level Validation

Message Element	Required?	Notes
Routing Location	Optional	This is the default routing location for the import order. It refers the OUTLOC.OUTLOC_ID column.
Master PO No	Optional	This field holds the Master Order Number.
Re-Approve	Optional	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again.
Earliest Ship Date	Optional	The date before which the items on the purchase order cannot be shipped by the supplier. Represents the earliest earliest ship date of all the items on the order.
Latest Ship Date	Optional	The date after which the items on the purchase order cannot be shipped by the supplier. Represents the greatest latest ship date of all the items on the order.
File ID	Optional	In order to support supplier pooling via the buyer worksheet, the recommended order quantities for the individual suppliers are linked together for logistical purposes while generating orders from the worksheet based on this identifier.

Table 3-88 Detail Level Validation

Message Element	Required?	Notes
Item Number	Required	Must be an approved, orderable transaction level item in Merchandising supplied by the supplier indicated in the message. The Item Number should not be in the process of being deleted and must be active at the location specified in the message.
Location	Required	Must be a valid stockholding store or virtual warehouse in Merchandising.
Location Type	Required	Valid values are store (S) and warehouse (W).
Unit Cost	Optional	Must be greater than or equal to zero. If more than one virtual warehouse for the same physical warehouse are included in the details for the order, then the unit cost must be the same for those item/warehouses.
Reference Item	Optional	Must be a valid, approved reference item of the item being ordered and should not be in the process of being deleted. The Reference Item should be supplied by the supplier in the message.
Origin Country ID	Optional	Must exist as a valid country for the supplier/item provided in the message. If the country is not provided in the message, the value is defaulted to the item's primary country of sourcing.
Supplier Pack Size	Optional	Must be greater than zero. If there are several order lines with the same item in the message, the supplier pack size and origin country of these records should all be the same. If not provided in the message, the API will default the value based on the supplier/country for the item.
Quantity Ordered	Required	Must be greater than zero.

Table 3-88 (Cont.) Detail Level Validation

Message Element	Required?	Notes
Cancel Indicator	Optional	This is used for purchase order detail modification only.
Reinstate Indicator	Optional	This is used for purchase order detail modification only.
Delivery Date	Optional	The date by which goods are to be delivered.
Qty Cancelled	Optional	This field contains the cancelled quantity for the item in the order.
Cancel Code	Optional	This field contains the reason that the line item was cancelled. This field is required if a line item is cancelled.
Estimated In Stock Date	Optional	Date that the item on the PO is expected to be available to ship from the PO location to another location.
Earliest Ship Date	Optional	The date before which the item cannot be shipped by the supplier. It will be validated to be a valid date equal to or greater than the business date
Latest Ship Date	Optional	The date after which the item cannot be shipped by the supplier. It will be validated to be a valid date equal to or greater than the business date.

If the above validation passes, then the purchase order and details will be created with the status set in the message. If the status in the message is approved, then the order will also be subjected to a series of approval checks. If the order cannot be approved, it will not be created.

You can also include information on the order - letter of credit (if the payment method is letter of credit), landed cost expenses, and for import orders you can include HTS and assessments. If included, these will be created simultaneously with the creation of the order.

Updating Purchase Orders

Updates can be made either at header level or at detail level for orders that are in Worksheet, Approved or Closed status. For both kinds of update messages, the API will validate that the order number included in the message already exists in Merchandising while item number and location will be also validated for existence in detail level updates.

Header Level Updates

Only header level fields need to be provided for header-level updates. Any order details included in the message will be ignored for a header-level update message. There are certain fields that are not allowed to be updated at header level depending on the status, and if these are still provided in the message, no error message will be returned. The values will simply be ignored. However, modifying the following header level fields is allowed made while the order is submitted or approved, without having to set the order in worksheet status:

- status
- not before date
- not after date
- terms
- include on-order indicator

- comments

Detail Level Updates

Order details can be updated for orders in Approved, Worksheet, Submitted or Closed status. The only information needed at the header level is the order number, which if not provided, will cause the message to be rejected. All other details provided at the header level will be ignored. Modifying order quantity, as well as supplier pack size or unit cost on an approved or submitted order will in effect set the order status to worksheet and subject it for automatic re-approval. When modifying order quantities, the full amount should be provided, not just the difference in the old and new values. Validations are also done on quantity changes, such as the ordered quantity should not go below the allocated quantity or replenishment quantity, quantity ordered cannot not be less than quantity received.

Fields that can be modified in worksheet, submitted and approved status:

- Supplier Pack Size
- Unit Cost - for items with no received quantities
- Quantity Ordered

Fields that cannot be modified in statuses other than worksheet:

- Origin Country ID
- Location

Fields that can be modified only in approved status:

- Quantity Cancelled
- Cancel Code

Reinstating Order Lines

To reinstate orders, the reinstate indicator should be set to Y. In effect, this will set the cancelled quantities of the line items to 0 and reinstate the ordered quantities. This will set the status of the reinstated order to Worksheet.

Cancelling a Line Item in an Approved Order

In order to cancel a line item on the order, you can set the cancel indicator at the detail level to Y and at the same time, the quantity ordered for that line item must be set to 0. For partial cancellations, either reduce the quantity of an approved order or populate the quantity cancelled field making sure the cancel indicator is blank or set to N. This will allow for the automatic re-approval of the entire order, if there are line items still on the order once processed by the API. The cancel indicator and reinstate indicator cannot be set to Y at the same time.

Deleting Purchase Orders

If you are deleting a line item on the purchase order or deleting the whole purchase order, the API will first validate that the order number is valid. The order number is the only required field for a header delete message. All other fields will be ignored. For detail delete messages, you must provide the item as well and optionally, the location. These should exist in Merchandising, or else a reject message will be returned.

Deleting the Entire Order

In order to delete an entire order, you must send a header delete message. This will in effect set the status of the order at the header level to D. Only worksheet orders can be deleted. Deleting the purchase order cannot be done if the order is submitted, approved or has been approved, or if allocations exist for the order. Delete messages will still be processed, however it will be treated as an update of cancelled quantity and the quantity ordered will be reduced to the quantity available to be cancelled. If this results in all line items being cancelled or if the delete is made at header level, the status of the order will become Closed.

If an order is still in worksheet status, the entire order will be deleted. If the order involves any franchise stores, then any franchise order or return created with the order will also be cancelled or deleted.

Deleting a Line Item

If an order is still in worksheet status, line items will be deleted from the order. If all line items are deleted, the order header will also be deleted. For orders that are not in worksheet status, when a detail delete is requested, it will update the quantities to cancelled quantities and will be subject for re-approval.

Table 3-89 Header Level Validation

Message Element	Required?	Notes
Order Number	Always	Must be an existing order number in Merchandising.
Order Detail	Optional	Child node
Attempt RMS Load	Optional	This field indicates if the message will persist in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
Location Expense	Optional	Child node
Item HTS	Optional	Child node

Table 3-90 Detail Validation

Message Element	Required?	Notes
Item	Optional	This field contains the item number that will be deleted in the order.
Location	Always	This field contains the location the item is ordered to.
Reference Item	Optional	The ID of a reference item which can be used instead of using the item field. If the item field is not populated this field is required.

Creating a Purchase Order Letter of Credit

A letter of credit may be created together with the creation of a new order or added to an existing order with a payment method of Letter of Credit. In order to create/edit/delete a letter of credit, the order should be in worksheet status. Below are the validations:

Message Element	Required?	Notes
Letter of Credit Reference ID	Optional	The reference ID must exist in Merchandising for the given beneficiary and applicant. The Free on Board title pass description and purchase type in the letter of credit table must match that of the values in the order.
Letter of Credit Group ID	Optional	If included, must be a valid value in Merchandising.
Applicant	Always	The applicant must be an active partner in Merchandising.
Beneficiary	Always	Must be a valid active supplier who can be a beneficiary.
Merchandise description	Always	A description of the merchandise being imported for customs purposes.
Transshipment Indicator	Always	Valid values are Y or N.
Letter of credit indicator	Always	Valid values are Y or N.

Updating a Purchase Order Letter of Credit

In order to update an order letter of credit, the letter of credit must exist for the order in Merchandising, otherwise, an error will be returned. All fields identified in the create section above are updateable and will go through the same validation as in the creation of a letter of credit.

Deleting a Purchase Order Letter of Credit

In order to delete an order letter of credit, the letter of credit must exist for the order in Merchandising, otherwise, an error will be returned.

Creating Expenses

Expenses may be created together with the creation of a new order or added to an existing order that has location records defined. In order to create/edit/delete expenses, the order should be in worksheet status. Below are the validations:

Message Element	Required?	Notes
Item	Always	The item/location/location type combination must be present in the order. For buyer packs with an order as type of Pack, this should be a component item in the pack and the item on the order should be present in the pack item field.
Pack item	Optional	If provided, the item/pack item/location/location type combination must be present in the order. This is required if the item on the order is a buyer pack with an order as type of Pack.
Location	Always	The item/location/location type combination must be present in the order.
Location type	Always	The item/location/location type combination must be present in the order. Valid values are S or W.
Component ID	Always	Must be a valid expense component in Merchandising. This should be present in ELC_COMP.

Message Element	Required?	Notes
CVB code	Conditional	Required if the component rate calculation basis is value (V), otherwise this will be defaulted to NULL. Must be a valid CVB code in Merchandising.
Cost basis	Optional	Valid values are supplier (S) or order (O). If CVB code is provided, then this should be null.
Component rate	Optional	This will be defaulted based on the component if not provided.
Component currency	Optional	This will be defaulted based on the component if not provided. If it is present in the message, this must be a valid currency code.
Exchange rate	Optional	This should be the exchange rate used in relation to the location on the order. If this is not provided in the message, the API defaults it, depending on the order exchange indicator set at system level. If the indicator is Y, it defaults based on the component currency. If the component currency of the component is the same as that of the location, then exchange rate should be 1. If the system level indicator is set to N, the exchange rate will be based on the location currency .
Per count	Optional	If the component rate calculation basis is specific (S), it is defaulted based on the expensed component if not provided in the message.
Per count UOM	Optional	If the component rate calculation basis is specific (S), it is defaulted based on the expense component if not provided in the message. This must be a valid unit of measure.
Nominal flag 1	Optional	This will be defaulted based on the expense component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 2	Optional	This will be defaulted based on the expense component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 3	Optional	This will be defaulted based on the expense component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 4	Optional	This will be defaulted based on the expense component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 5	Optional	This will be defaulted based on the expense component if not provided. If it is present in the message, this must be N, +, or -.

Updating Expenses

In order to update expenses, the order/item/location/component ID must exist for the order in Merchandising, otherwise, an error will be returned. All fields identified in the create section above except for order/item/pack item/location/component id are updateable and will go through the same validation as in the creation of expenses.

Deleting Expenses

In order to delete expenses, the order/item/location/component ID must exist for the order in Merchandising, otherwise, an error will be returned.

Table 3-91 Order Expense Deletion

Message Element	Required?	Notes
Item	Always	This field contains the item number that will be deleted in the order.
Pack Item	Optional	If provided, the item/pack item/location combination must be present in the order.
Location	Always	This field contains the location that the item was ordered to. The item/location combination must be present in the order.
Component ID	Always	Must be a valid expense component in Merchandising. This should be present in ELC_COMP.

Creating HTS and Assessments

HTS and Assessment may be created together with the creation of a new import order or added to an existing order that has location records defined. In order to create/edit/delete HTS and assessments, the order should be in worksheet status. Below are the validations:

Table 3-92 HTS Validation Message Elements

Message Element	Required?	Notes
Item	Always	The item must exist on the order. For buyer packs with an order as type of Pack, this should be a component item in the pack and the item on the order should be present in the pack item field.
Pack item	Conditional	The item must exist on the order. This is required if the item on the order is a buyer pack with an order as type of Pack.
HTS	Always	Must be a valid HTS code for the supplier's import country in Merchandising
Status	Always	Valid values are Worksheet (W) or Approved (A).
Origin Country ID	Optional	Must be a valid country in Merchandising. If the HTS tracking level based on country of manufacture, this should be a valid country of manufacture for the item/supplier. If it is not provided, it will default to the item's primary manufacturing country. If the HTS tracking level is country of sourcing, it will default to the primary sourcing country for the item/supplier.
Import Country ID	Optional	Must be a valid country in Merchandising. If not provided, this will default to the import country ID at the order level.

Table 3-93 HTS Assessment Validation

Message Element	Required?	Notes
Component ID	Always	Must be a valid assessment component ID in Merchandising.
Component rate	Optional	This will be defaulted based on the component if not provided.
Per count	Optional	This is defaulted to the component when the calculation basis is specific (S), otherwise this will be defaulted to NULL.
Per count UOM	Optional	This is defaulted to the component when the calculation basis is specific (S), otherwise this will be defaulted to NULL.
CVB code	Optional	This is defaulted to the component when the calculation basis is value (V), otherwise this will be defaulted to NULL.
Nominal flag 1	Optional	This will be defaulted based on the assessment component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 2	Optional	This will be defaulted based on the assessment component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 3	Optional	This will be defaulted based on the assessment component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 4	Optional	This will be defaulted based on the assessment component if not provided. If it is present in the message, this must be N, +, or -.
Nominal flag 5	Optional	This will be defaulted based on the assessment component if not provided. If it is present in the message, this must be N, +, or -.

Updating HTS and Assessments

In order to update HTS and assessments, the record to be updated must exist in Merchandising, otherwise, an error will be returned. Status and origin country ID can be updated at the HTS level. For assessments, all fields identified in the create section above except for component ID are updateable and will go through the same validation as in the creation of assessments.

Deleting HTS and Assessments

In order to delete HTS and assessments, expenses, the record to be deleted must exist in Merchandising, otherwise, an error will be returned.

Table 3-94 Order HTS

Message Element	Required?	Notes
Item	Always	This field contains the item number that will be deleted in the order.

Table 3-94 (Cont.) Order HTS

Message Element	Required?	Notes
Pack Item	Optional	If provided, the item/pack item combination must be present in the order.
HTS	Always	This field contains the unique identifier for the Harmonized Tariff Schedule code. The item/pack item/HTS must be present in the order.
HTS Assessments	Optional	Child node

Table 3-95 Order HTS Assessments

Message Element	Required?	Notes
Component ID	Always	This field contains the assessment component ID. The item/pack item/HTS/component ID must be present in the order.

Publishing Updates

Purchase orders will be published back to the RIB if approved or previously approved, such that system responsible for managing the purchase orders are notified.

Flex Attributes

If custom flex attributes (CFAS) have been defined for purchase orders, or at the order/item or order/item/location level, then they can be integrated as part of this API. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute. Flex attributes can only be added to or updated on a purchase order at header and detail levels but cannot be deleted.

Table 3-96 Flex Attributes

Message Element	Required?	Notes
Name	Yes	Holds the attribute name.
Value	No	Holds the value of the attribute for number and character type attributes
Value Date	No	Holds the date for date type attributes.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
XorderCre	Order Create Message	XOrderDesc.xsd
XorderCre (CustFlexAttriVo)	Order Flex Attribute Create Message for the Order Header	XOrderDesc.xsd
XorderDtlCre	Order Detail Create Message	XOrderDesc.xsd
XorderDtlCre (CustFlexAttriVo)	Order Detail Flex Attribute Create message	XOrderDesc.xsd
XorderLCCre	Order LC Create Message	XOrderDesc.xsd
XorderLocExpCre	Order Location Expense Create Message	XOrderDesc.xsd
XorderSkuHtsCre	Order SKU HTS Create Message	XOrderDesc.xsd
XorderSkuHtsAssessCre	Order SKU HTS Assess Create Message	XOrderDesc.xsd
XorderMod	Order Modify Message	XOrderDesc.xsd
XorderMod (CustFlexAttriVo)	Order Flex Attribute Modify Message for the order header	XOrderDesc.xsd
XorderDtlMod	Order Detail Modify Message	XOrderDesc.xsd
XorderDtlMod (CustFlexAttriVo)	Order Detail Flex Attribute Modify Message	XOrderDesc.xsd
XorderLCMod	Order LC Modify Message	XOrderDesc.xsd
XorderLocExpMod	Order Location Expense Modify Message	XOrderDesc.xsd
XorderSkuHtsMod	Order SKU HTS Modify Message	XOrderDesc.xsd
XorderSkuHtsAssessMod	Order SKU HTS Assess Modify Message	XOrderDesc.xsd
XorderDel	Order Delete Message	XOrderRef.xsd
XorderDtlDel	Order Detail Delete Message	XOrderRef.xsd
XorderLCDel	Order LC Delete Message	XOrderRef.xsd
XorderLocExpDel	Order Location Expense Delete Message	XOrderRef.xsd
XorderSkuHtsDel	Order SKU HTS Delete Message	XOrderRef.xsd
XorderSkuHtsAssessDel	Order SKU HTS Assess Delete Message	XOrderRef.xsd

Organizational Hierarchy Subscription API

This section describes the organizational hierarchy subscription API.

Functional Area

Organizational Hierarchy

Business Overview

If Merchandising is not the system of record for organizational hierarchy information for an implementation, then this API may be used to create, update or delete elements of the hierarchy based on an external system. The organization hierarchy subscription also assigns existing location traits to, or deletes them from the stores in the given organization hierarchy level.

The following organizational hierarchy elements can be created, modified, or deleted using this API: chain, area, region, or district. The organizational hierarchy must be created from the highest level down. Conversely, the hierarchy must be deleted from the lowest level up.

The following validation is applicable for the data sent in this API:

Message Element	Required?	Notes
Hierarchy Value	Always	Indicates the ID of the hierarchy component being added, modified, or deleted. It is validated based on the hierarchy level attribute included.
Hierarchy Description	Conditional	Indicates the description for the hierarchy component. This is required for create only. For modifications, it is optional. For delete, it is ignored.
Hierarchy Level	Always	Indicates the level of the hierarchy being created, updated, or deleted. Valid values are CH (chain), AR (area), RE (region) and DI (district).
Parent	Conditional	Identifies the parent level of the hierarchy for this component. For example, if the hierarchy level is district, this would be the ID of the region for the district.
Manager Name	Optional	Name of the manager for this hierarchy component.
Currency Code	Optional	The code that identifies the currency under which the hierarchy component operations.

Other Notes:

- Location trait records must exist prior to attaching them to any hierarchy.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Type	Message Type Description	XML Schema Definition (XSD)
XOrgHrCre	External Create Organizational Hierarchy	XOrgHrDesc.xsd
XOrgHrLocTrtCre	External Create Location Trait	XOrgHrDesc.xsd
XOrgHrDel	External Delete Organizational Hierarchy	XOrgHrRef.xsd
XOrgHrLocTrtDel	External Delete Location Trait	XOrgHrRef.xsd
XOrgHrMod	External Modify Organizational Hierarchy	XOrgHrDesc.xsd

Payment Terms Subscription API

This section describes the payment terms subscription API.

Functional Area

Payment Terms

Business Overview

Payment terms are supplier-related financial arrangement information that can be subscribed to by Merchandising from a financial system. Payment terms are the terms established for paying a supplier (for example, 2.5% for 30 days) for purchase orders. After confirming the validity of the records enclosed within the message, Merchandising updates its tables with the information.

Creating Payment Terms

When a new payment term is subscribed to by Merchandising, it will first validate that all required fields are present in the message. Payment terms details should also be present when creating a new payment term, and when creating and updating a new payment term detail. After that, business level validation on the input information will be performed. The tables below summarize these two types of validations.

Table 3-97 Header Level Validation

Message Elements	Required?	Notes
terms	Always	This represents the unique ID to track this payment term in Merchandising.
terms code	Always	This is value is intended to hold the code in the financial system.
terms desc	Always	Description of the supplier terms
rank	Always	Unique number to rate invoice payment terms against purchase order terms

Table 3-98 Message Elements

Message Elements	Required?	Notes
due max amount	Always	This is the maximum amount due by a certain date
percent	Always	Percentage discount if payment is made within the time frame
terms sequence	Optional	The order in which to apply the discount percent
due days	Conditional	This is the number of days until payment is due. The following due days data combinations are valid: 1. Due days, due day of month and due months forward should all be provided 2. Due days, due day of month and due months forward are all NULL 3. Due days should be provided and both due day of month and due months forward are provided.
due day of month	Conditional	Day of month used to calculate due date of invoice payment line. The following due days data combinations are valid: 1. Due days, due day of month and due months forward should all be provided 2. Due days, due day of month and due months forward are all NULL 3. Due days should be provided and both due day of month and due months forward are provided.
due months forward	Conditional	Number of months ahead used to calculate due date of invoice payment line. The following due days data combinations are valid: 1. Due days, due day of month and due months forward should all be provided 2. Due days, due day of month and due months forward are all NULL 3. Due days should be provided and both due day of month and due months forward are provided.
discount days	Conditional	This is the number of days in which payment must be made in order to receive the discount. The following discount days data combinations are valid: 1. Discount days, discount day of month and discount months forward should all be provided 2. Discount days, discount day of month and discount months forward are all NULL 3. Discount days should be provided and both discount day of month and discount months forward are provided. 4. Discount days is NULL and both discount day of month and discount months forward are provided.

Table 3-98 (Cont.) Message Elements

Message Elements	Required?	Notes
discount day of month	Conditional	Day of month used to calculate discount date for invoice payment line The following discount days data combinations are valid: 1. Discount days, discount day of month and discount months forward should all be provided 2. Discount days, discount day of month and discount months forward are all NULL 3. Discount days should be provided and both discount day of month and discount months forward are provided. 4. Discount days is NULL and both discount day of month and discount months forward are provided.
discount months forward	Conditional	Number of months ahead to calculate discount date for invoice payment line. The following discount days data combinations are valid: 1. Discount days, discount day of month and discount months forward should all be provided 2. Discount days, discount day of month and discount months forward are all NULL 3. Discount days should be provided and both discount day of month and discount months forward are provided. 4. Discount days is NULL and both discount day of month and discount months forward are provided.
fixed date	Optional	This is the fixed due date
enabled flag	Always	This flag should be 'Y' if the start active date is less than or equal to the current date and the end date is greater than or equal to the current date. Otherwise, the enabled flag should be 'N'.
start active date	Optional	Start active date must be less than end active date.
end active date	Optional	End date should be greater than start active date.
cutoff day	Optional	This is the last day before payment is scheduled

Payment term details sent via the detail message can also be added when a payment term already exists in Merchandising. If the terms detail being added already exists, an error is raised.

After the message passes all validations, the payment terms are inserted into the Merchandising tables.

Updating Payment Terms

Payment terms can be updated at header or detail level. When updating at the header level, the payment term details for the term being updated should already exist in Merchandising. When updating payment term details, the term header and detail must already exist.

After the message passes all validations, the payment terms in Merchandising are updated. Rank, terms, code, and terms description are the values that can be updated at header level. At the detail level, due days, due max amount, due months forward, discount days, percent, discount day of month, discount months forward, fixed date, enabled flag, start active date, end active date and cutoff day may be updated.

Error Handling

If an error occurs in this procedure, a call will be placed to a function to build a complete error message. This message together with a status of E is returned to the external system. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
PayTermCre	Payment Terms Create Message	PayTermDesc.xsd
PayTermDtlCre	Payment Terms Detail Create Message	PayTermDesc.xsd
PayTermMod	Payment Terms Modify Message	PayTermDesc.xsd
PayTermDtlMod	Payment Terms Detail Modify Message	PayTermDesc.xsd

Receiving Subscription API

This section describes the receiving subscription API.

Functional Area

Receiving

Business Overview

This API processes receipts that Merchandising receives against purchase orders, transfers, and allocations received at a store, warehouse or finisher. Purchase orders can be received at the item level only using this message. Transfers and allocations, collectively referred to as stock orders, may be received at the bill of lading (BOL) level, where the entire shipment is received without including details, carton level, or item level.

When a transfer, PO or allocation is received at a location, Merchandising will update the appropriate tables, including the shipment, the transfer, allocation, or purchase order, stock on hand, as well as other inventory buckets (for example, in-transit). A record of the receipt is also made in the transaction-level stock ledger.

Receiving Exceptions

Merchandising has the ability to automatically resolve several common exceptions that can occur during the receiving process. Here are some of the exceptions that can be supported:

Additionally, the following exceptions are automatically processed when a stock order is received through this API:

Receipt Against the Wrong BOL

In this case, the receiving location should send a carton status of Dummy (D) or Overage (O), indicating that a dummy BOL number was used. But, even if that status is not used, this exception processing can still take place. Merchandising will attempt to match the contents of the receipt to a valid BOL as follows:

- If the carton belongs to a valid BOL at the given location, Merchandising receives the carton against the intended BOL at the given location.
- If the carton belongs to a valid BOL at a related **walk-through store**, Merchandising receives the carton against the intended BOL at the intended location.
- If the carton belongs to a valid BOL at an unrelated location, Merchandising uses the **wrong store receiving** process.

Walk-through Store Receiving

If you have configured two or more stores as “walk-through” locations, through attribution on the store table in Merchandising then, if the BOL was intended to be received at the walk-through location instead of the location on the message, Merchandising will automatically adjust the receipt and process against the correct location.

Misdirected Container

Misdirected containers, or wrong store receiving, is when one or more containers on a receipt are identified as having been originally shipped to a different location (Location A) than the location that sent the receipt message (Location B). Whether or not misdirected container receiving is supported in Merchandising is controlled by a system option called Wrong Store Receipt Exception Handling. If this option is unchecked (N), then the receipt at the Location B will raise an error in this API. If set to checked (Y), then the shipment at Location A will be backed out, including in-transit updates, WAC adjustments, and stock ledger postings, and re-applied to Location B, prior to processing the receipt into Location B.

In order to correctly manage this processing, Merchandising must receive the original carton number on the receipt. In some cases, such as when integrating with Store Inventory and Operations Cloud Service (SIOCS), the carton ID is reassigned by the receiving location. In that case, the reference carton field in the Receipt Detail node of the message must be populated to trigger this process.

In the event of a carton number being available in the container ID field, the shipment at Location A will be backed out prior to processing the receipt into Location B. If a container ID is not populated, then it will be treated as an unexpected line item on the receipt and a new line item will be inserted against the shipment. The original item line will be corrected only when the original ASN is received (while using the auto-close option) or when the transfer is closed.

Note

Wrong location receiving is supported only for item-level transfer/allocation receipts.

Unwanded Cartons

An unwanded carton is a situation where Merchandising never received notification of the original shipment, only the receipt. In this case, if receiving is done at the item level, Merchandising will process both the shipment and receipt together. If item level details are not included for the carton, an error will be raised, as Merchandising will not be able to determine the contents of the carton, having never received the initial shipment details.

Zero Receipts

This type of exception occurs when a location indicates to Merchandising that nothing was received for the item at the location by sending a receipt of zero and indicating that the carton is closed. Merchandising will reconcile the original ship-to location based on system option settings to determine where to write off the lost items.

If a zero receipt occurs for an item that is part of a misdirected container, then some slightly different processing will occur. If the zero receipt is sent **after** a misdirected container reconciliation, then no further updates will be made, as the line would have been previously reconciled. If a zero receipt occurs **before** misdirected container processing, then the misdirected container processing at the actual receiving location will be treated as an overage, as the original location would have already been reconciled.

Other Key Notes

- Externally generated warehouse-to-warehouse transfers are not supported in Merchandising, where the transfer is created in Merchandising at the physical warehouse level for both locations. For example, a warehouse-to-warehouse transfer created in WMS. This includes the receipt of such a transfer using this API.
- Wrong store receiving is not supported for franchise transactions.
- Merchandising doesn't process the top level of this message (ReceiptDesc) or the UIN level details.

Receipt

This node contains the receipt level details to be processed by Merchandising.

Table 3-99 Receipt

Message Element	Required?	Notes
DC Destination ID	Always	Indicates the location that has processed the receipt. For stores, this will be the store ID. For warehouses, this will be the physical warehouse ID.
PO Number	Always	Indicates the transaction that the receipt is for. This must be a valid allocation, transfer or PO number
Customer Order Number	Optional	Not used in Merchandising
Fulfillment Order Number	Optional	Not used in Merchandising

Table 3-99 (Cont.) Receipt

Message Element	Required?	Notes
Document Type	Always	Specifies whether the receipt is for an allocation (A), purchase order (P), or transfer (T, D, or V).
Reference Document Number	Optional	Contains a reference number for a document associated to the shipment (e.g., Fiscal Document ID for Brazilian based transactions).
ASN Number	Conditional	Used to relate the receipt message to the previous ASN message. This field is required for transfers and allocations.
Receipt Type	Optional	This field is used in receiving transfers or allocations to determine if the receipt is at the BOL level (BL) or SKU level (SK). If not provided, the value will be defaulted to SK. It is not used for a PO receipt.
From Location	Optional	This field contains the source location of the shipment.
From Location Type	Optional	This field contains the source location type of the shipment. Valid values are: • SU - supplier • ST - store • WH - warehouse
Status	Optional	Indicates the status of the ASN received. This field is used only for stock order receiving. A status of C indicates that the entire ASN is will be set to closed.

Child Nodes

- Receipt Detail
- Receipt Carton Detail

Receipt Detail

This is a required child node to the receipt level only for item level stock order receipts. For carton level receipts, this should not be populated.

Table 3-100 Receipt Detail

Message Element	Required?	Notes
Item ID	Always	Specifies the item on the allocation, purchase order, or transfer that has been received.
Unit Quantity	Always	Contains the quantity received for the allocation, purchase order, or transfer in the standard unit of measure.
Receipt Transaction Type	Always	Specifies whether the receipt detail line item is for a Receipt (R), Transshipment (T), or Adjustment (A).
Receipt Date	Optional	Identifies the date on which the transaction was received.

Table 3-100 (Cont.) Receipt Detail

Message Element	Required?	Notes
Receipt Number	Optional	An externally generated numerical identifier corresponding to the receipt of the item at the location.
Destination ID	Optional	Only used for purchase order receipt, when the purchase order has an allocation attached to it. This element specifies the location to which the allocation is being sent.
Container ID	Optional	Identifies the carton number for the item being received.
Reference Container ID	Optional	Identifies the original carton number the item was shipped under, if it was being received at the wrong destination. This is required by Merchandising to process the updates correctly, as SIOCS reassigns the container ID at the receiving location.
Distro Number	Optional	Only used for purchase order receipts, when the purchase order has an allocation attached to it. This element contains the allocation id.
Distro Document Type	Optional	Only used for purchase orders, when the purchase order has an allocation attached to it. When populated, this value should always be A to specify an allocation.
From Disposition	Optional	This value is used to determine inventory availability. Valid values are in the INV_STATUS_CODES table. The from disposition is used when the to disposition is not provided.
To Disposition	Optional	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.
To WIP	Optional	Not used in Merchandising
From WIP	Optional	Not used in Merchandising
To Trouble	Optional	Not used in Merchandising
From Trouble	Optional	Not used in Merchandising
User ID	Optional	Not used in Merchandising
Dummy Carton	Optional	Indicates if this carton is a dummy carton. This field is only used for transfer receipts. Valid values are yes (Y) or no (N).
Tampered Carton	Optional	Indicates if the carton was tampered. This field is only used for transfer receipts. Valid values are yes (Y) or no (N).
Unit Cost	Optional	Used for purchase order receipts when configured for Global Tax (GTS) where the location's country is localized. Otherwise, it will be ignored except in scenarios where there is an unexpected item on the receipt, in which case it can be optionally included to indicate the receipt cost of the item. If not included for unexpected receipts, the default supplier cost will be used. Cost is converted to the order's currency before insert/update.

Table 3-100 (Cont.) Receipt Detail

Message Element	Required?	Notes
Shipped Quantity	Optional	Only used for purchase order receipts. Updates the number of items expected to be received, originally set by the ASN process.
Weight	Optional	Contains the actual weight of the item received for the shipment. This will be included for some catch weight items.
Weight UOM	Optional	Contains the unit of measure of the received weight (e.g., pounds, kilograms) where UOM class is of type MASS. Weight and Weight UOM must both be populated, or both must be NULL.
Gross Cost	Optional	Contains the Unit cost and Expenses incurred on an item in a particular transaction.
Item Line Number	Optional	This field indicates the item line number from customer orders. This must be populated for systems with OMS_IND = Y and RESV_CO_ON_RECEIPT = Y.
Inventory Identifier Type	Optional	This fields contains the inventory identifier type field which is passed during receipt message. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, include Lot (L), Expiry Date (E), Import Document (D).
Inventory Id	Optional	This fields contains the inventory identifier ID provided on receipt transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Receipt Carton Detail

This node is only used for stock order receiving as an alternative to the item level node.

Table 3-101 Receipt Carton Detail

Message Element	Required?	Notes
Carton Status	Conditional	Indicates the status of the carton being received. Valid values are: - Actual (A) – this is used for normal carton receiving at the correct location - Overage (O) – this should be used if the carton doesn't belong to the BOL/location being received. In this case, Merchandising will attempt to match the container to the correct BOL. See receiving exceptions above. - Dummy BOL (D) – this should be used if the BOL used in the receipt is not valid for the carton/location. In this case, Merchandising will attempt to match the container to the correct BOL. See receiving exceptions above. - Closed (C) – indicates that the carton is closed and no more receipts are expected. In this case, Merchandising will reconcile any outstanding quantity for the carton immediately.
Container ID	Conditional	Indicates the container being received. This must be populated for carton level receiving.
Reference Container ID	Optional	Not being used in Merchandising.
Destination ID	Optional	Identifies the location to which the stock order is received.
Receipt Transaction Type	Required	Specifies whether the carton receipt is for a Receipt (R), Transshipment (T), or Adjustment (A).
Receipt Date	Optional	This field contains the date when the carton is received.
Receipt Number	Optional	This field holds the externally generated number when the shipment is received at the location.
User ID	Optional	Not used by Merchandising.
To Disposition	Optional	This value is used to determine inventory availability of the received quantity. Valid values are in the INV_STATUS_CODES table.
Weight	Optional	Contains the actual weight of the item received for the carton. This is used for catch weight containers.
Weight UOM	Optional	Contains the unit of measure of the received weight (e.g., pounds, kilograms) where UOM class is of type MASS. Weight and Weight UOM must both be populated, or both must be NULL.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
receiptcre	Receipt Create Message	ReceiptDesc.xsd
receiptordcre	Receipt Create Message for Customer Orders	ReceiptDesc.xsd
receiptmod	Receipt Modify (Adjustment) Message	ReceiptDesc.xsd

RTV Subscription API

This section describes the RTV subscription API.

Functional Area

Returns to Vendor

Business Overview

Merchandising subscribes to return-to-vendor (RTV) messages when an RTV is shipped out from a warehouse or store. This shipment could be for an RTV that was initially created in Merchandising or one initiated in the store or warehouse. The RTV information is sent from a warehouse management system (WMS), such as Oracle WMS Cloud, or the store inventory solutions, such as Oracle Retail Store Inventory and Operations Cloud Service (SIOCS) when the RTV is shipped out of the location.

Note

Unlike other RIB messages, both new and updates sent through this message use the RTVCre message type.

New RTVs

If the message contains a new RTV generated in the store or warehouse, then it must contain both header and detail information. RTV create messages can only be sent in Approved or Shipped status.

Updated RTVs

If this is an update to an RTV, it can be performed through this API. To update an RTV, you can send either the header information only or both header and detail information. The most common update is to ship a previously created RTV. Approved RTVs can be Shipped if the RTV is created in Merchandising or SIOCS.

It is assumed that RTVs from the warehouse are always created in Shipped status.

Note

Once RTVs are shipped, they cannot be changed back to Approved. Alternatively, approved RTVs can also be set to Cancelled status, if for some reason they cannot be shipped.

RTV Header

Message Element	Required	Notes
Destination ID	Always	Contains the location shipping the RTV - either the store ID or the physical warehouse ID.
RTV ID	Always	This contains the external reference number for RTVs created outside of Merchandising.
Return Authorization Number	Optional	Contains the number that the supplier provides when the decision is made that the merchandise may be returned. This will be required depending on the configuration of the supplier site in Merchandising.
Vendor Number	Always	Contains the supplier site ID to which the merchandise is being returned. The site must be configured in Merchandising to allow returns.
Address 1	Optional	Contains the first line of the supplier's address for returns. If not provided, the address for an externally created RTV will be pulled from the supplier address information in Merchandising. This applies for all of the below address fields as well.
Address 2	Optional	Contains the second line of the supplier's address for returns.
Address 3	Optional	Contains the third line of the supplier's address for returns.
State	Optional	Contains the state of the supplier's address for returns. The state and country combination must be valid.
City	Optional	Contains the city of the supplier's address for returns.
Postal Code	Optional	Contains the postal code of the supplier's address for returns.

Message Element	Required	Notes
Country	Optional	Contains the country ID of the supplier's address for returns. The state and country combination must be valid.
Creation Time Stamp	Optional	Contains the date the vendor return was created. This defaults to current date if not specified for an externally generated RTV.
Comments	Optional	Contains any comments associated with the return.
RTV Order Number	Optional	This contains the RTV ID generated by Merchandising when the event was created. For externally generated RTVs, this should be NULL.
Status	Optional	This value is used to determine the current status of an externally generated RTV. Valid values are Approved (A) or Shipped (S). If this is Approved, Merchandising will create the RTV and set it to an In Progress status. If this is Shipped or null, it will be set to Shipped status when created or updated.
Consignment Vendor RTV	Optional	This value is used to determine whether the RTV is a consignment vendor RTV or not. Valid values are: Y and N. If it is set to Y, then the RTV is a consignment vendor RTV and the origin is set to CONVEN. If it is N, then the RTV is not a consignment vendor RTV.

Child Node

- RTV Details
- Flex Attribute

RTV Details

Message Element	Required	Notes
Item ID	Always	Unique identifier for the item on the RTV. The item must be a transaction level inventory item or reference item.

Message Element	Required	Notes
Unit Quantity	Always	Contains the quantity of the item being returned to the supplier under this RTV number. A quantity less than zero indicates a quantity cancellation. If the RTV is already in progress, the quantity in the message must not be zero.
Container Quantity	Optional	Not used in Merchandising
From Disposition	Optional	This value is used to determine if the inventory is available or unavailable. Valid values are based on the INV_STATUS values in the INV_STATUS_CODES table.
To Disposition	Optional	Not used in Merchandising
Unit Cost	Optional	Contains the cost per unit for the SKU being returned. This field is stored in the supplier's currency. If not provided, then value will be defaulted based on the system option RTV Unit Cost Source setting - either the from location's weighted average cost, last received cost, or standard cost.
Reason	Optional	Identifies the reason for the return. Valid values are for this field are held in the Merchandising codes tables under code type RTVR. By default, the values are: Q - QC Failed U - Unavailable Inventory. W - Externally Initiated RTV If the reason is not provided for an externally generated RTV, it will be defaulted to W.
Weight	Optional	Actual weight shipped for the item on the RTV, which is used for some catch weight items. It is expected that either both weight and weight UOM have a value or neither have a value.
Weight UOM	Optional	Indicates the unit of measure represented for the provided weight (for example, pounds, kilograms) where UOM class is of type MASS. It is expected that either both weight and weight UOM have a value or neither have a value.

Message Element	Required	Notes
Gross Cost	Optional	Contains the unit cost and expenses incurred on an item in a particular transaction. This is used for Brazil processing only. For all implementations not using the Brazil Localization configuration, this should be NULL.
RTV Detail UIN	Optional	Not used in Merchandising
Inventory Identifier Type	Optional	This field contains the inventory identifier type field, which is passed during return to vendor message. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
Inventory Id	Optional	This fields contains the inventory identifier ID provided on return to vendor transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Flex Attributes

If flex attributes have been defined for an RTV they can be included in this node of the message.

Message Element	Required	Notes
Name	Always	The flex attribute name defined by the business
Value	Optional	The value of the flex attribute defined by the business for alphanumeric or number attributes.
Value Date	Optional	The date value of the flex attribute, if the flex attribute is defined as a date.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
rtvcre	RTV Create Message	RTVDesc.xsd

Stock Count Schedule Subscription API

This section describes the stock count schedule subscription API.

Functional Area

Inventory

Business Overview

Stock count schedule messages are published to the RIB by an inventory sub-system, such as Oracle Retail Store Inventory and Operations Cloud Service (SIOCS), to communicate unit and value stock count schedules to Merchandising. Merchandising uses stock count schedule data to help synchronize the inventories of the integrated system and Merchandising. The integrated system then performs a physical inventory count and uploads the results, and Merchandising compares the discrepancies.

This API allows the external systems to create, update, and delete Unit and Value stock count requests within Merchandising. The count is assumed to be for the full location, unless department, class and subclass data are included.

Creating/Updating Stock Count Requests

When a new stock count request is created or an existing stock count request is modified, this API will validate all the required fields are present in the message. Required information for the stock count includes a description, date, type (always B), location type, and locations. Optionally the merchandise hierarchy information can also be included, but, if not included, it will be assumed the entire location will be counted. After the required field and business validations, the stock counts will be created or updated in Merchandising.

The format used when creating or modifying a stock count is shown below.

Table 3-102 Stock Count Header

Message Element	Required?	Notes
Cycle Count	Yes	This field contains the number which uniquely identifies the stock or cycle count.
Cycle Count Description	Yes	This field contains a description of the cycle or stock count which, along with the cycle count number, identifies the cycle or stock count.

Table 3-102 (Cont.) Stock Count Header

Message Element	Required?	Notes
Location Type	Yes	This field contains an indicator which identifies whether the cycle count will be for Stores or Warehouses. Valid values are Store (S) and Warehouse (W).
Stocktake Date	Yes	This field contains the date on which the stock or cycle count will take place. This date should be greater or equal to the current date plus the stake lockout days defined at system level.
Stocktake Type	Yes	This field contains a value which indicates the stock take type. Valid values are Both Unit and Dollar (B) and Unit only (U). However, the RIB interface only allows modification of Unit and Dollar stock take types.
Stock Count Schedule Products	No	Child node
Stock Count Schedule Locations	No	Child node

Table 3-103 Stock Count Schedule Products

Message Element	Required?	Notes
Department	Yes	This field contains the department number where the cycle count will occur. If the value = -1, the stock count will apply to all departments. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.
Class	No	This field contains the class number where the cycle count will occur. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.
Subclass	No	This field contains the subclass number where the cycle count will occur. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.

Table 3-104 Stock Count Schedule Locations

Message Element	Required?	Notes
Location	Yes	This field contains the store or warehouse number on the cycle count. This must be a valid store or a stockholding warehouse in Merchandising.

Deleting Stock Count Requests

When an existing stock count request is deleted, this API will validate all the required fields are present in the message. After required field and business validation, the stock counts will be removed in Merchandising. This API also supports deleting a location from the count. The count and locations can only be deleted through this API if no results have been processed for the location on the count. If the last location is deleted from the count, then the count itself will be deleted.

The format used when deleting a stock count is shown below.

Table 3-105 Stock Count Header

Message Element	Required?	Notes
Cycle Count	Yes	This field contains the number which uniquely identifies the stock or cycle count being deleted.
Stock Count Schedule Locations	No	Child node

Table 3-106 Stock Count Schedule Locations

Message Element	Required?	Notes
Location	Yes	This field contains the store or warehouse being deleted on the cycle count.

Error Handling

If an error occurs in this procedure, a call will be placed to a function to build a complete error message. This message together with a status of E is returned to the external system. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the RIB documentation for each message type in order to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
stkcountschcre	Stock Count Schedule Create Message	StkCountSchDesc.xsd
stkcountschmod	Stock Count Schedule Modify Message	StkCountSchDesc.xsd
stkcountschdel	Stock Count Schedule Delete Message	StkCountSchRef.xsd
stkcountschtdel	Stock Count Schedule Delete Message for a Location	StkCountSchRef.xsd

Stock Order Status Subscription API

Functional Area

Inventory Movement

Business Overview

A stock order is an outbound merchandise request from a warehouse or store. In Merchandising, a stock order takes the form of either a transfer or allocation. Merchandising subscribes to stock order status messages published by an external application, such as a store system (SIOCS, for example) or a warehouse management system (Oracle WMS Cloud, for example) to communicate the status of a specific stock order.

Stock Order Statuses

The following tables describe the stock order statuses for both transfers and allocation document types and what occurs in Merchandising after receiving the respective status. Statuses other than listed below are ignored by Merchandising.

Document types of T, D, and S all refer to transfers and indicate whether the transfer is initiated in Merchandising, a warehouse system, or a store system, respectively.

Table 3-107 Statuses for Document Types T, D, and S

Stock Order Status	What Merchandising does
SI (Stock Increased)	Insert or increase transfer quantity and increase item/location transfer reserve quantity for the source location and transfer expected quantity for the destination location.
SD (Stock Decreased)	Delete or decrease the transfer quantity for the transfer/item combination. Transfer quantity for the transfer/item combination will be deleted if the transfer has been created but has not been shipped. Additionally, the item/location transfer reserved quantity for the source location and the transfer expected quantity for the destination location will be decreased.
DS (Details Selected)	Increase the selected quantity for the transfer/item combination.
DU (Details Un-selected)	Decrease the selected quantity for the transfer/item combination.
NI (WMS Line Cancellation)	Decrease selected and transfer quantity for the transfer/item by the quantity on the message and increase the cancelled quantity. Additionally, it will decrease the reserved quantity for the source location and decrease the expected quantity for the destination location by the lesser of the quantity on the message and transfer - shipped quantity. The transfer will also be added to the document close queue if transfer status is not closed. Document Close batch program will then determine if the transfer should be closed based on certain conditions. Transfers with outstanding appointments are not closed.
PP (Distributed)	Decreases the selected quantity and increases the distro quantity for the transfer/item.
PU (Un-Distribute)	Decreases the distro quantity for the transfer/item.
RS (Return to Stock)	Decreases distro quantity and transfer quantity for the transfer/item; the cancelled quantity for the transfer/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the transfer - shipped quantity if the transfer status is not closed.

Table 3-107 (Cont.) Statuses for Document Types T, D, and S

Stock Order Status	What Merchandising does
EX (Expired)	Decreases transfer quantity for the transfer/item; the cancelled quantity for the transfer/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the transfer - shipped quantity if the transfer status is not closed. The transfer will also be added to the document close queue if transfer status is not closed. Document Close batch program will then determine if the transfer should be closed based on certain conditions. Transfers with outstanding appointments are not closed.
SR (Store Reassign)	Updates the distro quantity for the transfer/item. This can either increase or decrease the value, depending on whether a positive or negative value is sent.

Document type A is always used for Allocations.

Table 3-108 Statuses for Document Type A

Stock Order Status	What Merchandising does
SI (Stock Increased)	Insert or increase allocated quantity and increase item/location transfer reserve quantity for the source location and transfer expected quantity for the destination location.
SD (Stock Decreased)	Decrease the allocated quantity for the allocation/item combination. Additionally, the item/location transfer reserved quantity for the source location and the transfer expected quantity for the destination location will be decreased.
DS (Details Selected)	Increase the selected quantity for the allocation/item combination.
DU (Details Un-Selected)	Decrease the selected quantity for the allocation/item combination.
NI (WMS Line Cancellation)	Decrease selected and allocation quantity for the allocation/item by the quantity on the message and increase the cancelled quantity. Additionally, it will decrease the reserved quantity for the source location and decrease the expected quantity for the destination location by the lesser of the quantity on the message and allocation - shipped quantity if the allocation is not closed. The allocation will also be added to the document close queue if allocation status is not closed. Document Close batch program will then determine if the allocation should be closed based on certain conditions. Allocations with outstanding appointments are not closed.
PP (Distributed)	Decreases the selected quantity and increases the distro quantity for the allocation/item.
PU (Un-Distribute)	Decreases the distro quantity for the allocation/item.

Table 3-108 (Cont.) Statuses for Document Type A

Stock Order Status	What Merchandising does
RS (Return to Stock)	Decreases distro quantity and allocation quantity for the allocation/item; the cancelled quantity for the allocation/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the allocation - shipped quantity if the allocation status is not closed.
EX (Expired)	Decreases allocation quantity for the allocation/item; the cancelled quantity for the allocation/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the allocation - shipped quantity if the allocation status is not closed. The allocation will also be added to the document close queue if allocation status is not closed. Document Close batch program will then determine if the allocation should be closed based on certain conditions. Allocations with outstanding appointments are not closed.
SR (Store Reassign)	Updates the distro quantity for the allocation/item. This can either increase or decrease the value, depending on whether a positive or negative value is sent.

For customer orders, Merchandising assumes it will receive updates from an OMS for customer-order-related stock orders. Therefore, to avoid duplicate processing, Merchandising will ignore No Inventory, Expired, Stock Decreased, and Stock Increased statuses received for a customer order transfer.

Stock Order Status Message Details

The table below summarizes the elements applicable for this API.

Table 3-109 Stock Order Status Header

Message Element	Required?	Notes
DC Destination ID	Always	This field contains the location number of the stock order source location.
Location Type	Always	This field contains the type of location in the DC Destination ID field. Valid values are: • S - Store • W - Warehouse • E - Finisher
Store Type	Optional	This field indicates the store type of the DC Destination ID if it is a store. Valid values are: • C - company store • F - franchise store
Stockholding Indicator	Optional	This field indicates if the DC Destination ID is stockholding or not. Only populated if location type is 'S'. Valid values are Yes (Y) and No (N).

Table 3-109 (Cont.) Stock Order Status Header

Message Element	Required?	Notes
Distro Number	Optional	This field contains the stock order number. This is either the transfer or allocation number in Merchandising.
Distro Document Type	Always	This field specifies whether the stock order status pertains to an allocation (A) or transfer (T) that is already existing in Merchandising. Created Stock Order (D), Customer Order (C), and Virtual Distro (V) are also valid document types but will be ignored by Merchandising.
Context Type	Optional	This field holds the functional area code to which the transfer relates to. Valid values are Promotion (PROM) and Repairing (REPAIR).
Context Value	Optional	This field holds the value relating to the context type like promotion number.
Inventory Type	Optional	This field indicates if a transfer is made from the available (A) or unavailable (U) inventory.
Customer Order Number	Optional	This field holds the master customer order number for a stock order associated with a customer order.
Fulfillment Order Number	Optional	This field holds the number related to the fulfillment details for a stock order associated with a customer order. One or more fulfillment orders could relate back to a single customer order.
System Code	Optional	Not used by Merchandising
From Location Virtual WH	Optional	This field contains the virtual warehouse number of the stock order source location.
Stock Order Status Details	Optional	Child node
Custom Flex Attributes	Optional	Child node – not used by Merchandising

Table 3-110 Stock Order Status Detail

Message Element	Required?	Notes
Destination ID	Always	This field contains the location number of the stock order receiving location.
Location Type	Always	This field contains the type of location in the Destination ID field. Valid values are: • S - Store • W – Warehouse • E - Finisher
Store Type	Optional	This field indicates the store type of the Destination ID, if a store. Valid values are: • C - Company Store • F – Franchise Store
Stockholding Indicator	Optional	This field indicates if the Destination ID is stockholding. Only populated if location type is 'S'. Valid values are: Yes (Y) and No (N).
Item	Always	This field contains the unique identifier for the item.

Table 3-110 (Cont.) Stock Order Status Detail

Message Element	Required?	Notes
Original Item	Optional	This field contains the ID of the item being replaced. Populated only when this record is for a substitute item on a customer order.
Order Line Number	Optional	This field is used to carry the customer order line number value for customer orders. It is a derived value for non-customer orders.
Unit Quantity	Optional	This field contains the difference between the number of item units shipped versus the receiving count for the given item. This is subtracted from the document-line-item-unit-count to yield an over/under variance between what a supplier said was shipped and what was counted and received at by the store's staff.
Status	Optional	This field contains the status of the stock order. Valid values are: Accepted (SI), Rejected (SD), Distributed (PP), Un-Distributed (PU), Details Selected (DS), Details Un-selected (DU), WMS Line Cancellation (NI), Return To Stock (RS), Expired (EX), and Store Reassign (SR). Statuses other than listed are ignored by Merchandising. For stock order status explanations, please refer to the Business Overview section .
User ID	Optional	This field contains the user ID of the user who created the stock order status.
Updated Date	Optional	This field contains the date the stock order status was updated.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
sostatuscre	Stock Order Status Create Message	SOStatusDesc.xsd
CustFlexAttriVo	Stock Order Status Flex Attribute Create	SOStatusDesc.xsd

Store Subscription API

This section describes the store subscription API.

Functional Area

Foundation Data

Business Overview

The Store Subscription API provides the ability to keep store data in Merchandising in sync with an external system if Merchandising is not being used as the system of record for organizational hierarchy information. The store data handled by the API includes basic store data in addition to addresses, store hours, location traits, up-charges, and walk-through stores. Walkthrough stores are used in Merchandising as part of the transfer reconciliation process and are used to indicate two or more stores that have a 'walk through' connection between them - on the sales floor and/or the backroom.

Create or Update Stores

When creating a new store in Merchandising through this API, the data is validated and placed onto a staging table `STORE_ADD`. The store creation in Merchandising reads from this table and creates the store in Merchandising in an asynchronous mode. When updating an existing store in Merchandising, the API performs the update immediately upon message receipt.

Delete Stores

The API also handles store delete messages. But, like the store creation message subscription process, stores will not actually be deleted from the system upon receipt of the message. After the data has been validated, the store is added to the `DAILY_PURGE` table for processing through a batch process.

Other Store Attributes

Location traits, walkthrough stores, department-level up-charges, store hours, address data and CFAs must be processed separately as they each have their own distinct message types for creation and deletion.

Location traits and walkthrough store attributes cannot be sent in a store create message; the store create batch must first process the store before it can have these attributes attached to it. Up-charges, store hours, and addresses can be included for new stores.

The store subscription message also supports the ability to import custom flex attributes from an external system.

Other Notes

- Location traits must already exist prior to being added to the store.
- Stores and warehouses in Merchandising cannot have the same unique identifier.
- Location trait and walkthrough store data cannot be sent in a store create message.

Store Message Details

The table below summarizes the elements applicable for this API.

Table 3-111 Store Create or Update

Message Element	Required?	Notes
Store	Always	This field contains the unique identifier of the store.
Store Name	Always	This field contains the name of the store.
Store Type	Optional	Indicates the type of store. Valid values are: company store (C), franchise store (F).
Store Name (10)	Optional	This field contains the ten-character abbreviation of the store name.
Store Name (3)	Optional	This field contains the three-character abbreviation of the store name
Store Class	Always	This field contains code of the class of which the store is a member. Valid values are A through E.
Store Manager Name	Always	This field holds the name of the store manager.
Store Open Date	Always	This field holds the date on which the store opened.
Store Close Date	Optional	This field holds the date on which the store closed.
Acquired Date	Optional	This field holds the date on which the store was acquired.
Remodel Date	Optional	This field holds the date on which the store was remodeled.
Fax Number	Optional	This field contains the fax number for the store.
Phone Number	Optional	This field contains the phone number for the store.
Email	Optional	This field contains the email address of the store.
Total Square Feet	Optional	This field contains the total square footage of the store.
Selling Square Feet	Optional	This field contains the total square footage of the store's selling area.
Linear Distance	Optional	This field contains the total merchandise space of the store.
Stockholding Indicator	Always	This field indicates if the store can hold stock. Valid values are yes (Y) or no (N). This field cannot be modified.
Channel ID	Always	This field contains the identifier of the channel. This value must be predefined on the CHANNELS table.
Store Format	Optional	This field contains the code of the store format of the store. This value must be predefined on the STORE_FORMAT table.
Mall Name	Optional	This field contains the name of the mall in which the store is located.
District	Always	This field contains the number of the district of which the store is a member. This value must be predefined on the DISTRICT table.
Promo Zone	Optional	Not used by Merchandising.

Table 3-111 (Cont.) Store Create or Update

Message Element	Required?	Notes
Transfer Zone	Always	This field contains the transfer zone in which the store is located. This value must be predefined on the TSFZONE table.
Default WH	Optional	This field contains the default warehouse for the store. This value must be a virtual warehouse predefined on the WH table.
Stop Order Days	Optional	This field contains the number of days before the store close date that the store will stop accepting orders.
Start Order Days	Always	This field contains the number of days before the store open date that the store will begin accepting orders.
Currency Code	Always	This field contains the currency code under which the store operates. This value must be predefined on the CURRENCIES table. It cannot be modified.
Language	Optional	This field contains the code of the language used at the store. This value must be predefined on the LANG table. Either ISO code or Lang should not be null. If this field is null, ISO Code will be used for looking up the language to be used.
ISO Code	Optional	This field contains the International Organization for Standards (ISO) character code corresponding to the language used at the store. Either ISO code or Lang should not be null.
Integrated POS Indicator	Always	This field indicates whether Sales Audit should expect files from this store for processing. Valid values are Yes (Y) or No (N).
DUNS Number	Optional	This field holds the Dun and Bradstreet number to identify the store.
DUNS Location	Optional	This field holds the Dun and Bradstreet number to identify the location.
Copy Delivery Indicator	Conditional	This field indicates if the like store's delivery schedule information should be copied to the new store. This value cannot be modified. It will only be used on a store create message.
Copy Activity Indicator	Conditional	This field indicates if the like store's closing date schedule should be copied to the new store. This value cannot be modified. It will only be used on a store create message.
Price Store	Conditional	This field contains the store from which pricing information will be copied to the new store. The pricing store does not need the same currency as the new store.
Cost Location	Conditional	This field contains the location from which to copy cost information to the new store. This field should only be included on store create messages. This value must be existing store or virtual warehouse.
VAT Include Indicator	Optional	This field contains whether tax will be included in the retail prices for the store. Valid values are Yes (Y) or No (N).

Table 3-111 (Cont.) Store Create or Update

Message Element	Required?	Notes
VAT Region	Conditional	This field contains the ID of the tax region the store is associated with. This value must be predefined on the VAT_REGION table. It is required if Merchandising is configured for Simple VAT or Global Tax.
Like Store	Conditional	This field holds the store from which the new store will have item/locations copied. This value must be predefined on the STORE table. It cannot be modified and will only be used in a create message.
Copy Replenishment Indicator	Conditional	This field indicates whether replenishment information should be copied from the like store to the new store. This field cannot be modified. It will only be populated on a store create message.
Transfer Entity	Conditional	This field contains the transfer entity of which the store is a part. This value must be predefined on the TSF_ENTITY table. If the system allows intercompany transfers this field is required.
Sister Store	Optional	This field contains the store which will be used to relate historical data to the new store in Allocation. This value must be predefined on the STORE table.
Transaction Number Generated	Always	This field holds the level at which unique POS transaction numbers are generated. Valid values are Store (S) and Register (R).
County	Optional	This field contains the county in which the store is located.
Time Zone Name	Optional	This field contains the Time Zone name.
WF Customer ID	Optional	This field contains the ID associated with a franchise customer for a franchise store. This is required for a franchise store type and should be null for a company store.
Org Unit ID	Optional	This field contains the organizational unit ID value of the store.
Secondary Store Name	Conditional	This field contains the secondary name of the store.
Customer Order Location Indicator	Optional	Indicates that the store can be used to source or fulfill customer orders. Valid values are yes (Y) or no (N).
Gift Wrapping Indicator	Optional	This field indicates whether gift wrapping is required.
Customer Order Ship Indicator	Optional	This field indicates whether the customer order has been shipped from Warehouse.
Tax Id	Optional	Contains the unique tax identification number of the store.
Online Store Indicator	Optional	This field indicates how store day will be managed by ReSA. If the indicator is Y then ReSA will automatically open and close the store day.
Store Location Trait	Optional	Child node
Walkthrough Store	Optional	Child node
Store Address	Optional	Child node

Table 3-111 (Cont.) Store Create or Update

Message Element	Required?	Notes
Store Hours	Optional	Child node
Store Department Up-charge	Optional	Child node
Custom Flex Attributes	Optional	Child node

Table 3-112 Store Location Trait Create

Message Element	Required?	Notes
Location Trait	Conditional	The identifier of the location trait. Though the node is optional, this field is required if the node is included. The node cannot be populated in the store create message.

Table 3-113 Walkthrough Store Create

Message Element	Required?	Notes
Walkthrough Store	Conditional	A walk-through store of the store being modified. Though the node is optional if it is included this field is required. This node cannot be populated in a store create message.

Table 3-114 Store Address Create or Update

Message Element	Required?	Notes
City ID	Optional	This field contains the city ID or code.
State Name	Optional	This field contains the state name.
Country Name	Optional	This field contains the country name.
Address	Always	This field holds the unique address ID from the source system.
Address Type	Always	This field indicates the type for the address. Valid values are in the ADD_TYPE table. Mandatory address types for a new store are Business (01) and Postal (02).
Primary Address Indicator	Always	This field indicates whether this address is the primary address for this address type. At least one address must be designated as primary for each type.
Address 1	Always	This field contains the first line of the address.
Address 2	Optional	This field contains the second line of the address.
Address 3	Optional	This field contains the third line of the address.
City	Always	This field contains the name of the city that is associated with the address.
State	Optional	This field contains the abbreviation for the state in which the store is located.

Table 3-114 (Cont.) Store Address Create or Update

Message Element	Required?	Notes
Country ID	Always	This field contains the country code where the address exists.
Post	Optional	This field contains the postal code for the address.
Contact Name	Optional	This field contains the name of the contact for the supplier at this address.
Contact Phone	Optional	This field contains the phone number of the contact person at this address.
Contact Telex	Optional	This field contains the telex number of the store's representative contact.
Contact Fax	Optional	This field contains the fax number of the contact person at this address.
Contact Email	Optional	This field contains the email address of the store's representative contact.
Oracle Vendor Site ID	Conditional	This field contains the unique identifier of this address in the Oracle Financials, if used.
County	Optional	This field contains the county name for the location.
Jurisdiction Code	Optional	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
Custom Flex Attributes	Optional	Child node

Table 3-115 Store Hours Create or Update

Message Element	Required?	Notes
Store ID	Always	This field contains the unique ID of the store.
Day Number	Always	This field indicates the day of the week for which store timing is being stored. Valid values are from 1 (Sunday) through 7 (Saturday).
Store Open Time	Optional	This field contains the open time for the store. Open time should be in 12-hour format (HH:MI AM or HH:MI).
Store Close Time	Optional	This field contains the close time for the store. Close time should be in 12-hour format (HH:MI AM or HH:MI).

Table 3-116 Store Department Up-charge Header Create or Update

Message Element	Required?	Notes
Hierarchy Level	Always	The hierarchy level for which the up charges are being added or updated. Valid values are Division (DI), Group (GR), Department (DE), and All Departments (AD).
Hierarchy	Optional	The value of the hierarchy based on the level. If the level is division, group, or department, then this should contain a valid ID of that type. If All Departments is the hierarchy level, then this should be left blank.

Table 3-116 (Cont.) Store Department Up-charge Header Create or Update

Message Element	Required?	Notes
From Location	Optional	Contains the source location for which the up charges for the hierarchy will apply. Valid values are a country, area, region, store, virtual warehouse, or physical warehouse ID, depending on the from location type included. If the from location type sent is AS or AW, this should be left null.
To Location	Optional	Contains the destination location for which the up charges for the hierarchy will apply. Valid values are a country, area, region, store, virtual warehouse, or physical warehouse ID, depending on the from location type included. If the from location type sent is AS or AW, this should be left null.
From Location Type	Always	Indicates the type of location included as the From Location. Valid values are Country (C), Area (A), Region (R), Store (S), Virtual Warehouse (W), Physical Warehouse (PW), All Stores (AS), or All Warehouses (AW).
To Location Type	Always	Contains the type of location included as the To Location. Valid values are Country (C), Area (A), Region (R), Store (S), Virtual Warehouse (W), Physical Warehouse (PW), All Stores (AS), or All Warehouses (AW).
Store Department Up-charge Details	Optional	Child node

Table 3-117 Store Department Up-charge Details Create or Update

Message Element	Required?	Notes
Component ID	Always	This field contains the unique identifier of the up-charge component.
Component Rate	Always	This field contains the rate to be charged based on the transfer/allocation cost for items in the hierarchy and between from and to locations defined in the up-charge header.
Per Count	Optional	If the component was defined with a calculation basis of Specific, then this field is used to determine how the rate is applied to the items on a transfer or allocation. If the component is defined as Value, this should be blank.
Per Count UOM	Optional	Indicates the unit of measure in which the Per Count is specified.
Up-charge Group	Always	This field contains the group to which the component ID belongs. Valid values can be found on the codes table with a code type of UCHG.
Component Currency	Always	This field contains the currency of the Up Charge component.
Effective Date	Optional	The date from which the new values are effective in the system.

Table 3-117 (Cont.) Store Department Up-charge Details Create or Update

Message Element	Required?	Notes
Item Default Indicator	Optional	By default, up-charges defined will be defaulted for new items created in the hierarchy designated at the header level. But this indicator allows you to add the new or updated upcharge to an existing item in the hierarchy. Valid values are Y or N.
Transfer Allocation Default Indicator	Optional	Indicates if component information should be applied to any unshipped transfers or allocations with items and locations that qualify. Valid values are Y or N.
Computation Value Base (CVB)	No	This field contains the Computation Value Basis for the Up Charge component at department level.
Cost Basis	No	This field contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
Include in Total Up Charge Indicator	No	This field contains the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level. Valid values are Y or N.

Table 3-118 Store Delete

Message Element	Required?	Notes
Store	Always	The store number being deleted, or for which a location trait or walk through store is being disassociated.
Store Location Trait	Optional	Child node
Walkthrough Store	Optional	Child node
Store Address	Optional	Child node
Store Hours	Optional	Child node
Store Department Up-charge	Optional	Child node

Table 3-119 Store Location Trait Delete

Message Element	Required?	Notes
Location Trait	Conditional	The identifier of the location trait. Though the node is optional, this field is required if the node is included. The node cannot be populated in the store create message.

Table 3-120 Walkthrough Store Delete

Message Element	Required?	Notes
Walkthrough Store	Conditional	A walk-through store of the store being modified. Though the node is optional if it is included this field is required. This node cannot be populated in a store create message.

Table 3-121 Store Address Delete

Message Element	Required?	Notes
Address	Always	The unique identifier of the address being deleted.

Table 3-122 Store Hours Delete

Message Element	Required?	Notes
Store ID	Always	This field contains the unique ID of the store.
Day Number	Always	This field indicates the day of the week for which store timing is being stored. Valid values are from 1 (Sunday) though 7 (Saturday).

Table 3-123 Store Department Up-charge Header Delete

Message Element	Required?	Notes
Hierarchy Level	Always	The hierarchy level for which the up charges are being deleted. Valid values are Division (DI), Group (GR), Department (DE), All Departments (AD).
Hierarchy	Optional	The value of the hierarchy.
From Location	Optional	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse, or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.
To Location	Optional	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
From Location Type	Always	Contains the type of source location from which goods will be transferred. Valid values are Country (C), Area (A), Region (R), Store (S), Virtual Warehouse (W), Physical Warehouse (PW), All Stores (AS), or All Warehouses (AW).

Table 3-123 (Cont.) Store Department Up-charge Header Delete

Message Element	Required?	Notes
To Location Type	Always	Contains the type of destination location to which goods will be transferred. Valid values are Country (C), Area (A), Region (R), Store (S), Virtual Warehouse (W), Physical Warehouse (PW), All Stores (AS), or All Warehouses (AW).
Store Department Up-charge Details	Optional	Child node

Table 3-124 Store Department Up-charge Details Delete

Message Element	Required?	Notes
Component ID	Optional	This field contains the unique identifier of the up-charge component.
Item Default Indicator	Optional	Indicates if component rate information is deleted or not for existing items under the department.
Transfer Allocation Default Indicator	Optional	Indicates if component rate information is deleted or not for existing transfers and allocations under the department.

Flex Attributes

If you have defined any custom flex attributes (CFAs) for stores or store addresses, then they can be integrated as part of this API. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute.

Table 3-125 Custom Flex Attributes

Message Element	Required?	Notes
Name	Always	Indicates the name of the column defined in the group set view for flex attributes defined for the store.
Value	Always	Indicates the value of the attribute for the store if the attribute is a character or number.
Value Date	Always	Indicates the value of the attribute for the store if the attribute is a date.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
XStoreCre	External Store Create	XStoreDesc.xsd
XStoreLocTrtCre	External Store Location Trait Create	XStoreDesc.xsd
XStoreWTCre	External Walk-Through Store Create	XStoreDesc.xsd
XStoreDeptChrgCre	External Department Up-Charge Create	XStoreDesc.xsd
XStoreAddrCre	External Store Address Create	XStoreDesc.xsd
XStoreHrCre	External Store Hours Create	XStoreDesc.xsd
CustFlexAttriVO	External Store Flex Attribute Create	XStoreDesc.xsd
XStoreMod	External Store Modification	XStoreDesc.xsd
XStoreDeptChrgMod	External Department Up-Charge Modify	XStoreDesc.xsd
XStoreAddrMod	External Store Address Modify	XStoreDesc.xsd
XStoreHrMod	External Store Hours Modify	XStoreDesc.xsd
CustFlexAttriVO	External Store Flex Attribute Modify	XStoreDesc.xsd
XStoreDel	External Store Delete	XStoreRef.xsd
XStoreLocTrtDel	External Store Location Trait Delete	XStoreRef.xsd
XStoreWTDel	External Walk-Through Store Delete	XStoreRef.xsd
XStoreDeptChrgDel	External Department Up-Charge Delete	XStoreRef.xsd
XStoreAddrDel	External Store Address Create	XStoreRef.xsd
XStoreHrDel	External Store Hours Delete	XStoreRef.xsd

Transfer Subscription API

This section describes the transfer subscription API.

Functional Area

Transfers

Business Overview

This API subscribes to transfers from external systems to create, update or delete transfers in Merchandising. Within Oracle Retail solutions, this API is also leveraged by Advanced Inventory Planning (AIP) to create standalone transfers generated out of its replenishment processing. AIP does not use this API to update or delete previously created transfers.

Creating Transfers

When a new transfer is created, this API will first validate that all required fields are present in the message. Certain of the fields are required regardless of transfer type and system

configuration, while others are dependent on other Merchandising configurations. Additionally, when creating a new transfer at least one detail line must also be included in the message. After that, business level validation on the input information will be performed. The tables below summarize these two types of validation.

Table 3-126 Header Level Validation

Message Element	Required?	Notes
Transfer Number	Yes	Must be a unique transfer number not used by any existing transfers in Merchandising.
From Location Type	Yes	Must be either a store (S) or a warehouse (W).
From Location	Yes	See below.
To Location Type	Yes	Must be either a store (S), warehouse (W), or external finisher (E). For more on transfers with finishing, see below.
To Location	Yes	See below.
Delivery Date	Conditional	When AIP is part of your implementation, this is required for all transfer types, except RAC, EG, and SIM transfers. If included in the message, this must be today or a future date.
Department	Conditional	A system option determines whether or not the department is required for transfers. If the system option is set to require a department, then this must be included in the message. If the system option is set to not require the department, then the department must be null in this message unless the transfer type is SIM, AIP, or EG.
Routing Code	No	Valid values are 1, 2, or 3. The descriptions for these three options are held in the Codes table under code TRRC and can be configured as needed for your business.
Freight Code	No	If this is included in the message, it must have a value of normal (N), hold (H), or expedite (E). If no value is provided, it will default to normal.
Transfer Type	No	The following types of transfers can be created in this API: • Administrative (AD) • AIP Generated (AIP) • Book (BT) • Confirmation (CF) • Externally Generated (EG) • Intercompany (IC) • Manual Requisition (MR) • Reallocation (RAC) • Return to Vendor (RV) • SIM Generated (SIM) If the transfer type is not specified for the new transfer, then it will be defaulted to either Manual Requisition or Intercompany, depending on the legal entities of the locations on the transfer. See below for more details on transfer types.
Status	No	Transfers can be created in Input (I) or Approved (A) status in this API. See below for more on transfer status validation.
Create ID	No	If not passed into the message, then a value will be defaulted for auditing purposes.
Comments	No	Can support up to 2000 characters of text.
Context Type	No	Valid values for this field are found in the Codes table under code type CNTX.

Table 3-126 (Cont.) Header Level Validation

Message Element	Required?	Notes
Context Value	No	This may be used to provide additional information about the context of the transfer. For example, if the context type is promotion, this may indicate the promotion number or a description.
Custom Flex Attribute	No	Child node

Table 3-127 Detail Level Validation

Message Element	Required?	Notes
Item	Yes	An item must be a transaction-level, inventoried, and approved in order to be included on a transfer. If the transfer is from a store, then it cannot be a pack item, unless the transfer is of type AIP, SIM, or EG. Also, packs can also only be transferred from a warehouse if they have a receive as type of Pack, so that inventory exists at that level. If a department has been included in the transfer message at the header level, then all items must belong to that department.
Transfer Quantity	Yes	If the item has a standard unit of measure in the quantity class, then the quantity for the transfer must be an integer.
Supplier Pack Size	No	If included, this must be greater than zero. If not included, this will default to the primary supplier's pack size for the item at the from location, if an orderable item. If not orderable, this will default to 1.
Inventory Status	No	All items on a transfer must either be from available status or from unavailable status. A single transfer cannot mix available and unavailable statuses. Available inventory transfers should use a -1 or a null value in this field in the message. Unavailable inventory transfers should include a valid status from the Inventory Status Types configured in your environment.
Unit Cost	No	This is not used by Merchandising.
Adjustment Type	Conditional	This field, along with the adjustment value, is used to calculate the transfer price for intercompany transfers. It will be ignored for all other transfers. If the adjustment value is provided, then the type must also be specified. The valid values for this field are: • IA - Increase by Amount • IP - Increase by Percent • DA - Decrease by Amount • DP - Decrease by Percent • S - Set Price IA and IP can only be used if you have your system options set to allow the transfer price to exceed weighted average cost.
Adjustment Value	Conditional	If the adjustment type is provided, then the value must also be specified. This must always be a positive amount.

Location Validation

The from and to locations passed into the message must be valid stores or warehouses in Merchandising; but they cannot be the same. If both locations are stores, then they must both exist in the same transfer zone. Additionally, if the to location is a store, then it must be open. This is determined based on whether there is a close date defined for the store and the stop order days.

If either location is a warehouse, then it can be either a physical warehouse or a virtual warehouse, depending on transfer type. A physical warehouse is only allowed as the from location type for an EG type of transfer. Additionally, only Book type transfers are allowed between two warehouses in the same physical warehouse.

If either the from or to location is a franchise store, then the other location cannot be a finisher. If the franchise store is a non-stockholding location, then the other location on the transfer must be a warehouse.

Validation is also done at the item level based on the locations on the transfer. Each item on the transfer must be in active, inactive, or discontinued status at the from location. It also must have been ranged to the from location in Merchandising, when that location is a warehouse. However, if the from location is a store, there is an exception where the transfer can still be created even though it is not yet ranged, which also bypasses inventory validation. This is to support a specific function in Oracle Retail Store Inventory Management (SIM). See the section on SIM Generated transfers below for more details.

If the item is not already ranged to the to location, then ranging will occur when the transfer is created, regardless of status. The ranging that occurs will flag the item/location as unintentionally ranged for all transfer types except AIP.

If the to location is an external finisher, see the section below on transfers with finishing.

Inventory Validation

Another part of the validation that is applicable for all transfers created is that inventory is available for transfer if the status passed through the integration is approve (A), with a few exceptions. First, EG type transfers do not have inventory validated as it is assumed that this type of transfer is generated in the store or warehouse and the inventory availability check has been done in that solution as part of the shipping of the inventory. Additionally, if the system option titled Validate External Warehouse Availability is set to No (unchecked), then warehouse inventory will not be validated for any transfers initiated in this API regardless of type. Store inventory availability is never validated by this API because of support for the process where the item does not need to be ranged to the shipping store.

Status Validation

Transfers can be created in a status of Input (I) or Approved (A) using this API. Transfers in input status are not subject to inventory validation, but all other validations are applicable. Book type transfers can only be created in Input status using this API, as there isn't really a concept of an "approved" book transfer - as soon as it is approved it is executed. Additionally, transfers of type Reallocation (RAC) and Return to Vendor (RV) can also only be created in Input status. Conversely, transfers of type AIP, SIM, and EG must always be created in Approved status. If any validation fails when processing the new transfer that results in it not being able to be approved, the transfer will be created but will remain in input status. The exception to this is for transfers of type AIP, SIM, and EG, as they must always be created in approved status. If they are not able to be approved, the transfer is not created or updated.

Transfer Type Specific Validation

Most of the validation defined above is relevant regardless of transfer type, except where noted. However, there are also some other validations done as part of this API's processing that are specific to a type of transfer.

Administration (AD)

- See Manual Requisition

AIP Generated (AIP)

This type of transfer is expected only to be sent from AIP as an output of the replenishment process. As such, Merchandising assumes certain validations have been done by AIP in advance of receiving the transfer and slightly different validation is enforced. The following special validations apply for this transfer type using this API:

- Must be created in Approved status
- Can only be to stockholding locations
- Supports transferring packs from stores
- Allows the department number to be passed even when the system option is N
- Item/location ranging to the to location will result in the Ranged flag being set to Yes as it is assumed this an intentional ranging.
- Can be an intercompany transfer

Book (BT)

Book transfers processed through this API can be created for two virtual warehouses in the same physical warehouse only. This is usually used for inventory rebalancing between virtual locations. The following special validations apply for this transfer type using this API:

- Can only be created in Input status
- Can only be created for virtual warehouses in the same physical warehouse
- Warehouses must be in the same legal entity

Confirmation (CF)

- See Manual Requisition

Externally Generated (EG)

Externally Generated transfers are assumed to be created in the store or warehouse. Further, it is assumed that once they get to Merchandising, the transfer is already in process at that location. As such, there are certain validations that are managed differently for this transfer type in this API:

- Must be created in Approved status
- Supports transferring packs from stores
- Allows the department number to be passed even when the system option is N
- Can be an intercompany transfer

- Uses the physical warehouse number, not a virtual warehouse number, if warehouses are involved

Intercompany (IC)

An intercompany transfer is a type of business to business transaction that sells product from one legal entity and purchases it into another. Legal entities in Merchandising are determined based on the setting of the Intercompany Basis system option, which indicates whether the transfer entity or the set of books of a location should be used. This transfer type is used when either it is explicitly passed into the API or if the transfer type is NULL in the inbound message and the locations are in different legal entities. Other transfer types may also be intercompany, as well, but the below rules apply for those flagged as intercompany type explicitly:

- The legal entity of the from and to locations must be different.
- If an adjustment type or value is passed into the message, that will be used to calculate the "selling" price between entities. Otherwise, the from location's weighted average cost is used.

Manual Requisition (MR)

This is the most basic type of transfer in Merchandising, so it is used as a default transfer type when either it is explicitly passed into the API or if the transfer type is NULL in the inbound message and the locations are in the same legal entity. The behavior for this transfer type is the same as that for AD and CF types of transfers - those could be used as different reasons for a transfer. For this transfer type the following validation rules are enforced:

- Locations must be in the same legal entity

Reallocation (RAC)

A reallocation transfer is assumed to be used to pull back inventory from stores or warehouses to a single warehouse for re-allocation to other stores or other warehouses. This is the type of transfer that is created when a mass-return transfer is created, for example. Because it has unique rules tied to it related to MRTs, some additional validations are followed:

- Can only be created in Input status in this API
- Locations must be in the same legal entity

Return to Vendor (RV)

A return to vendor type of transfer is similar to a reallocation type, in that it is assumed to be pulling inventory back to a warehouse from stores or other warehouses, but in this case, for the purpose of returning the merchandise to the supplier. This is the type of transfer that is created when a mass-return transfer is created, for example. Because it has some unique rules tied to it related to MRTs, some additional validations are followed:

- Can only be created in Input status in this API
- Locations must be in the same legal entity

SIM Generated (SIM)

SIM generated transfers are created only by the store orders process in SIM. This functionality is not available in SIOCS. Because of this, they have special rules applied, including the ability to create the transfer even though no item/store relationship exists for the originating location in Merchandising. The rules that apply for this type of transfer include:

- Must be created in Approved status

- Supports transferring packs from stores
- Allows the department number to be passed even when the Merchandising system option is No
- Can be an intercompany transfer

All Transfer Types

For all of the above transfer types, if all validation described above passes, then the transfer will be created. If the transfer is created in Approved status, then in addition to the transfer itself, other details may also be created based on the items and locations involved.

- Inventory will be updated to reflect the reserved quantity at the from location and expected quantity at the to location.
- Upcharges will be applied, if configured, for transfers that do not include a physical warehouse location. For transfers with a physical warehouse, the records for upcharges are added when the transfer is shipped.
- An associated franchise order or return will be created if the transfer involves a franchise location.

Transfers with Finishing

Transfers with finishing are sometimes referred to as a two-legged transfer, as they generate two transfers in Merchandising. One from the originating store or warehouse to the finisher and one from the finisher back to a store or warehouse. This API supports the creation of a transfer with finishing only through an external finisher, a type of partner, and back to the originating location. Transfers to an internal finisher are not supported via this integration. To do this, when sending the transfer details in the message, you will indicate the external finisher as the "to" location. Then when the transfer is created, it will automatically generate the second leg.

When creating transfers in this way, it does not generate any work order activities to send to the finisher with the transfer - these will either need to be added manually in the Merchandising screens, or sent separately to the finisher.

Updating Transfers

For updates, the transfer number included in the message must already exist in Merchandising. Changes can be sent for header level updates or detail level updates. If the changes are at the header level, then the all the required header level information need to be included in the update, similar to that described above for creating a new transfer. However, the transfer details should not be included in a header level update. Fields that can be updated at the header level using this API include:

- Delivery Date - must always be a date today or later.
- Freight Code
- Status - to move from Input or Submitted to Approved only. Transfers cannot be moved back to Input status using this API.
- Comments
- Context Type
- Context Value

If the update is at the detail level - to add or update a line item - only the transfer number is required in the header record, the other details are ignored. If not included, then the message

will be rejected. Adding a new item to the transfer will use similar validation to that described above when creating the transfer.

If modifying an existing transfer line item, the full transfer quantity should be sent with the update, not the difference from the original quantity. This will be compared to the previous transfer quantity to determine how to update the transfer. For example, if the transfer is in approved or submitted status, a reduction in quantity would update the cancelled quantity on the transfer. It will also be validated to ensure that the quantity change doesn't result in the total transfer quantity being lower than what has already been shipped or what is expected to be picked based on updates to the selected or distro quantities on the transfer. For increases in transfer quantity, if the transfer is in submitted or approved status, then inventory will be validated based on the changed quantity (depending on system option settings) to validate that the additional units are available. The inventory status for the item cannot be modified.

Deleting Transfers

If you are deleting a line item on the transfer or deleting the whole transfer, then the API will validate that the transfer number is valid and that the transfer or transfer line was not already shipped or received, at least partially, or is not in process at the shipping warehouse or store. If you are deleting the whole transfer, then no details should be included in the message. If you are deleting a line on the transfer, then validation will be done to ensure that the item exists on the transfer.

Transfers are not actually deleted via this API, rather they are updated to a deleted status and a secondary process does the actual removal. Transfers can be deleted in any status, other than those already in a closed or deleted status, using this API. If the transfer involved an external finisher, then both legs on the transfer will be marked for delete. Deleting the last line on the transfer will also result in the transfer being flagged for delete.

If the transfer is in a status other than input, moving it to a deleted status or deleting a line will also update inventory to release the reserved inventory at the from location and decrease expected quantity at the to location. As well, if the transfer involves any franchise stores, then any franchise order or return created with the transfer will also be cancelled.

Table 3-128 Transfer Header Delete

Message Element	Required?	Notes
Transfer Number	Always	Must be an existing transfer in Merchandising.
Transfer Detail	Optional	Child node.

Table 3-129 Transfer Detail Delete

Message Element	Required?	Notes
Item	Always	Must be an item on an existing transfer in Merchandising.

Publishing Updates

Because these transfers that can be created, updated, or deleted using this API are managed in an external system, there are some cases where it is not published back out by Merchandising after it is processed to avoid the source system from receiving unneeded updates. This applies for transfers of type EG only. All other transfers will be published back to the RIB if approved or previously approved, such that the store and warehouse solutions responsible for executing the transfers are notified.

Flex Attributes

If you have defined any custom flex attributes (CFAS) for transfers, then they can be integrated as part of this API. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute. Flex attributes can only be added or updated to a transfer, they cannot be deleted. Additionally, for transfers with finishing, flex attributes can only be added to the first leg of the transfer.

Table 3-130 Flex Attributes

Message Element	Required?	Notes
Name	Yes	Holds the attribute name.
Value	No	Holds the value of the attribute for number and character type attributes
Value Date	No	Holds the date for date type attributes.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
Xtsfcre	Transfer Create	XTsfDesc.xsd
Xtsfdtlcre	Transfer Detail Create	XTsfDesc.xsd
CustFlexAttriVO	Transfer Flex Attribute Create	XTsfDesc.xsd
Xtsfmod	Transfer Modify	XTsfDesc.xsd
Xtsfdtlmod	Transfer Detail Modify	XTsfDesc.xsd
Xtsfdtlmod	Transfer Flex Attribute Modify	XTsfDesc.xsd
Xtsfdel	Transfer Delete	XTsfDesc.xsd
Xtsfdtldel	Transfer Detail Delete	XTsfRef.xsd

Vendor Subscription API

This section describes the vendor subscription API.

Functional Area

Foundation Data

Business Overview

Merchandising subscribes to vendor information that is published from an external financial application; however, this API is not used by Oracle Retail Financial Integration (RFI). Vendor can refer to either a partner or a supplier. Any partners loaded in this API must be created in Merchandising manually using the same ID as is sent in this API, to facilitate integration back to financials. Supplier information subscribed to by this API includes supplier addresses, org unit, and any flex attributes defined at the supplier or address levels.

Vendor Create and Update

When new suppliers or supplier sites are sent from an external system, they must contain the required header details, as well as address and org unit information. If the supplier information sent is for an ID that does not already exist, then this API will create a new supplier or site. If the ID already exists, then this API will update the existing supplier or site. For supplier sites, all addresses are expected to be the same, regardless of type. If a mandatory address type is not included in the message, it will be sent to the default value, based on the provided address. One or more org units can be associated with a supplier site using this API, if applicable.

Table 3-131 Vendor Create and Update

Message Element	Required?	Notes
Vendor Header	Always	Child node
Vendor Address	Optional	Child node
Vendor Org Unit	Optional	Child node

Table 3-132 Vendor Header Create and Update

Message Element	Required?	Notes
Supplier	Always	Unique identifying number for a supplier site within the system.
Supplier Name	Always	This field contains the supplier site name.
Supplier Name Secondary	Optional	This type can hold secondary name for the supplier site with a max length of 240 characters.
Contact Name	Always	This field contains the name of the supplier representative contact for this site.
Contact Phone	Always	This field contains a telephone number for the supplier's representative contact.
Contact Fax	Optional	This field contains a fax number for the supplier's representative contact.
Contact Pager	Optional	This field contains a pager number for the supplier's representative contact.
Supplier Status	Always	This field contains the status of the supplier site. Valid values are: - A - Active - I - Inactive
QC Ind	Always	This field determines whether orders from this supplier will require quality control.

Table 3-132 (Cont.) Vendor Header Create and Update

Message Element	Required?	Notes
QC Percentage	Optional	This field contains the percentage of items per receipt that will be marked for quality checking.
QC Frequency	Optional	This field contains the frequency for which items per receipt will be marked for quality checking.
VC Ind	Always	This field determines whether orders from this supplier will require vendor control.
VC Percentage	Optional	This field contains percentage of items per receipt that will be marked for vendor checking.
VC Frequency	Optional	This field contains the frequency for which items per receipt that will be marked for vendor checking.
Currency Code	Always	This field contains code identifying the currency the supplier site uses for business transactions.
Language	Optional	This field contains the supplier's preferred language.
Terms	Always	This field contains an indicator identifying the purchase terms that will default when an order is created for the supplier site. These terms specify when payment is due and if any discounts exist for early payment.
Freight Terms	Always	This field contains code indicating what freight terms will default when an order is created for the supplier site.
Return Allow Indicator	Always	This field indicates whether the supplier site will accept returns. Valid values are Yes (Y) or No (N).
Return Authorization Required	Always	This field indicates if returns must be accompanied by an authorization number when sent back to the vendor.
Returns Minimum Amount	Optional	This field contains a value if the supplier site requires a minimum merchandise value to be returned to accept the return. Returns of less than this amount will not be processed by the system. This field is stored in the supplier's currency.
Return Courier	Optional	This field contains the name of the courier that should be used for returns to the supplier site.
Handling Percentage	Optional	This field contains the default percent to be multiplied by the return's total cost to determine the handling cost for the return.
EDI PO Ind	Always	This field indicates whether purchase orders will be sent to the supplier via EDI.
EDI PO Change	Always	This field indicates whether purchase order changes will be sent to the supplier via EDI.
EDI PO Confirmation	Always	This field indicates whether acknowledgements of purchase orders will be sent from the supplier via EDI.
EDI ASN	Always	This field indicates whether the supplier will send Advance Shipment Notifications electronically.
EDI Sales Report Frequency	Optional	This field contains the EDI sales report frequency for the supplier. Valid values are weekly (W) or daily (D).

Table 3-132 (Cont.) Vendor Header Create and Update

Message Element	Required?	Notes
EDI Supplier Availability Indicator	Always	This field indicates whether the supplier will send availability via EDI.
EDI Contract Indicator	Always	This field indicates whether the supplier site supports contract ordering sent via EDI.
EDI Invoice Indicator	Always	This field indicates whether invoices, debit memos, and credit note requests will be sent to/from the supplier via EDI.
Cost Change Percentage Variance	Optional	This field contains a percent that determines whether a cost change can be auto approve via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
Cost Change Amount Variance	Optional	This field contains an amount (in supplier currency) that determines whether a cost change can be auto approve via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
Replenishment Approval Indicator	Always	This field indicates whether contract orders created via replenishment should be created in Approved status.
Ship Method	Optional	This field contains the default method used to ship the items on the purchase order from the supplier site. Valid values are held in code type SHPM.
Payment Method	Optional	This field indicates the default method for how purchase orders for this site will be paid. Valid options are: • LC - Letter of Credit • WT - Wire Transfer • OA - Open Account
Contact Telex	Optional	This field contains a telex number for the supplier's representative contact.
Contact Email	Optional	This field contains an email address for the supplier's representative contact.
Settlement Code	Always	This field indicates which payment process method is used for the supplier. Valid values are N/A (N) or Evaluated Receipts Settlement (E).
Pre-Mark Indicator	Always	This field indicates whether the supplier site supports pre-marking containers for cross dock orders.
Auto Approved Invoice Indicator	Always	This field indicates whether the supplier's invoices can be automatically approved for payment.
Debit Memo Code	Optional	This field indicates when a debit memo will be sent to the supplier site to resolve a discrepancy. Valid values are: • Y - if debit memos are always to be sent • L - if debit memos are used only if a credit note is not sent by the invoice due date • N - if debit memos are never sent
Freight Charge Indicator	Always	This field indicates whether a supplier site can charge freight costs.

Table 3-132 (Cont.) Vendor Header Create and Update

Message Element	Required?	Notes
Auto Approve Debit Memo Indicator	Always	This field indicates whether debit memos sent to the supplier site can be automatically approved on creation.
Inventory Management Level	Optional	This field indicates the level for managing supplier inventory information. Valid values are supplier (S), supplier/location (L), supplier/department (D), or supplier/department/location (A). If no value is provided, then if the department level orders system option is set to Yes, then this is defaulted to supplier/department, otherwise it is defaulted to supplier.
Backorder Indicator	Always	This field indicates if backorders or partial shipments will be accepted.
VAT Region	Optional	This field contains the unique identifying number for the VAT region applicable for this site.
Prepay Invoice Indicator	Always	This field indicates whether all invoices for the supplier can be pre-paid.
Service Performed Required Indicator	Always	This field indicates if the supplier's services must be confirmed as performed before paying an invoice from that supplier site.
Invoice Payment Location	Optional	This field indicates where invoices from this supplier site are paid - at the store (S) or centrally through corporate accounting (C).
Invoice Received Location	Optional	This field indicates where invoices from this supplier site are received - at the store (S) or centrally through corporate accounting (C).
Invoice At	Optional	This field indicates if the supplier site invoice lists items at gross cost (G) or net cost (N).
Delivery Policy	Always	This field contains the default delivery policy of the supplier site. Valid values come from the DLVY code type.
Comments	Optional	This field contains any miscellaneous comments associated with the supplier.
Default Item Lead Time	Optional	This field holds the default lead time for the supplier site. The lead time is the time the supplier needs between receiving an order and having the order ready to ship. This value will be defaulted to item/supplier relationships.
DUNS Number	Optional	The Dun and Bradstreet number of the supplier.
DUNS Location	Optional	The Dun and Bradstreet number of the location of the supplier.
Bracket Costing Indicator	Always	This field will determine if the supplier site supports bracket costing pricing structures.
VMI Order Status	Optional	This field determines the status in which any inbound POs from this supplier will be created. A NULL value indicates that the supplier is not a VMI supplier.
End Active Date	Optional	Not used by Merchandising.

Table 3-132 (Cont.) Vendor Header Create and Update

Message Element	Required?	Notes
DSD Supplier Indicator	Always	This field specifies whether the vendor supports DSD ordering, where the supplier replenishes the store directly, creating the PO and receipt at the same time.
Supplier Quantity Level	Always	This field indicates the supplier site order quantity level. Valid values are cases (CA) or eaches (EA).
Supplier Parent	Optional	This is the supplier number for the supplier sites.
Store Delivery Discrepancy	Optional	Not used by Merchandising
Final Destination Indicator	Always	This field indicates whether the supplier site can ship to final destination or not. Valid values are Yes (Y) or No (N).
Custom Flex Attributes	Optional	Child Node
External Reference Indicator	Optional	This column holds the ID for the supplier used in the external financial system.
Deal Upload Status	No	This field indicates the status in which the deal will be created when uploaded. The valid values are: - W - Worksheet - S - Submitted - A - Approved
Tax ID	No	This field contains the unique tax identification number of the supplier site.
Default Pricing Control	No	Indicates the default value of the pricing control attribute at the supplier site level when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: - R - Retailer - S - Supplier
Generate Consignment Order	No	This field will allow specification of whether Merchandising is required to generate POs against consignment sales. Valid values are: 'Y' - MFCS will create PO, 'N' - POs will get created externally.
Generate Self Invoicing	No	This field specifies whether self-invoicing needs to be done for consignment purchases or if the vendor will send invoices to be matched. Valid values are: Y - Merchandising creates the invoice, N - Invoice will be created by the vendor.

Table 3-133 Vendor Address Create or Update

Message Element	Required?	Notes
Module	Always	This field indicates the data type that the address is attached to. In this case, it will always be 'SUPP'.
Key Value 1	Always	This field holds the ID the address is attached to. In this case, it will be the supplier number.
Key Value 2	Optional	This is not used.

Table 3-133 (Cont.) Vendor Address Create or Update

Message Element	Required?	Notes
Sequence Number	Always	Number indicating the sequence that addresses within the same type were entered.
Address Type	Always	This field contains the address type. Valid address types are: • Business (01) • Postal (02) • Returns (03) • Order (04) • Invoice (05) • Remittance (06) If any address types have been flagged as mandatory and are not included when the supplier is being created, then if an ordering address has been included, the missing mandatory addresses will be defaulted to that address. If not, then the remittance address will be used. If neither an order nor remittance address is included, then the first address sent is used.
Primary Address Indicator	Always	This field indicates whether the address is the primary address for the address type.
Address 1	Always	This field contains the first line of the address.
Address 2	Optional	This field contains the second line of the address.
Address 3	Optional	This field contains the third line of the address.
City	Always	This field contains the name of the city that is associated with the address.
State	Optional	This field contains the state abbreviation for the address.
Country ID	Always	This field contains the country where the address exists.
Jurisdiction Code	Optional	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
Post	Optional	This field contains the zip code for the address.
Contact Name	Optional	This field contains the name of the contact for the supplier at this address.
Contact Phone	Optional	This field contains the phone number of the contact person at this address.
Contact Telex	Optional	This field contains the telex number of the contact person at this address.
Contact Fax	Optional	This field contains the fax number of the contact person at this address.
Contact Email	Optional	This field contains the email address of the supplier site's contact person.
Custom Flex Attributes	Optional	Child Node

Table 3-134 Vendor Org Unit Create

Message Element	Required?	Notes
Org Unit ID	Always	This field contains org unit ID added or updated for the supplier site.
Primary Pay Site Indicator	Always	This field contains the primary pay site indicator.

Flex Attributes

If custom flex attributes (CFAS) have been defined for suppliers or addresses, then they can be integrated as part of this API. The node of the integration that supports this will accept the name of the attribute as it is defined in the group-set-level view and the value for the attribute. Flex attributes can only be added to or updated for a supplier and supplier address but cannot be deleted.

Message Element	Required?	Notes
Name	Always	Holds the name of the attribute.
Value	Optional	Holds the value of the attribute for non-date attributes.
Value Date	Optional	Holds the value of the attribute for date attributes.

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Below are the filenames that correspond with each message type. Please consult the Oracle Retail Integration Guide for each message type for the details on the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
vendorcre	Vendor Create	VendorDesc.xsd

Work Order Status Subscription API

This section describes the work order status subscription API.

Functional Area

Transfers

Business Overview

For transfers with finishing, Merchandising subscribes to work order status messages sent from internal finishers indicating that the work order activities are complete. This message is used for internal finishers located in the same physical warehouse as the final destination for the transfer, as there is no physical shipment of goods. Other finishing scenarios exist in which the finisher is not a virtual warehouse that shares a physical warehouse with the transfer's final receiving location. In these instances, Work Order Status messages are not necessary, and Merchandising will disregard Work Order Status messages sent in these scenarios.

Work order status messages contain the items for which the activities have been completed, along with the quantity that was completed. All items on transfers that pass through an internal finisher must have at least one work order activity associated with them. When work order status messages are received for a particular item/quantity, it is assumed that all activities on the work order associated with the item/quantity have been completed. If work order activities involve item transformation or repacking, the work order status messages are always created in terms of the resultant item or pack.

On processing the work order status update, a book transfer is executed between the internal finisher (which is held as a virtual warehouse) and the final receiving location (also a virtual warehouse). If the internal finisher belongs to the sending location's transfer entity, intercompany out and intercompany in transactions are recorded. Quantities on hand, reserved quantities, and weighted average costs are adjusted to accurately reflect the status of the stock.

It is possible to receive multiple Work Order Status messages for a particular item/transfer. Work order completion of partial quantities addresses the following scenarios:

1. Work order activities could not be performed for the entire quantity of a particular item at one time.
2. A given quantity of the particular item was damaged while work order activities were performed.

Work Order Example

Assume that a quantity of 20 of item 100 (White XL T-shirt) are sent to an internal finisher at the receiving physical warehouse, where they will be dyed black, thereby transforming them into item 101 (Black XL T-shirt). If all finishing activities were successfully completed in this example, Merchandising could expect to receive a Work Order Status message containing item 101 with a quantity of 20.

Work Order Status Creation

While consuming the Work Order Status message, Merchandising validates that the finisher and the transfer's final receiving location are in the same physical warehouse. If not, processing is halted. If the message contains an item, work order complete processing will be called for that item. Otherwise, said processing will be called for all items on the transfer. If the entire transfer is processed, the child transfer (that is, the second leg) will be set to Shipped status. Note that work orders are always associated with the second leg of multi-leg transfers. Whether processing is performed at the item or transfer level, transfer closing logic will be used to determine if the entire multi-leg transfer can be closed.

Table 3-135 Work Order Status

Message Element	Required?	Notes
Work Order ID	Yes	This field contains the work order number under which the finishing activities were performed.
Distro Number	Yes	This field contains the transfer number containing the work order. This is the 2nd leg transfer number.
Distro Doc Type	Yes	This should always be transfer (T).
Distro Parent Number	No	This is the first leg transfer number. If this field must be the same as the tsf_parent_no of the distro number in the tsfhead table in Merchandising.
Distro Parent Type	No	Not used
Item	Conditional	This contains the item on which the work order has been completed. If an item transformation occurred, this should be the resultant item. If the completed quantity field has a value, then this field is required.
Warehouse	Yes	Not used, but required as input
Location Type	Yes	This field contains the destination location type where the finished goods are sent. Not used, but required as input
Location	Yes	This field contains the destination location where the finished goods are sent. This is the final location of the 2-legged transfer.
Sequence Number	Yes	Not used, but required as input
Work In Process Code	Yes	Not used, but required as input
Instructions	Yes	Not used, but required as input
Complete Date	Yes	This field contains the date when the work order was completed.
Completed Quantity	Conditional	This contains the number of items that resulted from the work order activity. If the item field has a value, then this field is required.
Completed Indicator	Yes	Not used
Work Order Status	No	Child node
Inventory Adjustment		Not used

Table 3-136 Work Order Status Inventory Adjustment

Message Element	Required?	Notes
From Disposition	No	Not used
To Disposition	No	Not used
Unit Quantity	No	Not used

Error Handling

If any errors are encountered in the validations described above or any of the message structure validations, a status of E is returned to the external system along with the appropriate

error message. If the message has been successfully persisted, a success status (S), is returned to the external system indicating that the message has been successfully received and persisted to the Merchandising database.

Message XSD

Here are the filenames that correspond with each message type. Please consult RIB documentation for each message type to get a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
wostatuscre	Work Order Status Create Message	WOStatusDesc.xsd

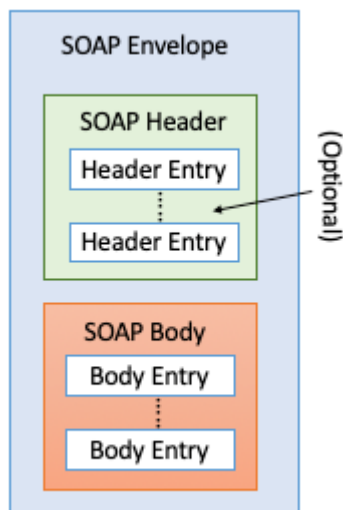
4

SOAP Web Services

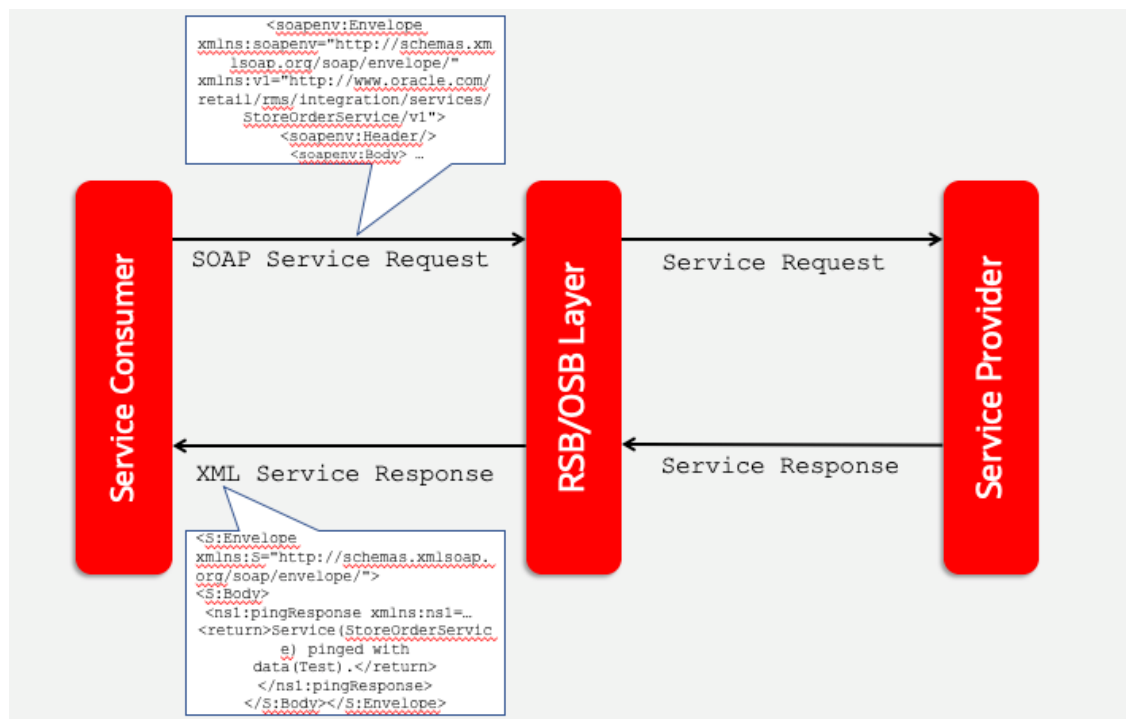
This chapter gives an overview about the SOAP Web service implementation used in Merchandising.

The Simple Object Access Protocol (SOAP) is a general-purpose messaging protocol that is the de facto standard for web services messaging and interaction through the Oracle Retail Integration Cloud Service (RICS) Retail Service Backbone (RSB), which provides monitoring for the SOAP services used by Merchandising. The basic unit of interaction between a SOAP client and a SOAP-enabled service is a message. A SOAP message is basically an XML document that consists of two parts:

1. An optional header providing information on authentication, encoding of data, or how a recipient of a SOAP message should process the message.
2. The body that contains the message. These messages are defined using the WSDL specification.



An envelope can enclose any number of optional headers. The following diagram shows the high-level architecture of SOAP web service implementation with respect to Merchandising:



Using SOAP Services During Batch Window

The services should not be used during the restricted batch window.

Common Characteristics of Merchandising SOAP Services

A Retail Application will package its SOAP services as part of the application's Enterprise Archive (EAR) file. Installation of the SOAP web services is therefore done by default as part of the application install. Refer to *Oracle Retail Service Backbone Implementation Guide* for more details.

Security

Services are secured using a standard policy-based security model supported by WebLogic and OSB.

For more details, refer to the *Oracle Retail Service Backbone Security Guide*.

Standard Success Response

Example response payload in case of service success is depicted below:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns1:createDetailXAllocDescResponse xmlns:ns1="http://www.oracle.com/retail/rms/integration/services/AllocationService/v1"
    xmlns:ns3="http://www.oracle.com/retail/integration/base/bo/InvocationSuccess/v1"
    xmlns:ns2="http://www.oracle.com/retail/integration/base/bo/XAllocDesc/v1"
    xmlns:ns4="http://www.oracle.com/retail/integration/base/bo/XAllocColRef/v1"
    xmlns:ns5="http://www.oracle.com/retail/integration/base/bo/XAllocRef/v1">
```

```

        <ns3:InvocationSuccess>
          <ns3:success_message>createDetailXAllocDesc service call was successful.</
ns3:success_message>
        </ns3:InvocationSuccess>
      </ns1:createDetailXAllocDescResponse>
    </S:Body>
  </S:Envelope>

```

Standard Error Response

Example response payload in case of service error is depicted below:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns0:Fault xmlns:ns0="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://
www.w3.org/2003/05/soap-envelope">
      <faultcode>ns0:Server</faultcode>
      <faultstring>Invalid Item. 1003500087</faultstring>
      <detail>
        <ns0:IllegalStateWSFaultException xmlns:ns0="http://www.oracle.com/retail/
integration/services/exception/v1">
          <ns0:shortErrorMessage>Invalid Item. 1003500087</ns0:shortErrorMessage>
          <ns0:errorDescription>
            com.oracle.retail.integration.services.exception...
          </ns0:errorDescription>
          <ns0:BusinessProblemDetail>
            <ns0:problemDescription>Invalid Item. 1003500087</
ns0:problemDescription>
          </ns0:BusinessProblemDetail>
        </ns0:IllegalStateWSFaultException>
      </detail>
    </ns0:Fault>
  </S:Body>
</S:Envelope>

```

URL Path

Based on your implementation you will need to prefix the end point with just the deployment hostname. The format that should be used for the hostname is `rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace>/`, where `<Region Name>` will be the region for your data center and `<Customer Subnamespace>` will be specific to your company's name and environment (production, stage, and so on).

To access the Merchandising SOAP web services WSDL file:

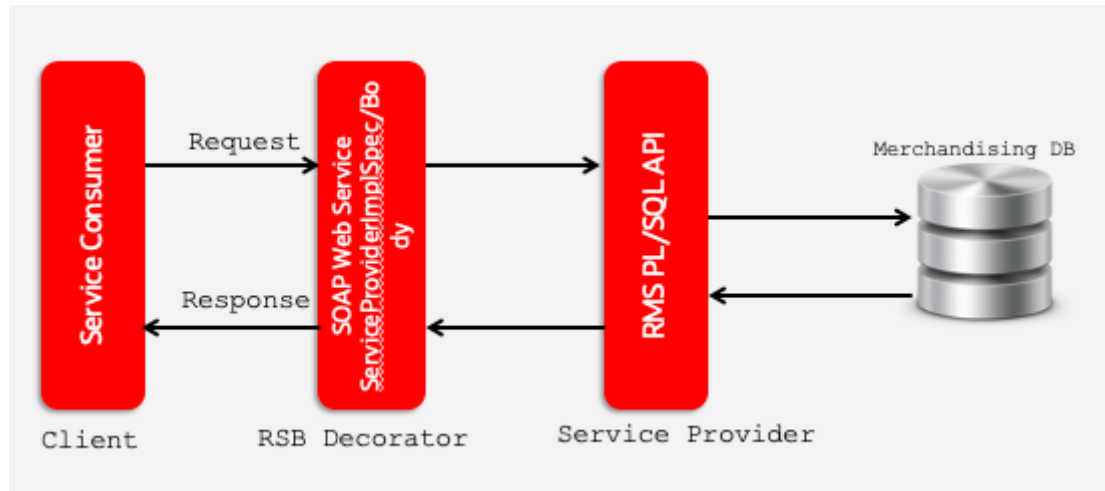
```
https://<hostname>/<end-point>
```

Based on your implementation you will need to prefix the end point with just the deployment hostname. The format to use for the hostname is `rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace>/`, where `<Region Name>` is the region for your data center and `<Customer Subnamespace>` is specific to your company's name and environment (production, stage, and so on).

The end point information is in the descriptions of each of the provider services later in this chapter.

Web Service APIs Process Flow

The following diagram shows the Web Service API process flow for Merchandising as a Service Provider:



Provider Services

This section gives an overview about the SOAP Web service provider implementation API designs used in the Merchandising environment and various functional attributes used in the APIs.

Note

The following service provider implementation API designs are intended only to give a high-level overview of the APIs available. The implementation of these services, along with the associated Web Service Definition Language (WSDL), may be used to get a full understanding of the data requirements, validation rules, persistence rules, and return values associated with the service.

To provide visibility to the background processing that's occurring, services write to the JOB_AUDIT_LOGS table in the database. Reports can be built based on this to provide visibility to what is happening in the background. Additionally, to assist users and developers in troubleshooting any error that may arise, the payload that was processed is also stored in the JOB_AUDIT_PAYLOAD table in the database.

Once the nightly batch run has started, web service execution will be halted, and users will receive a warning message that the nightly batch run has commenced.

Consumer Services

This section lists the details on the SOAP services where Merchandising is the consumer of the service.

Provider Services

This section gives an overview about the SOAP Web service provider implementation API designs used in the Merchandising environment and various functional attributes used in the APIs.

 Note

The following service provider implementation API designs are intended only to give a high level overview of the APIs available.

The implementation of these services, along with the associated Web Service Definition Language (WSDL), may be used to get a full understanding of the data requirements, validation rules, persistence rules, and return values associated with the service.

Allocation Service

Functional Area

Allocation

RSB Proxy WSDL

/rms-Allocation-AppServiceDecorator/ProxyService/AllocationAppServiceProxy?wsdl

Merchandising Service WSDL

/AllocationBean/AllocationService?WSDL

Overview

This service allows an external application to create, update, and delete allocations within Merchandising based on warehouse inventory or to cross-dock a purchase order.

This service uses the same logic as is supported in the Allocation Subscription RIB API. For information about this functionality, see [Allocation Subscription API](#) in the "RIB Subscription Designs" chapter of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	XAllocDesc	InvocationSuccess	XAllocDesc.xsd
createDetail	XAllocDesc	InvocationSuccess	XAllocDesc.xsd
modifyHeader	XAllocDesc	InvocationSuccess	XAllocDesc.xsd

Operation Name	Input Object Type	Output Object Type	XML Definition
modifyDetail	XAllocDesc	InvocationSuccess	XAllocDesc.xsd
delete	XAllocColRef	InvocationSuccess	XAllocRef.xsd XAllocColRef.xsd
deleteDetail	XAllocColRef	InvocationSuccess	XAllocRef.xsd XAllocColRef.xsd

Average Cost Service

Functional Area

Finance

RSB Proxy WSDL

/rms-AverageCost-AppServiceDecorator/ProxyService/AverageCostAppServiceProxy?wsdl

Merchandising Service WSDL

/AverageCostBean/AverageCostService?WSDL

Overview

This service supports updating weighted average cost from an external system for one or more item/locations combinations. It also creates a tran data record posting with tran code 70 for the difference in cost, based on owned inventory at the location at the time the cost change is applied.

The web service will be called with the following details:

- Item
- Location
- Location Type
- New average cost (must be greater than 0)

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
modifyAvgCost	ItLocAgCstColDesc	InvocationSuccess	ItLocAgCstDesc.xsd ItLocAgCstColDesc.xsd

Cost Change Service

Functional Area

Cost Change

RSB Proxy WSDL

/rms-CostChange-AppServiceDecorator/ProxyService/CostChangeAppServiceProxy?wsdl

Merchandising Service WSDL

/CostChangeBean/CostChangeService?WSDL

Overview

This service is exposed to allow an external application to create cost changes in Merchandising. It takes a collection of cost changes and will return success and failure through the service response object.

This service uses the same logic as is supported in the Cost Change Subscription RIB API. For information about this functionality, see [Cost Change Subscription](#) in the "[RIB Subscription Designs](#)" section of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	XCostChgColDesc	InvocationSuccess	XCostChgDesc.xsd XCostChgColDesc.xsd

Customer Credit Check Service

Functional Area

Franchise

RSB Proxy WSDL

/rms-CustomerCreditCheck-AppServiceDecorator/ProxyService/
CustomerCreditCheckAppServiceProxy?wsdl

Merchandising Service WSDL

/CustomerCreditCheckBean/CustomerCreditCheckService?WSDL

Overview

This API provides a way for an external source, usually a financials system, to update the credit status for a franchise customer in Merchandising. This status is used when determining

whether a franchisee order can be approved. Valid values are Y (credit is good) and N (credit issues). For each collection of customer and customer group passed into the API, the credit flag will be updated with the value indicated in the service call.

Merchandising returns failure status as part of the response object in the web service call if credit flag is not updated due to validation errors.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
updateCustCredit	CustCreditChkCol	InvocationSuccess	CustCreditChkDesc.xsd CustCreditChkCol.xsd

Customer Order Fulfillment Service

Functional Area

Customer Order Fulfillment

RSB Proxy WSDL

/rms-FulfillOrder-AppServiceDecorator/ProxyService/FulfillOrderAppServiceProxy?wsdl

Merchandising Service WSDL

/FulfillOrderBean/FulfillOrderService?WSDL

Overview

This service is used to process Customer Order Fulfillment requests from an order management system (OMS). Merchandising supports two integration methods for processing Customer Order Fulfillment messages from OMS - either through RIB or web service. At implementation time, you should decide on either one or the other integration method, but not both. The same core logic is used to validate and persist customer orders.

In a web service implementation, the web service is used to create or cancel a customer order in Merchandising. This service

- Accepts a collection of fulfillment orders as input. If one order fails, the entire service call fails and no orders will be created.
- Returns Failure status as part of the response object in the web service call if customer orders are not created due to validation errors.
- Returns Success status and a confirmation message as part of the response object of type
 - X if customer orders are not created due to lack of inventory
 - P if customer orders are partially created due to insufficient inventory
 - C if customer orders are completely created, when sufficient inventory is available

In a web service implementation, confirmation messages will be sent in a collection as part of the response object.

This is the web service version of the same logic as is supported in the RIB version of the API. See [Customer Order Fulfillment Subscription API](#) in the "[RIB Subscription Designs](#)" chapter for more information.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	FulfilOrdColDesc	FulfilOrdCfmCol	FulfilOrdDesc.xsd
			FulfilOrdColDesc.xsd
cancel	FulfilOrdColRef	InvocationSuccess	FulfilOrdRef.xsd
			FulfilOrdColRef.xsd

Customer Order Item Substitution Service

Functional Area

Customer Orders

RSB Proxy WSDL

/rms-CustOrdSubstitute-AppServiceDecorator/ProxyService/CustOrdSubstituteAppServiceProxy?
wsdl

Merchandising Service WSDL

/CustOrdSubstituteBean/CustOrdSubstituteService?WSDL

Business Overview

When a store is picking inventory to fulfill a customer order, if the inventory of the item ordered does not meet quality standards or is unavailable, and the order indicates that substitutions are allowed for that item, the store may choose to fulfill the order with a substitute item. If that occurs, SIM has the ability to substitute items on the customer order with another predefined substitute item. In such cases, SIM notifies OMS via the Stock Order Status message that an alternative item has been pushed into the order.

Based on the notification from SIM, OMS updates the customer order and notifies Merchandising with the same details received from SIM using this API. Merchandising will then update the inventory and customer order details - removing the reservation for the original item and adding a reservation for the new item. Merchandising will also update the cancelled quantity for the original item on the order and add the details for the substituted item, with a cross reference to the original item.

Assumptions

- Substitution logic holds good only for the customer orders fulfilled from stores.
- Catchweight, Transformable, Consignment, Concession and Deposit container items are not supported for customer order item substitution.
- The quantities are always in Standard UOM.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	CustOrdSubColDesc	InvocationSuccess	CustOrdSubDesc.xsd CustOrdSubColDesc.xsd

Diff Management Service

Functional Area

Foundation

RSB Proxy WSDL

/rms-DiffManagement-AppServiceDecorator/ProxyService/DiffManagementAppServiceProxy?wsdl

Merchandising Service WSDL

/DiffManagementBean/DiffManagementService?WSDL

Overview

This service supports the following functions

- Creating new differentiator (diff) IDs
- Updating existing diff IDs
- Deleting existing diff IDs
- Creating diff group header and details
- Updating existing diff group headers and details
- Deleting existing diff group headers and details

This API uses the same logic that is used for managing diffs through the Diff Subscription RIB API. See [Diff Group Subscription API](#) and [Differentiator Subscription API](#) in the "[RIB Subscription Designs](#)" chapter of this document for more details.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
createDiffId	XDiffIDColDesc	InvocationSuccess	XDiffIDDesc.xsd XDiffIDColDesc.xsd
modifyDiffId	XDiffIDColDesc	InvocationSuccess	XDiffIDDesc.xsd XDiffIDColDesc.xsd
deleteDiffId	XDiffIDColRef	InvocationSuccess	XDiffIDRef.xsd XDiffIDColRef.xsd
createDiffGrp	XDiffGrpColDesc	InvocationSuccess	XDiffGrpDesc.xsd XDiffGrpColDesc.xsd
createDiffGrpDtl			
modifyDiffGrp	XDiffGrpColDesc	InvocationSuccess	XDiffGrpDesc.xsd XDiffGrpColDesc.xsd
modifyDiffGrpDtl			
deleteDiffGrp	XDiffGrpColRef	InvocationSuccess	XDiffGrpRef.xsd XDiffGrpColRef.xsd
deleteDiffGrpDtl			

Inventory Back Order Service

Functional Area

Inventory

RSB Proxy WSDL

/rms-InventoryBackOrder-AppServiceDecorator/ProxyService/
InventoryBackOrderAppServiceProxy?wsdl

Merchandising Service WSDL

/InventoryBackOrderBean/InventoryBackOrderService?WSDL

Overview

Retailers selling through ecommerce channels often take customer orders even if inventory is not available with the expectation of future inventory being available to fill the order. If an order is captured against future inventory by the Order Management System (OMS), then a backorder message is sent to Merchandising through this service.

This web service will update the backorder quantity in Merchandising - increasing when the backorder is taken and decreasing when the backorder is released for fulfillment or cancellation.

Assumptions

- Backorders can be taken against both stores and warehouses. OMS will determine which location will be back ordered.
- An item does not need to have an open purchase order in order to increase backorder quantity.
- Catchweight, Transformable, Consignment, Concession and Deposit container items are not supported for backorder requests.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	InvBackOrdColDesc	InvocationSuccess	InvBackOrdDesc.xsd InvBackOrdColDesc.xsd

Inventory Lookup Service

Functional Area

Inventory

RSB Proxy WSDL

/rms-InventoryDetail-AppServiceDecorator/ProxyService/InventoryDetailAppServiceProxy?wsdl

Merchandising Service WSDL

/InventoryDetailBean/InventoryDetailService?WSDL

Overview

This real-time inventory availability lookup facility can be used by external systems, such as an on-line order capture system (OOC) or order management system (OMS), to retrieve item/location available inventory based on Merchandising's current view of inventory. Merchandising will provide this information for any warehouse or store which is valid for customer order sourcing/fulfillment.

Available inventory is calculated as

Stock on Hand - (transfer reserved + customer reserved + RTV + non-sellable)

Any failures (validation errors) encountered during the processing are passed back into the response object. If there are no failures, success status is returned.

Assumptions

- Catchweight, transformable, consignment, concession and deposit container items are not supported in this API.
- This inventory detail lookup is only for customer orderable inventory - sellable items at customer order locations. If a physical warehouse and channel are passed into the API, then only the inventory for the customer orderable virtual warehouses for that physical warehouse/channel are returned.
- If the inventory lookup is for a pack item at store, the pack inventory is estimated based on the maximum number of complete packs which can be created by using the available inventory of its components.
- Merchandising does not use the Search Area information in this service. It will only lookup inventory for the specific locations included in the input object.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
lookup	InvAvailCriVo	InvAvailColDesc	InvAvailCriVo.xsd InvAvailDesc.xsd InvAvailColDesc.xsd

Item Management Service

Functional Area

Item

RSB Proxy WSDL

/rms-ItemManagement-AppServiceDecorator/ProxyService/ItemManagementAppServiceProxy?wsdl

Merchandising Service WSDL

/ItemManagementBean/ItemManagementService?WSDL

Overview

The Item Management service allows an external systems to request pre-issued item numbers, as well as create, modify and delete various aspects of an item.

Request Item Numbers

The Item Number Reservation web service allows external systems such as Oracle Retail Assortment Planning (AP) to reserve item numbers in Merchandising. This web service contains the following details:

Column Name	Notes
Item Number Type	Required. Indicates the type of items numbers being requested. Valid options are: • ITEM - which is type Oracle Retail Item Number • UPC-A - which is type UCC12 • UPC-AS - which is type UCC12 with Supplement • EAN13 - which is type EAN/UCC-13
Quantity	Indicates the number of item numbers being requested. Required.
Days Until Expiry	Indicates how long the calling solution wants Merchandising to retain the reservation. After this many days, the reservation will be released allowing these numbers to be used for other purposes. This is required and must be a value greater than 0.

The requested item numbers are sent back to the calling solution as a response. This operation is only available as part of the web service.

Create/Manage Items

The operations supported in this service for creating and managing items are as follows:

- Creating and modifying items
- Creating, modifying, and removing item suppliers
- Creating, modifying, and removing item supplier sourcing country
- Creating, modifying, and removing item supplier country dimensions
- Creating, modifying, and removing item UDA combinations
- Creating and removing item reclassifications

This service uses the same logic to manage these operations as is used in the Item Subscription RIB API. For information on this functionality, see [Item Subscription API](#) in the "[RIB Subscription Designs](#)" chapter of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
reserveItemNumber	ItemNumCriVo	ItemNumColDesc	ItemNumCriVo.xsd ItemNumDesc.xsd ItemNumColDesc.xsd
createItem	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
createSupplier	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
createSupplierCountry	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
createSupplierCountry Dim	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd

Operation Name	Input Object Type	Output Object Type	XML Definition
createUDA	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
modifyItem	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
modifySupplier	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
modifySupplierCountry	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
modifySupplierCountryDim	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
modifyUDA	XItemColDesc	InvocationSuccess	XItemDesc.xsd XItemColDesc.xsd
deleteSupplier	XItemColRef	InvocationSuccess	XItemRef.xsd XItemColRef.xsd
deleteSupplierCountry	XItemColRef	InvocationSuccess	XItemRef.xsd XItemColRef.xsd
deleteSupplierCountryDim	XItemColRef	InvocationSuccess	XItemRef.xsd XItemColRef.xsd
deleteUDA	XItemColRef	InvocationSuccess	XItemRef.xsd XItemColRef.xsd
createItemReclass	XItemRclsDesc	InvocationSuccess	XItemRclsDesc.xsd
createItemReclassDetail	XItemRclsDesc	InvocationSuccess	XItemRclsDesc.xsd
deleteItemReclass	XItemRclsRef	InvocationSuccess	XItemRclsRef.xsd
deleteItemReclassDetail	XItemRclsRef	InvocationSuccess	XItemRclsRef.xsd

Pay Term Service

Functional Area

Financial Integration

RSB Proxy WSDL

/rms-PayTerm-AppServiceDecorator/ProxyService/PayTermAppServiceProxy?wsdl

Merchandising Service WSDL

/PayTermBean/PayTermService?WSDL

Overview

The Pay Term Service is used by Oracle Retail Financial Integration (RFI) for integration of payment terms with PeopleSoft Financials, and can also be used by an external financial systems to send new and updated payment terms information to Merchandising. The operations supported in this service are:

- Create: Create payment terms and details
- Create Detail: Add details to an existing payment term
- Update Header: Modify existing payment term header information
- "Update or Update Details: Modify existing details for a payment term

The operations supported by this service involve an external system sending Merchandising details to create or update payment terms. In the response back, the terms keys are returned on success. The create and update options for this service use the same logic as is supported in the Payment Terms Subscription RIB API.

For information about this functionality, see [Payment Terms Subscription API](#) in the "[RIB Subscription Designs](#)" chapter of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	PayTermDesc	PayTermRef	PayTermDesc.xsdPayTermRef.xsd
createDetail	PayTermDesc	PayTermRef	PayTermDesc.xsdPayTermRef.xsd
update	PayTermDesc	PayTermDesc	PayTermDesc.xsd
updateHeader	PayTermDesc	PayTermDesc	PayTermDesc.xsd
updateDetail	PayTermDesc	PayTermDesc	PayTermDesc.xsd

Pricing Cost Service

Functional Area

Foundation Data

RSB Proxy WSDL

/rms-PricingCost-AppServiceDecorator/ProxyService/PricingCostAppServiceProxy?wsdl

Merchandising Service WSDL

/PricingCostBean/PricingCostService?WSDL

Overview

This web service is used to expose pricing cost information to external systems. The primary user of this information is assumed to be an Order Management System (OMS), which manages franchise customer orders and needs visibility to cost information as part of the negotiation process for margin visibility.

Pricing cost for an item at an owned location is the unit cost for the primary supplier/country, less off invoice deals, plus estimated landed costs. Pricing cost for an item at a customer (franchise) location is the unit cost for the costing location, less any deals passed through, plus estimated landed costs (based on system option), plus the franchise cost template details. This API supports providing cost information for an item/location or item/supplier/location.

Any failures (validation errors) encountered during the processing are passed back into the response object. If there are no failures, success status is returned.

Assumptions

- Only Approved and transaction level items are valid.
- Location must be company store or physical warehouse that is customer orderable. For a physical warehouse, it must also include the channel ID that should be used.
- For physical warehouses, the cost returned will be for the virtual warehouse that matches channel ID included in the inputs. If there is not a virtual warehouse that matches that channel in the physical warehouse, then next best match will be determined based on channel type and the primary warehouse and protected flags on the virtual warehouses in the physical warehouse.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
lookup	PrcCostColCriVo	PrcCostColDesc	PrcCostCriVo.xsd PrcCostColCriVo.xsd PrcCostDesc.xsd PrcCostColDesc.xsd

Purchase Order Management Service

Functional Area

Purchase Order

RSB Proxy WSDL

```
/rms-PurchaseOrderManagement-AppServiceDecorator/ProxyService/  
PurchaseOrderManagementAppServiceProxy?wsdl
```

Merchandising Service WSDL

/PurchaseOrderManagementBean/PurchaseOrderManagementService?WSDL

Overview

The Purchase Order Management service allows an external systems to request pre-issued order numbers, create a purchase order, modify a purchase order, or delete purchase order details.

Request Order Numbers

The order reservation operation allows external systems such as Oracle Retail Assortment Planning (AP) to reserve order numbers in Merchandising to be used in orders that will later be created and integrated to Merchandising. This web service contains the following inputs:

Column Name	Notes
Supplier Site ID	Optional - used if the requesting entity is a supplier
Quantity	Indicates the number of order numbers being requested. Required.
Days Until Expiry	Indicates how long the calling solution wants Merchandising to retain the reservation. After this many days, the reservation will be released allowing these numbers to be used for other purposes. This is required and must be a value greater than 0.

The requested order numbers are sent back to the calling solution in the response. This operation is only available as part of the web service.

Create/Manage Purchase Orders

The operations supported in this service for creating and managing purchase orders are as follows:

- Create a purchase order header and details
- Modify purchase order header and details
- Delete purchase order details

For the operations, this service uses the same logic as is used in the PO Subscription RIB API. For more information on the functionality, see [PO Subscription API](#) in the "[RIB Subscription Designs](#)" section of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
preIssueOrderNumber	OrdNumCriVo	OrdNumColDesc	OrdNumCriVo.xsd OrdNumDesc.xsd OrdNumColDesc.xsd
create	XOrderColDesc	InvocationSuccess	XOrderDesc.xsd XOrderColDesc.xsd

Operation Name	Input Object Type	Output Object Type	XML Definition
modifyHeader	XOrderColDesc	InvocationSuccess	XOrderDesc.xsd XOrderColDesc.xsd
createDetail	XOrderColDesc	InvocationSuccess	XOrderDesc.xsd XOrderColDesc.xsd
modifyDetail	XOrderColDesc	InvocationSuccess	XOrderDesc.xsd XOrderColDesc.xsd
deleteDetail	XOrderColRef	InvocationSuccess	XOrderRef.xsd XOrderColRef.xsd

Report Locator Service

Functional Area

Financial Integration

RSB Proxy WSDL

/rms-ReportLocator-AppServiceDecorator/ProxyService/ReportLocatorAppServiceProxy?wsdl

Merchandising Service WSDL

/ReportLocatorBean/ReportLocatorService?WSDL

Overview

This service is used by Oracle Retail Financial Integration (RFI) to retrieve the URL of a BI Publisher report from Merchandising or Sales Audit that can be invoked from the PeopleSoft Financials General Ledger based on a particular journal entry. The report URL that will be returned will be differ based on the ID sent in the service call. Based on that ID, Merchandising will determine if it was a Merchandising, Sales Audit, or Invoice Matching ID and return a URL for the appropriate report. Possible reports for Merchandising and Sales Audit are:

- GL Fixed Deal Data Report
- GL Item level Data Report
- GL Item Rollup Daily Data Report
- GL Item Rollup Monthly Data Report
- GL Sales Audit Data Report

Also, for Invoice Matching, one of the following reports might be returned:

- Merchandise Invoice Document Report
- Non-Merchandise Invoice Document Report
- Credit Note Document Report
- Credit Memo Cost Document Report
- Credit Memo Quantity Document Report

- Debit Memo Cost Document Report
- Debit Memo Quantity Document Report
- Debit Memo VAT Document Report
- Receipt Write Off Document Report

For cloud service implementations, this configuration should be done for you if you are configured to run with PeopleSoft Financials. For on premise implementations, you may need to configure this yourself in the RETAIL_SERVICE_REPORT_URL table. For cloud service implementations, coordinate this configuration with the Oracle Cloud Operations team.

The report is similar functionality to the Drill Forward and Drill Back functionality available in the Merchandising Transaction Data and Fixed Deal pages and the Sales Audit General Ledger Transaction page. See also [Get Drill Back Forward URL Service](#) for more on the APIs that support this functionality.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
publish	ReportLocDesc	ReportLocRef	ReportLocRef.xsd ReportLocDesc.xsd

Store Order Service

Functional Area

Procurement

RSB Proxy WSDL

/rms-StoreOrder-AppServiceDecorator/ProxyService/StoreOrderAppServiceProxy?wsdl

Merchandising Service WSDL

/StoreOrderBean/StoreOrderService?WSDL

Overview

This service is used by Oracle Retail Store Inventory Management (SIM) to create and manage store orders, as well as to query details to support these two operations.

Create Store Order

The majority of the operations in this service are related to creating, updating, or deleting a store order. A store order is a request from the store for inventory that can result in either a purchase order or transfer being created in Merchandising.

The Create operation allows SIM to request the creation of an order for inventory from either a supplier or warehouse for one or more items. If the source of the inventory will be the supplier,

then the order can be for more than one store. For warehouse sourced orders, it will be for a single store.

The Create Detail operation allows SIM to request the addition of an item to a previously created transfer or an item/location to a previously created purchase order.

Orders will be created in either Approved or Worksheet/Input status in Merchandising, depending on what is sent from SIM.

Modify Store Order

The Modify and Modify Detail operations allow SIM to update a previously created transfer or purchase order. For this type of update, SIM must send the status.

Delete Store Order

The Delete and Delete Detail operations allow SIM to request a delete of a previously created order or an order line item. If the order is in approved status and it is being sourced from a supplier, then the result will be a modification of the order to cancel the quantity or full order, rather than delete it. If it is not yet approved, then order details or order can be deleted.

Query Deals

This operation allows SIM to query Merchandising for the deals that an item/store, based on a specific date and source (supplier, partner). Merchandising will reply with details on the off-invoice deals that the item/location are part of based on the date provided. The details provided include the deal dates and discount details.

Query Sales

This operation allows SIM to query Merchandising for a specific item/location combination's sales data. Merchandising will respond by sending the available weeks of sales data, including the quantity sold, retail value, and sales type (for example, regular, promotion, clearance).

Query Store Orders

There are two operations that allow SIM to query store orders from Merchandising. Query Store Order accepts location and location type (store or warehouse), as well as optional filtering details like item, source (supplier site or physical warehouse), dates, and status. The operation returns a collection of header level details for the purchase orders or transfers that match the criteria, including the quantity on the order for the location.

The other operation, Query Store Order Details, accepts a specific order (purchase order or transfer), source type, and source and returns the details of that order, including the destination locations, status, dates, items, cost, and quantity.

Assumptions

- Service operations will return back with the first error encountered.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	LocPOTsfDesc	LocPOTsfRef	LocPOTsfDesc.xsdLocPOTsfRef.xsd
createDetail	LocPOTsfDesc	InvocationSuccess	LocPOTsfDesc.xsd
modify	LocPOTsfDesc	InvocationSuccess	LocPOTsfDesc.xsd
modifyDetail	LocPOTsfDesc	InvocationSuccess	LocPOTsfDesc.xsd
delete	LocPOTsfDesc	InvocationSuccess	LocPOTsfDesc.xsd
deleteDetail	LocPOTsfDesc	InvocationSuccess	LocPOTsfDesc.xsd
queryDeal	LocPOTsfDealsCriVo	LocPOTsfDealsColDesc	LocPOTsfDealsCriVo.xsd LocPOTsfDealsDesc.xsd LocPOTsfDealsColDesc.xsd
queryItemSales	LocPOTsfltmSlsCriVo	LocPOTsfltmSlsColDesc	LocPOTsfltmSlsCriVo.xsd LocPOTsfltmSlsDesc.xsd LocPOTsfltmSlsColDesc.xsd
queryStoreOrder	LocPOTsfHdrCriVo	LocPOTsfHdrColDesc	LocPOTsfHdrCriVo.xsd LocPOTsfHdrDesc.xsd LocPOTsfHdrColDesc.xsd
queryStoreOrderDetail	LocPOTsfDtIsCriVo	LocPOTsfDesc	LocPOTsfDtIsCriVo.xsdLocPOTsfDesc.xsd

Supplier Service

Functional Area

Foundation Data

RSB Proxy WSDL

/rms-Supplier-AppServiceDecorator/ProxyService/SupplierAppServiceProxy?wsdl

Merchandising Service WSDL

/SupplierBean/SupplierService?WSDL

Overview

This service allows Merchandising to subscribe to supplier information from external financial applications. It is also used by Oracle Retail Financials Integration (RFI) for integrating supplier information into Merchandising from EBS, PeopleSoft, or Cloud Financials. The operations supported by this service are as follows:

- Create a new parent supplier, including the associated sites, org unit, and address; it also supports adding flex attributes (CFAS) for supplier, supplier site, and address levels
- Update an existing supplier, including adding or updating sites, org unit, address for the supplier, and flex attributes for the supplier, sites, and address levels

The operations supported by this service involve an external system sending Merchandising details to create or update suppliers or supplier sites. In the response back, the supplier or site IDs are returned on success. The create and update options for this service use the same logic as is supported in the Vendor Subscription RIB API.

For information about this functionality, see [Vendor Subscription API](#) in the "[RIB Subscription Designs](#)" chapter of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	SupplierDesc	SupplierRef	SupplierRef.xsd SupplierDesc.xsd
create	SupplierColDesc	SupplierColRef	SupplierColDesc.xsdSupplierColRef.xsd
update	SupplierDesc	SupplierDesc	SupplierDesc.xsd
update	SupplierColDesc	SupplierColDesc	SupplierColDesc.xsd

Transfer Service

Functional Area

Transfer

RSB Proxy WSDL

```
/rms-TransferManagement-AppServiceDecorator/ProxyService/  
TransferManagementAppServiceProxy?wsdl
```

Merchandising Service WSDL

```
/TransferManagementBean/TransferManagementService?WSDL
```

Overview

Merchandising exposes a Transfer Management service to allow an external application to create, update, and delete transfers. The web service takes in a collection of transfers and will return success and failure through the service response object. The operations supported in this service for creating and managing transfers are as follows:

- Create a transfer header and details
- Modify transfer header and details
- Delete transfer header and details

For the operations, this service uses the same logic as is used in the Transfer Subscription RIB API. For more information on the functionality, see [Transfer Subscription API](#) in the "[RIB Subscription Designs](#)" chapter of this document.

Operation XSD

Here are the filenames that correspond with each operation. Please consult the RSB documentation for each in order to get a detailed picture of the composition.

Operation Name	Input Object Type	Output Object Type	XML Definition
create	XTsfColDesc	InvocationSuccess	XTsfDesc.xsd
			XTsfColDesc.xsd
createDetail	XTsfColDesc	InvocationSuccess	XTsfDesc.xsd
			XTsfColDesc.xsd
modifyHeader	XTsfColDesc	InvocationSuccess	XTsfDesc.xsd
			XTsfColDesc.xsd
modifyDetail	XTsfColDesc	InvocationSuccess	XTsfDesc.xsd
			XTsfColDesc.xsd
deleteHeader	XTsfColRef	InvocationSuccess	XTsfRef.xsd
			XTsfColRef.xsd
deleteDetail	XTsfColRef	InvocationSuccess	XTsfRef.xsd
			XTsfColRef.xsd

Consumer Services

This section gives an overview about the SOAP Web service Consumer Implementation API designs used in the Merchandising environment and various functional attributes used in the APIs.

Customer Address Service

Functional Area

Financials

Overview

The primary role of this service is to query customer address details related to a Sales Audit transaction. This may be required if you have configured Merchandising to not retain customer information (Retain Customer Information system options unchecked) for customer orders, but you wish to have visibility to it in Sales Audit when viewing/auditing transactions.

When Sales Audit calls this service, it will pass the customer ID and expect to receive back the following information in response:

- First Name
- Last Name
- Company Name (if applicable)
- Address Line 1
- Address Line 2

- Address Line 3
- County
- City
- State
- Country
- Postal Code
- Jurisdiction
- Phone
- Email
- Birthdate

As part of your implementation, if you have this system option configured off, you will need to provide a URL for Sales Audit to call. For on premise implementations, this will require updating the RETAIL_SERVICE_REPORT_URL table for code CAS. For cloud service implementations, configuration of this service call should be done in coordination with the Oracle Cloud Operations team by logging an SR.

Customer Order Address Service

Functional Area

Procurement

Overview

The primary role of this service is for Merchandising to query customer/shipping details related to a customer order from an order management system (OMS). This is required if you have configured Merchandising to not retain customer information (Retain Customer Information system options unchecked) for customer orders and are sourcing customer orders from a warehouse or supplier where Merchandising needs to provide the address details for shipping to the customer.

When Merchandising calls this service, it will pass

- Customer order number
- Fulfillment order number
- "Fulfillment location type and ID
- Source location type and ID

And expect to receive back the following information in response:

- Customer ID
- Delivery Details
 - First Name
 - Phonetic First Name
 - Last Name
 - Phonetic Last Name

- Preferred Name
- Company Name
- Address Line 1
- Address Line 2
- Address Line 3
- County
- City
- State
- Country
- Postal Code
- Jurisdiction
- Phone
- Email
- Billing Details
 - First Name
 - Phonetic First Name
 - Last Name
 - Phonetic Last Name
 - Preferred Name
 - Company Name
 - Address Line 1
 - Address Line 2
 - Address Line 3
 - County
 - City
 - State
 - Country
 - Postal Code
 - Jurisdiction
 - Phone
 - Email

As part of your implementation, if you have this system option configured off, you will need to provide a URL for Merchandising to call. For on premise implementations, this will require updating the `RETAIL_SERVICE_REPORT_URL` table for code COA. For cloud service implementations, configuration of this service call should be done in coordination with the Oracle Cloud Operations team by logging an SR.

Get Drill Back Forward URL Service

Functional Area

Financial Integration

Overview

If you are implementing the Merchandising solutions with PeopleSoft Financials, then this service can be used allow users in Merchandising or Sales Audit to "drill forward" into Peoplesoft to view the General Ledger journal entries associated with a transaction or to "drill back" into Merchandising and Sales Audit from PeopleSoft General Ledger screens to view the source transactions associated with a journal entry. Both of these actions leverage this service call.

If you are configured use Merchandising with PeopleSoft Financials, then when you are in the following pages, you'll see Drill to Finance options that leverage this call:

- Merchandising Transaction Data
- Merchandising Fixed Deals
- Sales Audit General Ledger Transactions

Note

Oracle Retail Invoice Matching also leverages this service for viewing transactions in PeopleSoft Payables. And Peoplesoft Payables can drill back to Invoice Matching as well.

As part of your implementation, you will need to configure the URL for the service call in the `RETAIL_SERVICE_REPORT_URL` table for code RDF. For cloud service implementations, configuration of this service call should be done in coordination with the Oracle Cloud Operations team by logging an SR. For more information, see the *RFI Implementation Guide*.

GL Chart of Accounts Validation Service

Functional Area

Financial Integration

Overview

When using Oracle Retail Financials Integration (RFI) to manage General Ledger integration an Oracle financial solution, a validation service is used to ensure that the segment combinations mapped to by Merchandising and Sales Audit users are valid combinations in the General Ledger. This validation is called from Merchandising and Sales Audit when creating General Ledger cross-reference mappings.

Note

This validation is also used by Oracle Retail Invoice Matching

As part of your implementation, you will need to configure the URL for the service call in the RETAIL_SERVICE_REPORT_URL table for code RAV. For cloud service implementations, configuration of this service call should be done in coordination with the Oracle Cloud Operations team by logging an SR. For more information, see the *RFI Implementation Guide*.

5

ReSTful Web Services

This chapter gives an overview about the Merchandising and Sales Audit ReSTful Web service implementation and the APIs used in Merchandising and Sales Audit. For more information on ReST architectural style applied for building Web services, access the following URL:

<http://www.oracle.com/technetwork/articles/javase/index-137171.html>

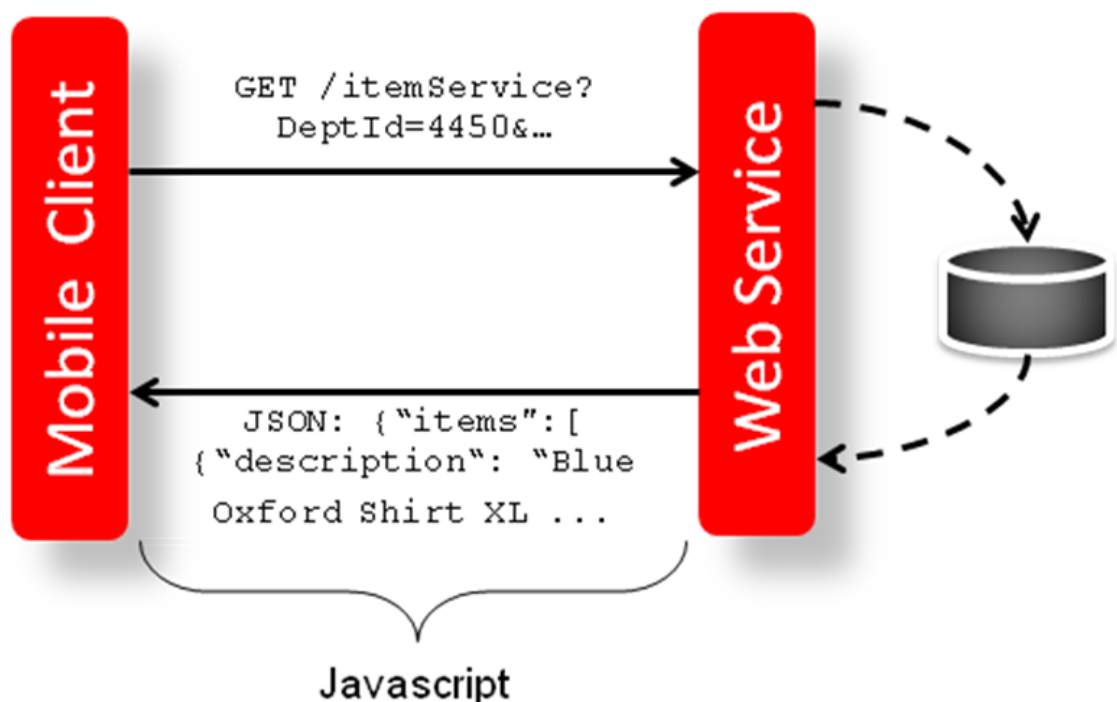
To provide visibility to the background processing that's occurring, services write to the JOB_AUDIT_LOGS table in the database. Reports can be built based on this to provide visibility to what is happening in the background. Additionally, to assist users and developers in troubleshooting any error that may arise, the payload that was processed is also stored in the JOB_AUDIT_PAYLOAD table in the database.

Once the nightly batch run has started, web service execution will be halted, and users will receive a warning message that the nightly batch run has commenced.

Introduction

Merchandising and Sales Audit ReST support several web services, including the ability to query data and the ability to create and update data within the solutions. The ReSTful Web services Java code cannot be customized. The diagram below shows how the services are intended to interact with an external application which is a mobile client.

Figure 5-1 Mobile Client and Web Services Integration through Javascript



Note

Several of the ReST services that may impact inventory or conflicts with data processing happening during nightly batch should not be used during the restricted nightly batch window.

Common Characteristics

Security

Services are secured using J2EE-based security model.

- Realm-based User Authentication: This verifies users through an underlying Realm. The username and password are passed using HTTP basic authentication.
- Role-based Authorization: This assigns users to roles; authenticated users can access the services with Merchandising or Sales Audit application roles or custom roles that are assigned to:
 - For Merchandising MERCH_SERVICE_ACCESS_PRIV
 - For Sales Audit MERCH_SERVICE_ACCESS_PRIV
- The communication between the server and client is encrypted using one-way SSL. In non-SSL environments the encoding defaults to BASE-64 so it is highly recommended that these ReST services are configured to be used in production environments secured with SSL connections.
- If you are using Merchandising data filtering, that will apply to the services as well. The app user ID used for the calling the service should be added to the Merchandising SEC_USER table (APP_USER_ID), and then associated to the appropriate group in SEC_USER_GROUP table. For more information on this see the *Merchandising Security Guide - Volume 2*.

Note

oAuth App/Client Name (which is different than client ID) should be configured in the SEC_USER table.

Standard Request and Response Headers

Merchandising and Sales Audit ReSTful web services have the following standard HTTP headers:

Accept: application/xml or application/JSON
Accept-Version: 16.0 (service version number)
Accept-Language: en-US,en;q=0.8

Note

Accept-Language is not mandatory, and defaults to en-US. The user can change it though; in case they need content in a specific language. The language support in the response payload is limited to a selective service which was created for the mobile client use case.

Depending on the type of the operation or HTTP method, the corresponding response header is updated in the HTTP response with the following codes:

- GET/READ : 200
- PUT/CREATE : 201 (Created) or 200 (Success)
- POST/UPDATE : 204 or 200 (Success)
- DELETE : 204 or 200 (Success)

Standard Error Response

Example response payload in case of service error is depicted below:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<messagesRDOes>
  <messagesRDO>
    <message>REST Service Version Mismatch</message>
    <messageType>ERROR</messageType>
    <status>BAD_REQUEST</status>
  </messagesRDO>
</messagesRDOes>
```

- **Message:** The error message - translated.
- **Message Type:** Value of 'ERROR' is returned.
- **Status:** For a bad request or error, the status is BAD_REQUEST.
- The http error code for an error response is 400.

Merchandising URL Paths

Based on your implementation you will need to prefix the end point with just the deployment hostname. The format that should be used for the hostname is `rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace>/`, where `<Region Name>` will be the region for your data center and `<Customer Subnamespace>` will be specific to your company's name and environment (production, stage, and so on).

The Merchandising services are hosted on two separate application servers. The application server named `MerchIntegrations` contains new Integration web services for data publishing and subscription. The `RmsReSTServices` contains other, legacy web services.

The following links provide access to the Merchandising services:

- The ReSTful Web services swagger URL is available at:
`https://<hostname/namespace>/RmsReSTServices/swagger-ui/`
`https://<hostname/namespace>/MerchIntegrations/swagger-ui/`
- The ReSTful Web services `openapi.yaml` is available at:

- `https://<hostname/namespace>/RmsReSTServices/services/openapi.yaml`
`https://<hostname/namespace>/MerchIntegrations/services/openapi.yaml`
- The ReSTful Web services's WADL file is available at:
`https://<hostname/namespace>/RmsReSTServices/services/private/application.wadl`
`https://<hostname/namespace>/MerchIntegrations/services/application.wadl`
- The ReSTful Web services are available at:
`https://<hostname/namespace>/RmsReSTServices/services/private/<service>`
`https://<hostname/namespace>/MerchIntegrations/services/<service>`

Sales Audit URL Paths

Based on the Implementation you may need to prefix the end point with just the deployment hostname or hostname plus access port.

The following links provide access to the Sales Audit services:

- The ReSTful Web services swagger URL is available at:
`https://<hostname/namespace>/ResaReSTServices/swagger-ui/`
- The ReSTful Web services openapi.yaml is available at:
`https://<hostname/namespace>/ResaReSTServices/services/openapi.yaml`
- The ReSTful Web services's WADL file is available at:
`https://<hostname/namespace>/ResaReSTServices/services/private/application.wadl`
- To access the ReSTful web services:
`https://<hostname/namespace>/ResaReSTServices/services/private/<ResaService>`

Fiscal Document Generation URL Paths

Based on your implementation you will need to prefix the end point with just the deployment hostname. The format to be used for the hostname is xxx-yyy-mfcs-rhs.oracleindustry.com, where xxx-yyy is specific to your company's name and environment (production, stage, and so on). The following links provide access to the Fiscal Document Generation services:

- The ReSTful Web services Javadocs are available at:
`https://<hostname>/RfmReSTServices`
- The ReSTful Web services's WADL file is available at:
`https://<hostname>/RfmReSTServices/services/private/application.wadl`
- The ReSTful Web services are available at:
`https://<hostname>/RfmReSTServices/services/private/application.wadl`

OAuth for REST Service Authentication

OAuth 2.0 is the industry standard protocol for authorization. Merchandising cloud services REST Services now supports OAuth 2.0. To invoke these services, you need to obtain and access token and use it as a bearer token.

Note

Basic Authentication access is no longer supported for these services. Allocation Cloud Service will still use Basic Authentication with this release.

Steps to Enable

To obtain a token and call the services, use the following steps:

1. This step should be performed one time for a given environment. Generate OAuth Client App with the environment-specific scope.

Each environment has a different scope that is available after the provisioning of the environment. Use this scope to generate the OAuth Client App using the Retail Home UI. For details, refer to the "Creating OAuth Client Apps" section of the *Retail Home Administration Guide*, which can be found under Common Cloud Services Solutions in the [Oracle Retail Help Center](#). Once the OAuth Client App is generated, note the client ID and secret. This information will be required for an access token to call REST services.

2. Get an access token using the OAuth client ID and secret from IDCS.

```
export ACCESS_TOKEN="$(curl -u <Client ID>:<Secret> -H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' --request POST https://<IDCS_BASE_URL>/oauth2/v1/token -d 'grant_type=client_credentials&scope=<Customer Environment Specific Scope>' | jq -r '.access_token')"
```

The token is generally valid for 1 hour.

3. REST clients that need to call Merchandising REST service end points should use the client ID and secret of the OAuth client generated in the previous step to get an access token.

OAuth tokens can also be obtained by REST client tools like Postman for testing purposes by filling in the necessary details like client ID/secret and scope. Use the below information in such cases:

- Authorization: OAuth 2.0
- Access Token URL: `https://<IDCS_BASE_URL>/oauth2/v1/token`
- Client ID: <Client id of OAuth client app>
- Client Secret: <Client secret of OAuth client app>
- Scope: <Custom environment specific scope>
 - The scope pattern that is used in the MFCS IDCS app creation template is `rgbu:merch:<SERVICETYPE>-<ENVIRONMENT>` where `SERVICETYPE` is MFCS and `ENVIRONMENT` is the environment type (STG, PRD, UAT, DEV1, DEV2, and so on).

For example:

- * "scope": "rgbu:merch:MFCs-PRD"
- * "scope": "rgbu:merch:MFCs-STG"

Example using the access token as a bearer token:

```
curl --location --request GET 'http://<hostname or IP address>:<port number>/  
RmsReSTServices/services/private/Common/vDate' \ --header 'Authorization:  
Bearer $ACCESS_TOKEN'
```

Date Format

Few input date and output date fields are in long format. The others are in SQL Date format. The new MerchIntegrations service supports dates in the yyyy-mm-dd format and datetimes in the yyyy-mm-ddTHH24:Mi:SS.SSSZ format.

Dynamic Response Field Filtering

The ReST service call responses can be customised for precise data retrieval through the use of the include and exclude query parameters. These parameters provide the capability to selectively choose data fields to be included or excluded in the API response to ensure the presence of only relevant information.

'Include' Parameter

Through the 'include' parameter, you can provide a list of specific attributes/data fields to be included in the specific response from available fields in the service. This list should be formatted using JSON dot notation; for example, `items.item`, `items.status`, `items.dept`, `items.itemSupplier.supplier`. Use of this parameter will make the API response concise, containing only the requested fields.

'Exclude' Parameter

In contrast, the exclude parameter offers a complementary approach. You can provide a list of attributes to be excluded from the response, using JSON dot notation. The API then returns a comprehensive dataset, filtered to exclude only the specified fields. This is particularly useful when most of the data is required but you need to omit certain fields from the response.

For details on the Publish API, refer to reference paper *Publish APIs - Customer Enablement Overview* available in Oracle Retail Merchandising Documentation Library (Doc ID [1585843.1](#)) in My Oracle Support.

Limited Availability During Nightly Batch Cycle

Several of the consume ReST services can interfere with batch jobs running during the MFCs nightly batch window. To avoid data processing conflicts, several ReST services are either not available during the entire duration of the nightly batch window or unavailable for a portion of the nightly batch. These services will fail with a 400 error if they are invoked during the window of unavailability.

The availability of the Rest services is categorized as:

1. **Available 24/7** – These services are available 24/7 except for the time during which the application is being patched. Most services are available 24/7 and this includes all the GET services and most of the foundation data setup.

2. **Not available during entire nightly batch** – A small set of services are not available during the entire window of the nightly batch cycle. This includes services that are only active on certain dates (like stock count schedule, cost changes) or services that can influence the business logic of several jobs (like update item, item/supplier, store, reclassification, and so on).

The POM job `START_BATCH_JOB` disables these services by setting the value of `batch_running_ind` to Y in the `rms_batch_status` table. The POM job `END_BATCH_JOB` switches the indicator back to N and the services are then available to process the data.

3. **Not available during Inventory batch window** - The web services that impact inventory buckets are not available during portions of the nightly batch during which inventory-related batch processes are running. This includes the services related to the functionality of receipts, transfers, allocations, inventory adjustment, customer orders, and so on.

The POM job `STOP_RIB_ADAPTOR_INV_JOB` disables these services by setting the value of `inventory_batch_running_ind` to Y in the table `rms_batch_status`. The job `START_RIB_ADAPTOR_INV_JOB` switches the indicator to N to reenables the services.

4. **Not available during Costing and Inventory batch window** – Some of the services interfere with (in addition to impacting) the inventory buckets, which also impacts the cost component and expenses. These services are not available during the inventory batch window (`inventory_batch_running_ind = Y`) and while costing batch jobs are running at the initial phase of the nightly batch. This includes services related to functionality of purchase orders, item expenses, item HTS, supplier updates, exchange rate updates, and so on.

The POM job `DUMMY_START_NIGHT_BATCH_JOB` disables these services during the costing batch window by setting `costing_batch_running_ind` to Y in the `rms_batch_status` table. The job `END_COSTING_BATCH_JOB` switches the indicator to N, which reenables the services once the costing batches are complete.

Note

MerchIntegration services having limited availability during nightly batch have a custom property named `x-batch-dependency` in the `openapi.yaml` file.

Custom HTTP Headers

The Rest Service supports custom HTTP Headers that can be used to exchange additional metadata about the service calls. These are:

- **X-Correlation-ID:** The client can include a unique ID in the HTTP header “X-Correlation-ID” to uniquely trace the service call. The input X-Correlation-ID is captured in server logs and in the `service_metrics` table to provide details of the outcome of the service call. While the expectation is that the client should have a unique correlation ID in every request, the uniqueness is not validated by the ReST service.

In the response HTTP header, X-Correlation-ID is always included. If the request HTTP header contains a X-Correlation-ID, the same value IS returned in the response HTTP header. If the input header is missing the correlation id, a server generated request id (UUID) IS populated in the response X-Correlation-ID header field. The server generated request ID is captured in `service_metrics` table.

Best Practice – The client should generate a UUID (up to 36 bytes) and include the UUID in the request in the HTTP header X-Correlation-ID and persist the ID in some data store

for traceability purposes. In case of exceptions or troubleshooting, this ID is handy when checking the table `service_metrics` for how the server processed the request.

Example usage: The client makes a call and the client crashes before the response is received, or there was timeout thrown by one of the infrastructure components between the client and the application server (like load balancer or routers). The correlation ID can be checked in the table `service_metrics` to identify the actual response. If the service method is POST or PUT or DELETE, the table `service_payload` can be referenced to get the response payload for that request.

Note

Capturing X-Correlation-ID in server logs or in the table `service_metrics` is done by the ReST Service application server and the request should have arrived at this server for the service request to be logged. The application server captures the request in the `service_metrics` table when the request is received. After the request is processed, the server updates the same row in the `service_metrics` table to capture the response status.

- **X-Client-Principal-User:** The client can include an optional HTTP Header in the request to provide a business user identifier in a system-to-system service call to allow for capturing the business user that created or updated the content of the request. As part of the service request in which data is created or updated, the audit columns `create_id` and `last_update_id` capture the OAuth client name used to make the service call. In the request containing X-Client-Principal-User, this value is used as-is (without any validation) to populate the audit columns.
- **X-Content-Has-More:** This HTTP Header (true or false) is included in the response for several Merch Integration Publish API. The GET services that supports pagination includes the field `hasMore` in the payload to inform the client if there are additional pages to be requested. This information is also captured in the HTTP Header to allow the client to quickly access this information without parsing the response payload.
- **X-Content-Offset-Key:** This HTTP Header is included in the response, along with X-Content-Has-More, when there are additional pages to be requested for a service supporting pagination using the query parameter `offsetkey`. The client can use the value from the HTTP Header instead of reading it from inside the payload to construct the URL for the next page.
- **X-Content-Count:** This HTTP Header is included as an additional information in several Merch Integration Publish APIs to inform the client of the number of records in the response payload.

Paging

Some of the Merchandising and Sales Audit ReSTful web services have the potential to bring back a significant number of records, and therefore these services are equipped to segment the result into pages. The page number to retrieve and the size of the page are added as input parameters to all the paged services.

Each paged result includes the following information:

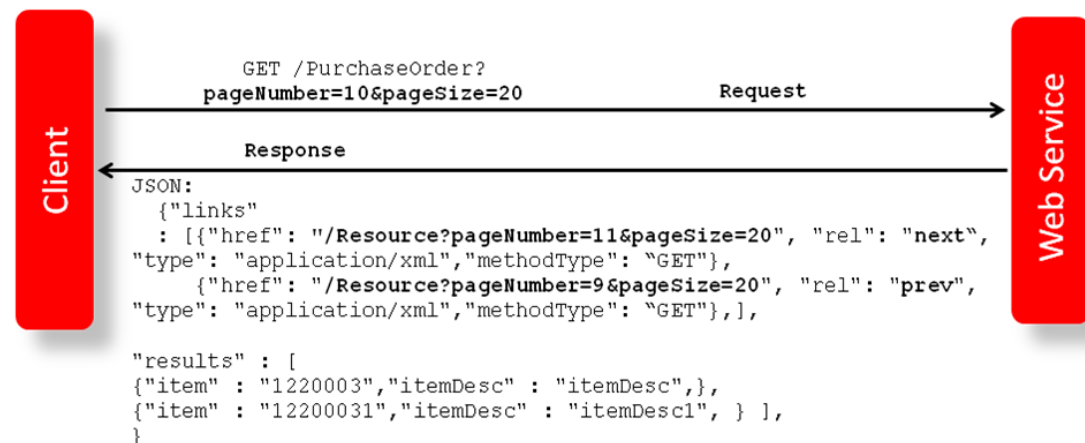
- **Total Record Count:** Displays the number of all records matching the service input criteria.
- **Next Page URL:** Shows the service URL with same input parameters, but with the `pageNumber` plus 1, when more records exist.

- **Previous Page URL:** Shows the service URL with same input parameters and the pageNumber input value minus 1, when page number is not 1.

Next or previous page URL is not provided when:

- No records are returned
- Previous page is not returned, when the page number is 1.
- Next page is not returned, when the record reaches the last page.

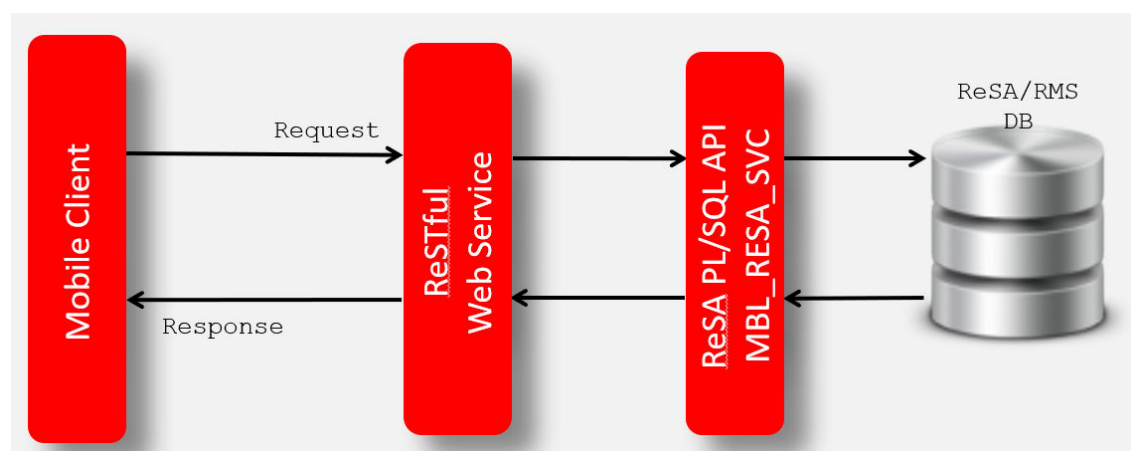
Figure 5-2 Javascript for Paging Information in RMS Web Services



Web Service APIs Process Flow

The diagram shows the Web Service API process flow.

Figure 5-3 Web Service APIs Process Flow



Configuring ReSTful Web Services for Integration with External Systems

Based on your implementation, you will need to enable the Merchandising ReSTful publishing services for data replication between Merchandising and external systems. Additionally, the

Omnichannel integration ReSTful services may also be enabled for external consumers. These two enablements are performed through the Web Service Configuration Screen in Merchandising. After a service is enabled, the table triggers will start writing data to respective change log tables and the corresponding background job will start populating JSON cache tables. When a service is disabled, the ReST service will be unavailable, triggers will stop capturing new changes, and background process will clean up the JSON cache table. For details on the Publish API, refer to reference paper [Publish APIs - Customer Enablement Overview](#) available in Oracle Retail Merchandising Documentation Library (Doc ID [1585843.1](#)) in My Oracle Support. For more details on how to enable a web service, see the *Merchandising Administration Guide*.

The below listed services can be enabled or disabled using the Merchandising UI,

- [Get Allocation Details for Provided Allocation](#)
- [Get Differentiator Groups](#)
- [Get Differentiator Details](#)
- [Get Differentiator Type Details for Provided Type](#)
- [Get Item Available Inventory at Customer Orderable Locations](#)
- [Get Item Location Details](#)
- [Get Item Details for All or Provided Items](#)
- [Get Item VAT Details](#)
- [Get Class Details for Provided Class](#)
- [Get Purchase Order Details](#)
- [Get Organizational Hierarchy](#)
- [Partner Publish Services](#)
- [Purchase Order Work Order Publish Services](#)
- [RTV Publish Services](#)
- [Get Receiver Unit Adjustment](#)
- [Get Store Details](#)
- [Supplier Publish Services](#)
- [Get Transfer](#)
- [Transfer Work Order Publish Services](#)
- [REST Endpoint for UDA details](#)
- [Get Warehouse Details](#)
- [REST Endpoint for Omni channel/Diff Group](#)
- [REST Endpoint for Omni channel/diff Ids](#)
- [REST Endpoint for Omnichannel/Dimension Type details](#)
- [REST Endpoint for Omnichannel/Dimension Value details](#)
- [REST Endpoint for Omnichannel/future Inventory](#)
- [REST Endpoint for Omni channel/Inventory/Store details](#)
- [REST Endpoint for Omni channel/Inventory/Warehouse details](#)
- [REST Endpoint for Omnichannel/item Image](#)

- [REST Endpoint for Omnichannel/Item Initial Price details](#)
- [REST Endpoint for Omnichannel/Item Location details](#)
- [REST Endpoint for Omni channel/ Item details](#)
- [REST Endpoint for Omni channel/ Item UPC details](#)
- [REST Endpoint for Omni channel/ Merchandise Hierarchy details](#)
- [REST Endpoint for Omni channel/Org hier descendant's details](#)
- [REST Endpoint for Omni channel/Org hierarchy details](#)
- [REST Endpoint for Omni channel/refresh Date](#)
- [REST Endpoint for Omnichannel/related Item](#)
- [REST Endpoint for Omnichannel/Retail Store details](#)
- [REST Endpoint for Omnichannel/vat](#)
- [REST Endpoint for Omni channel/Warehouse details](#)

File Transfer Services (FTS)

Business Overview

File Transfer Services (FTS) provide an endpoint for the customer to be used when working with files that are uploaded to and downloaded from the Merchandising Cloud Service Suite. These services are used by Merchandising and Sales Audit (RMFCS), Pricing (RPCS), Invoice Matching (ReIMCS), Allocation (RACS) and the Data Conversion tool. They allow you to manage uploading and downloading files to Oracle Cloud Infrastructure Object Storage, which is an internet-scale, high-performance storage platform that offers reliable and cost-efficient data durability.

For each customer environment, buckets, which are logical containers for storing objects, will be created in Object Storage. Any type of data, regardless of content type, is stored as an object. An object is composed of the object itself and metadata about the object. Any file uploaded through FTS to Object Storage has a retention period of 7 days before it is automatically purged.

The services use OAUTH 2.0 client credential authentication. Once authentication passes, the internal file transfer services that retrieve a pre-authenticated request (PAR) are called to upload and download files to Object Storage. The primary role of these services is to ensure that only valid external users can call the service by enforcing authorization policies.

List of services:

- [List Storage Prefixes](#)
- [Retrieve PAR for Uploading Files](#)
- [Retrieve PAR for Downloading Files](#)
- [Delete Files](#)
- [List Files](#)
- [FTS Health Check](#)

URL Prefix: `https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace>/RmsPlatformServices/services/private`

The final URL will be the URL Prefix followed by the ReST URL path mentioned in each service description below.

To start uploading or downloading the files to/from Object Storage, storage prefixes should be registered first.

The table below shows the prefixes used by each of the solutions when calling the above listed services. By default, all the storage prefixes below will be registered automatically.

	RMFCS	RPCS	ReIMCS	RACS	Data Conversion
Incoming	/incoming		/incoming/reim	/incoming/ alloc	/dataconversion/ incoming
Outgoing	/outgoing				/dataconversion/ outgoing
Reject	/reject			n/a	n/a
Downloaded	/downloaded		/downloaded/reim	/downloaded/ alloc	n/a
Archive		n/a	n/a	n/a	n/a

Table 5-1 Administrative Processes

Storage Prefix	Service	File Name
AdminProcess	Dynamic Hierarchy	dynamic_hierarchy/ dynamic_hierarchy_mapping_tl.csv

List Storage Prefixes

Business Overview

This service is used for retrieving Object Storage prefixes.

Service Type

Get

ReST URL

/FTSWrapper/listprefixes

Headers

Key	Value
Accept-Language	String (Ex: en-US)

Output

The output metadata contains the prefix details of the request including validation error, if any.

For Success:

```
[
    "downloaded",
    "incoming",
]
```

For Failure:

```
{
    "error_message": "string"
}
```

Retrieve PAR for Uploading Files

Business Overview

This service is used for retrieving an Object Storage pre-authenticated request (PAR) for uploading files. The response from the service is then used to upload the files in the user specified prefix from Object Storage using curl. The specified prefix is the staging location for input file that is processed by one of the Merchandising cloud service upload batches.

Service Type

Post

ReST URL

/FTSWrapper/upload

Headers

Key	Value
Accept-Language	String (Ex: en-US)

Request body

```
{
  "listOfFiles": [
    {
      "storagePrefix": "string",
      "fileName": "string"
    }
  ]
}
```

Output

Parameter Name	Data Type
id	String
name	String
accessUri	String
objectNames	<List> String
access	String
timeExpires	Timestamp
timeCreated	Timestamp

The output metadata contains the PAR details of the request including validation error, if any

For Success:

```
{
  "id": "string",
  "name": "string",
  "accessUri": "string",
  "objectNames": [
    "string"
  ],
  "accessType": "string",
  "timeExpires": "timestamp",
  "timeCreated": "timestamp"
}
```

For Failure:

```
{
  "error_message": "string"
}
```

Uploading the Batch Incoming File

To upload the file, use a `curl` command as shown below in the example, including the `accessUri` returned from the ReST service call. Provide the location of the file to be uploaded. The file must be uploaded before the expiration time. Below is an format of the `curl` command structure. The specified prefix is the staging location for input files that is processed by one of the Merchandising cloud service upload batches.

Example

```
curl https://<objectstorage_hostname>/p/<pre authenticated request>/n/
<namespace>/b/<bucketname>/o/<prefixname>/<filename> --upload-file <filename>
```

Retrieve PAR for Downloading Files

Business Overview

This service is used for retrieving an Object Storage pre-authenticated request (PAR) for downloading files. The response from the service is then used to download the files in the user specified prefix from Object Storage using `curl`. The specified prefix is the staging location for file that is processed by one of the Merchandising cloud service batches.

Service Type

Post

ReST URL

/FTSWrapper/download

Headers

Key	Value
Accept-Language	String (Ex: en-US)

Request Body

```
{
  "listOfFiles": [
    {
      "storagePrefix": "string",
      "fileName": "string"
    }
  ]
}
```

Output

Parameter Name	Data Type
id	String
name	String
accessUri	String
objectNames	<List> String
access	String
timeExpires	Timestamp
timeCreated	Timestamp

The output metadata contains the PAR details of the request including validation error, if any.

For Success:

```
{
  "id": "string",
  "name": "string",
  "accessUri": "string",
  "objectNames": [
    "string"
  ],
  "accessType": "string",
  "timeExpires": "timestamp",
  "timeCreated": "timestamp"
}
```

For Failure:

```
{
  "error_message": "string"
}
```

Downloading the File

To download the file, use the `curl` command to the `accessUri` returned from the output metadata. You can download the file before the expiration time. The specified prefix is the staging location for file that is processed by one of the Merchandising cloud service batches.

Example

```
curl https://<objectstorage_hostname>/p/<pre authenticated request>/n/
<namespace>/b/<bucketname>/o/<prefixname>/<filename> -O
```

Delete Files

Business Overview

This service is used for deleting files in Object Storage based on the provided prefix.

Service Type

Delete

ReST URL

/FTSWrapper/delete

Headers

Key	Value
Accept-Language	String (Ex: en-US)

Request body

```
{
  "listOfFiles": [
    {
      "storagePrefix": "string",
      "fileName": "string"
    }
  ]
}
```

Output

The output metadata contains the output of the FTS service for deletion. Validation error will be output, if any.

For Success:

```
{
  "filesDeleted": [
    {
      "filePath": {
        "storagePrefix": "incoming",
        "fileName": "testFile.txt"
      },
      "responseMessage": "File successfully deleted incoming/
testFile.txt"
    }
  ],
  "filesFailedDeletion": []
}
```

For Failure:

```
{
  "error_message": "string"
}
```

List Files

Business Overview

This service provides a list of Object Storage files under the specified directory/prefix.

Service Type

Get

ReST URL

/FTSWrapper/listfiles

Headers

Key	Value
Accept-Language	String (Ex: en-US)

Input Parameters

Parameter Name	Data Type
prefix	String
contains	String
scanStatus	String
Limit	String
Offset	String
Sort	String

Output

Parameter Name	Data Type
name	String
size	String
md5	String
version	String
etag	String
createdDate	Timestamp
modifiedDate	Timestamp
scanStatus	String
scanDate	Timestamp
scanMessage	String

The output metadata contains the file detail of the request.

For Success:

```
{
    "name": "string",
    "size": "string",
```

```
    "md5": "string",
    "version": "string",
    "etag": "string",
    "createdDate": "timestamp",
    "modifiedDate": "timestamp",
    "scanStatus": "string",
    "scanDate": "timestamp",
    "scanMessage": "string",
  }
```

FTS Health Check

Business Overview

This service checks the status of the File Transfer Service (FTS).

Service Type

Get

ReST URL

/FTSWrapper/ping

Headers

Key	Value
Accept-Language	String (Ex: en-US)

Output

The output metadata contains output of the FTS service for Health check which shows the status. Validation error will be output, if any.

For Success:

```
{
  "appStatus": 200
}
```

For Failure:

```
{
  "error_message": "string"
}
```

Merchandising to Omnichannel Services

The following services are included in this functional area:

- [REST Endpoint for Omni channel/Diff Group](#)
- [REST Endpoint for Omni channel/diff Ids](#)
- [REST Endpoint for Omni channel/Inventory/Store details](#)
- [REST Endpoint for Omni channel/Inventory/Warehouse details](#)
- [REST Endpoint for Omni channel/ Item details](#)
- [REST Endpoint for Omni channel/ Item UPC details](#)
- [REST Endpoint for Omni channel/ Merchandise Hierarchy details](#)
- [REST Endpoint for Omni channel/Org hier descendant's details](#)
- [REST Endpoint for Omni channel/Org hierarchy details](#)
- [REST Endpoint for Omni channel/refresh Date](#)
- [REST Endpoint for Omni channel/Warehouse details](#)
- [REST Endpoint for Omnichannel/related Item](#)
- [REST Endpoint for Omnichannel/Retail Store details](#)
- [REST Endpoint for Omnichannel/Dimension Type details](#)
- [REST Endpoint for Omnichannel/Dimension Value details](#)
- [REST Endpoint for Omnichannel/future Inventory](#)
- [REST Endpoint for Omnichannel/item Image](#)
- [REST Endpoint for Omnichannel/Item Initial Price details](#)
- [REST Endpoint for Omnichannel/Item Location details](#)
- [REST Endpoint for Omnichannel/vat](#)

REST Endpoint for Omni channel/Diff Group

Endpoint

MerchIntegrations/services/foundation/omnichannel/diff/group

Functional Area

Items - Item Foundation

Business Overview

This service publishes differentiator groups to the Omnichannel application and can also be consumed by downstream consuming systems. Functionally, the source view uses the differentiator-group header-plus-detail relationship, so only differentiator groups that currently have detail rows are published.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIFF_GROUP_HEAD, DIFF_GROUP_DETAIL

Additional business tables: DIFF_TYPE

JSON cache table: MERCHAPI_DIFF_GROUP

JSON generation view: V_MERCHAPI_DIFF_GROUP_JSON

The cache is refreshed from V_MERCHAPI_DIFF_GROUP_JSON, which builds the payload from DIFF_GROUP_HEAD, DIFF_TYPE, and DIFF_GROUP_DETAIL.

The view uses the header-plus-detail relationship, so only differentiator groups that currently have detail rows are published.

When the omnichannel dimension-type API is enabled, the same diff-group processing flow also repopulates the child dimension-type cache for impacted items.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_DIFF_GROUP_JSON. No additional filter criteria were identified beyond the cache-build logic.

Rebuild marks cache rows deleted when the group no longer exists in V_MERCHAPI_DIFF_GROUP_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIFF_GROUP (ICL consumer name = MERCHAPI)

Delta processing is driven by ICL_RMS_DIFF_GROUP.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_DIFF_GROUP

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_DIFF_GROUP_ADHOC_PROCESS ->
MERCHAPI_OMNI_DIFF_GROUP_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/diff/group

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/diff/group

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-2 PageResultsItemsVMerchapiDiffGrp - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-3 VMerchapiDiffGrp - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Diff group.
diffgroupid	No	String (10)	This column contains the Diff group ID.
diffgroupdesc	No	String (120)	This column contains the Diff group's description.
difftypeid	No	String (6)	This column contains the Diff type ID.
diffid	No	String (10)	This column contains the Diff ID.

Table 5-4 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "diffgroupid": "BASIC CLRS",
      "diffgroupdesc": "Basic Color Pallet",
      "difftypeid": "C",
      "diffid": "GREEN"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

```

}
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "diffgroupid": "1177",
      "diffgroupdesc": "EIT_Diff_Group_1",
      "difftypeid": "CF",
      "diffid": "4280"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/diff/group?
before=2021-05-06T16%3A12%3A02.551757%2B08%3A00&limit=1&since=1970-01-01T00%3A
00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http:// <hostname>/MerchIntegrations/services/foundation/
omnichannel/diff/group?
before=2021-05-06T16%3A12%3A02.551757%2B08%3A00&limit=1&offsetkey=1177%2C4280&
since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}

```

```
} ]  
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_GROUP	Yes	Yes	Yes	Yes
MERCHAPI_DIFF_GROUP	Yes	Yes	Yes	Yes
V_MERCHAPI_DIFF_GROUP	Yes	No	No	No
V_MERCHAPI_DIFF_GROUP_JSON	Yes	No	No	No
V_MERCHAPI_ITEM_JSON	Yes	No	No	No
MERCHAPI_ITEM_DIFF_TYPE	Yes	Yes	Yes	Yes

REST Endpoint for Omni channel/diff Ids

Endpoint

MerchIntegrations/services/foundation/omnichannel/diff

Functional Area

Items - Item Foundation

Business Overview

This service publishes differentiator IDs to the Omnichannel application and can also be consumed by downstream consuming systems. Functionally, it publishes the current differentiator master data only while the differentiator still resolves to a valid differentiator type.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIFF_IDS, DIFF_TYPE

Additional business tables: none

JSON cache table: MERCHAPI_DIFF_IDS

JSON generation view: V_MERCHAPI_DIFF_IDS_JSON

The cache is refreshed from V_MERCHAPI_DIFF_IDS_JSON, which joins DIFF_IDS to DIFF_TYPE so the payload includes both the differentiator and its type information.

Functionally, the published set is the current differentiator master data that still resolves to a valid differentiator type.

When the omnichannel dimension-type or dimension-value APIs are enabled, the same diff-ID processing flow also repopulates those child caches for impacted items.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_DIFF_IDS_JSON. No additional filter criteria were identified beyond the cache-build logic.

Rebuild marks cache rows deleted when the differentiator no longer exists in V_MERCHAPI_DIFF_IDS_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIFF_IDS, ICL_RMS_DIFF_TYPE (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on DIFF_IDS, DIFF_TYPE.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_DIFF_IDS

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_DIFF_IDS_ADHOC_PROCESS ->
MERCHAPI_OMNI_DIFF_IDS_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/diff

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/diff

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
offsetkey	No	String	Offset key
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-5 PageResultsItemsVMerchapiDiffIds - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-6 VMerchapiDiffIds - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Diff.
diffid	No	String (10)	This column contains the Diff ID.
diffdesc	No	String (120)	This column contains the Diff description.
difftype	No	String (6)	This column contains the Diff type.

Table 5-6 (Cont.) VMerchapiDifflds - Object. See list of elements for detail

Element Name	Required	Data Type	Description
difftypedesc	No	String (120)	This column contains the Diff type description.

Table 5-7 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "diffid": "1930",
      "diffdesc": "Green",
      "difftype": "C",
      "difftypedesc": "Color"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```


Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "action": "INSERT",
      "diffid": "1005",
      "diffdesc": "EIT_Diff_ID : Black",
      "difftype": "CF",
      "difftypedesc": "Cereal Flavour"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/diff?before=2021-05-05T16%3A18%3A57.100893%2B08%3A00&limit=1&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http:// <hostname> /MerchIntegrations/services/foundation/omnichannel/diff?before=2021-05-05T16%3A18%3A57.100893%2B08%3A00&limit=1&offsetkey=1005&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_IDS	Yes	Yes	Yes	Yes
ICL_RMS_DIFF_TYPE	Yes	Yes	Yes	Yes
DIFF_IDS	Yes	No	No	No
MERCHAPI_DIFF_IDS	Yes	Yes	Yes	Yes
V_MERCHAPI_DIFF_IDS_JSON	Yes	No	No	No
V_MERCHAPI_DIFF_IDS	Yes	No	No	No
V_MERCHAPI_ITEM_JSON	Yes	No	No	No
MERCHAPI_ITEM_DIFF_TYPE	Yes	Yes	Yes	Yes
MERCHAPI_ITEM_DIFF_VALUE	Yes	Yes	Yes	Yes

REST Endpoint for Omni channel/Inventory/Store details

Endpoint

MerchIntegrations/services/inventory/omnichannel/inventory/store

Functional Area

Inventory

Business Overview

This service publishes store available inventory to the Omnichannel application and can also be consumed by downstream consuming systems. It includes the store availability position for active, transaction-level inventory items at stockholding stores that are enabled for customer orders, limited to sellable non-pack store rows. The response exposes stock on hand together with the derived available quantity after subtracting reserved and other non-available inventory buckets so consuming systems receive the store availability position used for fulfillment.

The underlying technical component for this service is same as '/inventory/available'. Refer to the API documentation for '/inventory/available' for more details.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/omnichannel/inventory/store

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level. Allowed value is 'STORE'. If not defined, fetches details for all stores.
nodeid	No	String (10)	Node Id. Should be a valid store number.
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-8 PageResultsItemsStoreAvailable - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-8 (Cont.) PageResultsItemsStoreAvailable - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-9 StoreAvailable - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	This field contains the action for Store Inventory.
item	No	String (25)	This field contains the unique alphanumeric value that identifies the item. This will be an approved sellable transaction-level item.
location	No	Number(10,0)	This field contains the numeric identifier of the location in which the item is to be found.
loctype	No	String (1)	This field contains the type of location in the location field. Valid values are S (store).
availablequantity	No	Number(12,4)	This field represents the available inventory of an item in a particular location.
stockonhand	No	Number(12,4)	This field contains current stock on hand for the item.

Table 5-9 (Cont.) StoreAvailable - Object. See list of elements for detail

Element Name	Required	Data Type	Description
standarduom	No	String (4)	This field contains the unit of measure in which stock of the item is tracked at a corporate level.

Table 5-10 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": null,
      "item": "100100092",
      "location": 1521,
      "loctype": "S",
      "availablequantity": 10,
      "stockonhand": 12,
      "standarduom": "EA"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined, then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "action": "INSERT",
      "item": "100000286",
      "location": "1341",
      "loctype": "S",
      "availablequantity": 100,
      "stockonhand": 100,
      "standarduom": "EA"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/inventory/omnichannel/inventory/store?
before=2021-05-13T18%3A28%3A10.129377%2B05%3A30&limit=1&since=1970-01-01T00%3A
00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/inventory/omnichannel/inventory/store?
before=2021-05-13T18%3A28%3A10.129377%2B05%3A30&limit=1&offsetkey=1341%2C10000
0286&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}
```

```
}"]
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
STORE	Yes	No	No	No
MERCHAPI_ITEM_LOC	Yes	No	No	No

REST Endpoint for Omni channel/Inventory/Warehouse details

Endpoint

MerchIntegrations/services/inventory/omnichannel/inventory/warehouse

Functional Area

Inventory

Business Overview

This service publishes warehouse available inventory to the Omnichannel application and can also be consumed by downstream consuming systems. It includes the warehouse availability position for active, transaction-level inventory items at stockholding warehouses that are enabled for customer orders, limited to sellable rows while still allowing warehouse pack items to publish. The response exposes stock on hand, the derived available quantity, and the physical warehouse linkage used for virtual-warehouse sourcing scenarios.

The underlying technical component for this service is same as '/inventory/available'. Refer to the API documentation for '/inventory/available' for more details.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/omnichannel/inventory/warehouse

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level. Allowed value is 'WAREHOUSE'. If not defined, fetches details for all warehouses.
nodeid	No	String (10)	Node Id. Should be a valid warehouse number if nodelevel is 'WAREHOUSE'.
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-11 PageResultsItemsOmnichannel_MerchapiltemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-11 (Cont.) PageResultsItemsOmnichannel_MerchapilitemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-12 Omnichannel_MerchapilitemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	This field contains the action for Warehouse Inventory.
item	No	String (25)	This field contains the unique alphanumeric value that identifies the item. This will be an approved sellable transaction-level item. For warehouse locations, pack items are also included.
location	No	Number(10,0)	This field contains the numeric identifier of the location in which the item is to be found.
loctype	No	String (1)	This field contains the type of location in the location field. Valid values are W (warehouse).
availablequantity	No	Number(12,4)	This field represents the available inventory of an item in a particular location.
stockonhand	No	Number(12,4)	This field contains current stock on hand for the item.

Table 5-12 (Cont.) Omnichannel_MerchapiItemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
standarduom	No	String (4)	This field contains the unit of measure in which stock of the item is tracked at a corporate level.
physicalwarehouse	No	Number(10,0)	This field contains the physical warehouse for location type of warehouse.

Table 5-13 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": null,
      "item": "100100092",
      "location": 11001,
      "loctype": "W",
      "availablequantity": 10,
      "stockonhand": 12,
      "standarduom": "EA",
      "physicalwarehouse": 11
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

```

    ]
  }

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "item": "100000075",
      "location": "10001",
      "loctype": "W",
      "availablequantity": 0,
      "stockonhand": 0,
      "standarduom": "EA",
      "physicalwarehouse": 2
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>:<portnumber>/MerchIntegrations/services/inventory/omnichannel/inventory/warehouse?before=2021-05-13T19%3A14%3A44.470363%2B05%3A30&limit=1&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http://<hostname>:<portnumber>/MerchIntegrations/services/inventory/omnichannel/inventory/warehouse?before=2021-05-13T19%3A14%3A44.470363%2B05%3A30&limit=1&offsetkey=10001%2C1000000075&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}

```

```
}"]
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
WH	Yes	No	No	No
MERCHAPI_ITEM_LOC	Yes	No	No	No

REST Endpoint for Omni channel/ Item details

Endpoint

MerchIntegrations/services/foundation/omnichannel/item

Functional Area

Items - Item Definition

Business Overview

This service publishes the item payload to the Omnichannel application and can also be consumed by downstream consuming systems. It publishes item definitions from approved item records and enriches them with merchandise hierarchy, differentiator, brand, and selected code-detail attributes. When callers supply nodelevel and nodeid, the service returns the item in the context of the requested store or warehouse by joining to MERCHAPI_ITEM_LOC, which adds local item description, local short description, and UIN type for that node. The source view itself publishes only current ITEM_MASTER rows in status A, while refresh processing also selects status D rows so delete actions can still be represented in the cache.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_MASTER

Additional business tables: DEPS, GROUPS, DIVISION, CLASS, SUBCLASS, DIFF_GROUP_HEAD, DIFF_IDS, BRAND, SYSTEM_OPTIONS, CODE_DETAIL

JSON cache table: MERCHAPI_ITEM_MASTER

JSON generation view: V_MERCHAPI_ITEM_JSON

The JSON view is driven from ITEM_MASTER and enriched from hierarchy tables, differentiator setup, brand, system options, and code-detail lookups.

The source view publishes only items whose current ITEM_MASTER.STATUS = 'A', while refresh processing selects ITEM_MASTER rows in status A or D so deletes can still be represented in the cache.

The REST layer also supports location-scoped item output by joining the item cache to MERCHAPI_ITEM_LOC when nodelevel and nodeid are supplied, which adds local item description and UIN context for the requested store or warehouse.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_ITEM_JSON and apply ITEM_MASTER.STATUS IN ('A','D').

ICL (Integration Change Log) table: ICL_RMS_ITEM_MASTER (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on ITEM_MASTER, and the shared item delta flow also refreshes the related UPC and dimension projections maintained under the same process configuration.

Item Approval Handling

Approval of an item also results in publish of foundation/omnichannel/item/upc, foundation/omnichannel/item/dimensiontype, and foundation/omnichannel/item/dimensionvalue through the shared item flow. Approval also queues foundation/omnichannel/item/image, foundation/omnichannel/item/relateditem, foundation/item/location, foundation/item/vat, and foundation/taxLocationItem by writing the existing dependent source rows to those APIs' ICL tables when the item reaches approved status.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_ITEM_DIM_UPC

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_ITEM_DIM_UPC_ADHOC_PROCESS ->
MERCHAPI_OMNI_ITEM_DIM_UPC_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/item

Special handling

Additional GET query parameters: nodelevel, nodeid.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. The available configuration for this service should therefore be reviewed and validated against the customer production data profile, including expected volume, message size, and access pattern, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service shares the API_OMNI_ITEM_DIM_UPC runtime with the omnichannel item, UPC, and dimension caches. Delta processing on that shared flow remains single-threaded, while initial seeding supports multi-threaded execution. Delta commit sizing is controlled through API_COMMIT_MAX_COUNTER, and initial-seeding thread and commit sizing are controlled through API_REFRESH_OVERRIDE_NO_OF_THREADS and API_REFRESH_OVERRIDE_COMMIT_MAX_COUNTER.

For initial seeding, thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the item message from ITEM_MASTER together with hierarchy, differentiator, brand, and lookup data. Common starting points for large-volume seeding are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment. The delivered refresh override is currently 4 threads.

Commit sizing should be reviewed together with the thread setting. Typical values can range from about 1,000 rows, or even higher values such as 10,000. The delivered delta commit setting is currently 1000, and the delivered refresh override commit setting is currently 10000.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution of the shared item cache. Even with refresh-override threading, larger parallel insert volumes can increase index maintenance overhead and slow the load.

To reduce that overhead, DROP_INDEX_ON_REFRESH can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes.

This can improve data-load throughput, but the overall refresh plan should also account for the additional time needed for index rebuild after the seeding step finishes.

GATHER_STATS_ON_REFRESH should normally remain enabled unless testing shows a need to change it.

REST fetch considerations

Item payloads can be comparatively large, so REST page size should be selected based on average message size so that the average response remains below about 10 MB per call. Common page-limit values are 1,000, 5,000, or 10,000 rows.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the LAST_UPDATE_DATETIME index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping BUILD_HISTOGRAM enabled allows the service to support recovery through smaller, more selective polling windows using the optimalBefore query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides optimalBefore and leaves before empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups LAST_UPDATE_DATETIME values into buckets of 10,000 rows in MERCHAPI_TIMESTAMP_HISTOGRAM using the shared HISTOGRAM_ROW_COUNT configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level (STORE or WAREHOUSE)
nodeid	No	String (10)	Node Id
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-14 PageResultsItemsVMerchApiItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-14 (Cont.) PageResultsItemsVMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-15 VMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for item details.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.
itemparent	No	String (25)	This field specifies the alphanumeric value that uniquely identifies the item or group at the level above the item.
itemgrandparent	No	String (25)	This field specifies the alphanumeric value that uniquely identifies the item or group two levels above the item.
itemlevel	No	Number(1,0)	This field specifies the number indicating which of the three levels the item resides. The item level determines if the item stands alone or if it is part of a family of related items.

Table 5-15 (Cont.) VMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
tranlevel	No	Number(1,0)	This field specifies the number indicating which of the three levels transactions occur for the item's group. The transaction level is the level at which the item's inventory is tracked in the system.
inventoryind	No	String (1)	This field determines if an item holds inventory or not for item transformations.
diff1level	No	String (5)	This field specifies diff1 level.
diff1type	No	String (6)	This field specifies diff1 type.
diff1	No	String (10)	This field specifies the diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group or value. For an item that is not a parent, this field may contain a value.
diff2level	No	String (5)	This field specifies diff2 level.
diff2type	No	String (6)	This field specifies diff2 type.
diff2	No	String (10)	This field specifies the diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group or value. For an item that is not a parent, this field may contain a value.
diff3level	No	String (5)	This field specifies diff3 level.
diff3type	No	String (6)	This field specifies diff3 type.

Table 5-15 (Cont.) VMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff3	No	String (10)	This field specifies the diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group or value. For an item that is not a parent, this field may contain a value.
division	No	Number(4,0)	This field specifies the number which uniquely identifies the division of the company.
groupid	No	Number(4,0)	This field specifies the group in which the department exists.
dept	No	Number(4,0)	This field uniquely identifies the department.
class	No	Number(4,0)	This field uniquely identifies the class within the system.
uniqueclass	No	Number(10,0)	This field uniquely identifies the department/class combination.
subclass	No	Number(4,0)	This field uniquely identifies the subclass.
uniquesubclass	No	Number(10,0)	This field uniquely identifies the department/class/subclass combination.
description	No	String (250)	This field specifies the description of the item. For items that have parents, this description will default to the parent's description plus any differentiators.
shortdescription	No	String (120)	This field specifies the short description of the item.
brandname	No	String (30)	This field specifies the brand associated with an item.

Table 5-15 (Cont.) VMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
merchandiseind	No	String (1)	This field indicates if the item is a merchandise item. Merchandise items are generally physical items that must be shipped or received and for which there is inventory. Non-merchandise items are often items which do not have inventory. Common examples include extra fees for service, extended warranties, alterations, or endlessly available items such as downloads, in-app purchases of bonus content, and subscriptions.
mfgrecretail	No	Number(20,4)	This field specifies the manufacturer's recommended retail price for the item. This field is stored in the primary currency.
orginalunitretail	No	Number(20,4)	This field specifies the retail price for the item.
catchweightind	No	String (1)	This field specifies whether the item should be weighed when it arrives at a location.
itemservicelevel	No	String (6)	This field specifies the value that restricts the type of shipment methods that RCOM can select for an item.
giftwrapind	No	String (1)	This field determines whether the location will support gift wrapping or not.
shipaloneind	No	String (1)	This field indicates if the item should be shipped to the customer in a separate package versus being grouped together in a box.

Table 5-15 (Cont.) VMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
standarduom	No	String (4)	This field specifies the unit of measure in which stock of the item is tracked at a corporate level.
productclassification	No	String (6)	This field determines how to pack customer orders, such as to determine products that may not be able to be packaged together. Examples of classifications may include fragile, toxic, and heavy.
xdiff1desc	No	String (120)	This field specifies diff1 description.
xdiff2desc	No	String (120)	This field specifies diff2 description.
xdiff3desc	No	String (120)	This field specifies diff3 description.
itemnumbertype	No	String (6)	This field specifies what type the item is. Valid values for this field are in the code type UPCT on the code_head and code_detail tables.
itemlocation	No	String (10)	This field specifies if the service call was to fetch the enterprise view or a specific location. The value is either 'ent' or the node ID parameter.
localdescription	No	String (250)	This field specifies the local description of the item. This is applicable when the service query parameter of node level and node ID is used to fetch data for a specific location.
localshortdescription	No	String (120)	This field specifies the local short description of the item. This is applicable when the service query parameter of node level and node ID is used to fetch data for a specific location.

Table 5-15 (Cont.) VMerchApiltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uintype	No	String (6)	This field specifies the unique identification number (UIN) type used to identify the instances of the item at the location. This is applicable when the service query parameter of node level and node ID is used to fetch data for a specific location.

Table 5-16 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "item": "100100076",
      "itemparent": null,
      "itemgrandparent": null,
      "itemlevel": 1,
      "tranlevel": 1,
      "inventoryind": "Y",
      "diff1level": null,
      "diff1type": null,
      "diff1": null,
      "diff2level": null,
      "diff2type": null,
      "diff2": null,
      "diff3level": null,
      "diff3type": null,
      "diff3": null,
      "division": 1,
      "groupid": 1,
      "dept": 3041,

```

```

"class": 1,
"uniqueclass": 1,
"subclass": 1,
"uniquesubclass": 1,
"description": "Regular Item 100100076",
"shortdescription": "100100076 Item",
"brandname": null,
"merchandiseind": "Y",
"mfgrecretail": 12.99,
"orginalunitretail": 12.99,
"catchweightind": "Y",
"itemservicelevel": null,
"giftwrapind": "Y",
"shipaloneind": "Y",
"standarduom": "EA",
"productclassification": null,
"xdiff1desc": null,
"xdiff2desc": null,
"xdiff3desc": null,
"itemnumbertype": "MANL",
"itemlocation": "ent",
"localdescription": "localItemDesc",
"localshortdescription": "localShortDesc",
"uintype": null
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.

Elements	Description
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Input

```
{
  "items": [
    {
      "item": "100000024",
      "action": "INSERT",
      "itemlocation": "ent",
      "itemparent": null,
      "itemgrandparent": null,
      "itemlevel": 1,
      "tranlevel": 3,
      "inventoryind": "Y",
      "diff1level": null,
      "diff1type": null,
      "diff1": null,
      "diff2level": null,
      "diff2type": null,
      "diff2": null,
      "diff3level": null,
      "diff3type": null,
      "diff3": null,
      "division": 1000,
      "groupid": 108,
      "dept": 1117,
      "class": 1,
      "uniqueclass": 51,
      "subclass": 1,
      "uniquesubclass": 432,
      "description": "Test Item 100000024 ",
      "shortdescription": "Test Item",
      "localdescription": null,
      "localshortdescription": null,
      "brandname": null,
      "merchandiseind": "Y",
      "uintype": null,
      "mfgrecretail": null,
      "orginalunitretail": 13.2,
      "catchweightind": "N",
      "itemservicelevel": null,
      "giftwrapind": "N",
      "shipaloneind": "N",
    }
  ]
}
```

```

        "standarduom": "EA",
        "productclassification": "APP",
        "xdiff1desc": null,
        "xdiff2desc": null,
        "xdiff3desc": null,
        "itemnumbertype": "MANL" }
    ],
    "hasMore": true,
    "limit": 1,
    "count": 1,
    "links": [
        {
            "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/item?
before=2021-05-12T23%3A35%3A49.770127%2B05%3A30&nodelevel=COMPANY&limit=1&node
id=1&since=1970-01-01T00%3A00%3A00.001Z",
            "rel": "self"
        },
        {
            "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/item?
before=2021-05-12T23%3A35%3A49.770127%2B05%3A30&nodelevel=COMPANY&limit=1&node
id=1&offsetkey=100000024&since=1970-01-01T00%3A00%3A00.001Z ",
            "rel": "next"
        }
    ]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ICL_RMS_ITEM_MASTER	Yes	No	Yes	Yes
MERCHAPI_ITEM_MASTER	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_MASTER	Yes	No	No	No
ICL_RMS_ITEM_IMAGE	No	Yes	No	No
ICL_RMS_RELATED_ITEM	No	Yes	No	No
ICL_RMS_ITEM_LOC	No	Yes	No	No
SYSTEM_OPTIONS	Yes	No	No	No

REST Endpoint for Omni channel/ Item UPC details

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/upc

Functional Area

Items - Item Definition

Business Overview

This service publishes reference-item and UPC-style rows to the Omnichannel application and can also be consumed by downstream consuming systems. It uses the same underlying component as 'foundation/omnichannel/item'. Functionally, the enterprise query returns only below-transaction-level items whose parent item is present in the published item cache. When callers supply nodelevel and nodeid, the service further requires a matching MERCHAPI_ITEM_LOC row for that parent item at the requested store or warehouse, so the reference item is returned only when the parent is published for that node.

Refer to the documentation for 'foundation/omnichannel/item' for technical details and functional considerations.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/upc

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level (STORE or WAREHOUSE)
nodeid	No	String (10)	Node Id
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-17 PageResultsItemsVMerchApiItemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-18 VMerchApiItemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for item details.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.

Table 5-18 (Cont.) VMerchApiItemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemparent	No	String (25)	This field specifies the alphanumeric value that uniquely identifies the item or group at the level above the item.
itemgrandparent	No	String (25)	This field specifies the alphanumeric value that uniquely identifies the item or group two levels above the item.
itemlevel	No	Number(1,0)	This field specifies the number indicating which of the three levels the item resides. The item level determines if the item stands alone or if it is part of a family of related items.
tranlevel	No	Number(1,0)	This field specifies the number indicating which of the three levels transactions occur for the item's group. The transaction level is the level at which the item's inventory is tracked in the system.
inventoryind	No	String (1)	This field determines if an item holds inventory or not for item transformations.
diff1level	No	String (5)	This field specifies diff1 level.
diff1type	No	String (6)	This field specifies diff1 type.
diff1	No	String (10)	This field specifies the diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group or value. For an item that is not a parent, this field may contain a value.
diff2level	No	String (5)	This field specifies diff2 level.

Table 5-18 (Cont.) VMerchApiltemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff2type	No	String (6)	This field specifies diff2 type.
diff2	No	String (10)	This field specifies the diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group or value. For an item that is not a parent, this field may contain a value.
diff3level	No	String (5)	This field specifies diff3 level.
diff3type	No	String (6)	This field specifies diff3 type.
diff3	No	String (10)	This field specifies the diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group or value. For an item that is not a parent, this field may contain a value.
division	No	Number(4,0)	This field specifies the number which uniquely identifies the division of the company.
groupid	No	Number(4,0)	This field specifies the group in which the department exists.
dept	No	Number(4,0)	This field uniquely identifies the department.
class	No	Number(4,0)	This field uniquely identifies the class within the system.
uniqueclass	No	Number(10,0)	This field uniquely identifies the department/class combination.
subclass	No	Number(4,0)	This field uniquely identifies the subclass.

Table 5-18 (Cont.) VMerchApitemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uniquesubclass	No	Number(10,0)	This field uniquely identifies the department/class/subclass combination.
description	No	String (250)	This field specifies the description of the item. For items that have parents, this description will default to the parent's description plus any differentiators.
shortdescription	No	String (120)	This field specifies the short description of the item.
brandname	No	String (30)	This field specifies the brand associated with an item.
merchandiseind	No	String (1)	This field indicates if the item is a merchandise item. Merchandise items are generally physical items that must be shipped or received and for which there is inventory. Non-merchandise items are often items which do not have inventory. Common examples include extra fees for service, extended warranties, alterations, or endlessly available items such as downloads, in-app purchases of bonus content, and subscriptions.
mfgrecretail	No	Number(20,4)	This field specifies the manufacturer's recommended retail price for the item. This field is stored in the primary currency.
orginalunitretail	No	Number(20,4)	This field specifies the retail price for the item.

Table 5-18 (Cont.) VMerchApiltemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
catchweightind	No	String (1)	This field specifies whether the item should be weighed when it arrives at a location.
itemservicelevel	No	String (6)	This field specifies the value that restricts the type of shipment methods that RCOM can select for an item.
giftwrapind	No	String (1)	This field determines whether the location will support gift wrapping or not.
shipaloneind	No	String (1)	This field indicates if the item should be shipped to the customer in a separate package versus being grouped together in a box.
standarduom	No	String (4)	This field specifies the unit of measure in which stock of the item is tracked at a corporate level.
productclassification	No	String (6)	This field determines how to pack customer orders, such as to determine products that may not be able to be packaged together. Examples of classifications may include fragile, toxic, and heavy.
xdiff1desc	No	String (120)	This field specifies diff1 description.
xdiff2desc	No	String (120)	This field specifies diff2 description.
xdiff3desc	No	String (120)	This field specifies diff3 description.
itemnumbertype	No	String (6)	This field specifies what type the item is. Valid values for this field are in the code type UPCT on the code_head and code_detail tables.

Table 5-18 (Cont.) VMerchApiItemUPC - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemlocation	No	String (10)	This field specifies if the service call was to fetch the enterprise view or a specific location. The value is either 'ent' or the node ID parameter.
localdescription	No	String (250)	This field specifies the local description of the item. This is applicable when the service query parameter of node level and node ID is used to fetch data for a specific location.
localshortdescription	No	String (120)	This field specifies the local short description of the item. This is applicable when the service query parameter of node level and node ID is used to fetch data for a specific location.
uintype	No	String (6)	This field specifies the unique identification number (UIN) type used to identify the instances of the item at the location. This is applicable when the service query parameter of node level and node ID is used to fetch data for a specific location.

Table 5-19 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```

{
  "items": [
    {
      "action": "INSERT",
      "item": "100100076",
      "itemparent": null,
      "itemgrandparent": null,
      "itemlevel": 1,
      "tranlevel": 1,
      "inventoryind": "Y",
      "diff1level": null,
      "diff1type": null,
      "diff1": null,
      "diff2level": null,
      "diff2type": null,
      "diff2": null,
      "diff3level": null,
      "diff3type": null,
      "diff3": null,
      "division": 1,
      "groupid": 1,
      "dept": 3041,
      "class": 1,
      "uniqueclass": 1,
      "subclass": 1,
      "uniquesubclass": 1,
      "description": "Regular Item 100100076",
      "shortdescription": "100100076 Item",
      "brandname": null,
      "merchandiseind": "Y",
      "mfgrecretail": 12.99,
      "orginalunitretail": 12.99,
      "catchweightind": "Y",
      "itemservicelevel": null,
      "giftwrapind": "Y",
      "shipaloneind": "Y",
      "standarduom": "EA",
      "productclassification": null,
      "xdiff1desc": null,
      "xdiff2desc": null,
      "xdiff3desc": null,
      "itemnumbertype": "MANL",
      "itemlocation": "ent",
      "localdescription": "localItemDesc",
      "localshortdescription": "localShortDesc",
      "uintype": null
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {

```



```

"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "item": "101100001",
      "action": "INSERT",
      "itemlocation": "ent",
      "itemparent": "100250007",
      "itemgrandparent": null,
      "itemlevel": 2,
      "tranlevel": 1,
      "inventoryind": "Y",
      "diff1level": null,
      "diff1type": null,
      "diff1": null,
      "diff2level": null,
      "diff2type": null,
      "diff2": null,
      "diff3level": null,
      "diff3type": null,
      "diff3": null,
      "division": 3000,
      "groupid": 304,
      "dept": 3041,
    }
  ]
}

```

```

        "class": 1,
        "uniqueclass": 139,
        "subclass": 1,
        "uniquesubclass": 17,
        "description": "Something - MK",
        "shortdescription": "Something",
        "localdescription": null,
        "merchandiseind": "Y",
        "uintype": null,
        "mfgrecretail": null,
        "orginalunitretail": null,
        "catchweightind": "N",
        "itemservicelevel": null,
        "giftwrapind": "N",
        "shipaloneind": "N",
        "standarduom": "EA",
        "productclassification": null,
        "xdiff1desc": null,
        "xdiff2desc": null,
        "xdiff3desc": null,
        "itemnumbertype": "UPC-A"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/item/upc?
before=2021-05-13T16%3A38%3A07.284763%2B05%3A30&nodelevel=COMPANY&limit=1&node
id=1&since=1970-01-01T00%3A00%3A00.001Z ",
      "rel": "self"
    },
    {
      "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/item/upc?
before=2021-05-13T16%3A38%3A07.284763%2B05%3A30&nodelevel=COMPANY&limit=1&node
id=1&offsetkey=101100001&since=1970-01-01T00%3A00%3A00.001Z ",
      "rel": "next"
    }
  ]
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_MASTER	Yes	No	No	No
ICL_RMS_ITEM_MASTER	Yes	No	Yes	Yes
MERCHAPI_ITEM_MASTER	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_MASTER	Yes	No	No	No
ICL_RMS_ITEM_IMAGE	No	Yes	No	No
ICL_RMS_RELATED_ITEM	No	Yes	No	No
ICL_RMS_ITEM_LOC	No	Yes	No	No
SYSTEM_OPTIONS	Yes	No	No	No

REST Endpoint for Omni channel/ Merchandise Hierarchy details

Endpoint

MerchIntegrations/services/foundation/omnichannel/merchhier

Functional Area

Merchandise Hierarchy

Business Overview

This service retrieves the merchandise hierarchy relationships. No details about each hierarchy value are provided, only the hierarchy tree relationships are returned. In order to fetch new updates pass the parameter of since/before timestamp.

This service publishes the merchandise hierarchy to the Omnichannel application and can also be consumed by downstream consuming systems. It publishes the current company-through-subclass hierarchy tree with parent-child links, hierarchy display identifiers, and buyer assignments where they are maintained in Merchandising. This service does not provide details about individual hierarchy value, only the hierarchy tree relationships are returned.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, CODE_DETAIL

JSON cache table: MERCHAPI_MERCH_HIER

JSON generation view: V_MERCHAPI_MERCH_HIER_JSON

The source view builds the company-through-subclass hierarchy from COMPHEAD, DIVISION, GROUPS, DEPS, CLASS, SUBCLASS, and CODE_DETAIL, and it uses CODE_DETAIL (CODE_TYPE = 'MHSO') to label and sort the hierarchy levels.

Functionally, the service publishes the current hierarchy tree with parent-child links, display identifiers, and buyer assignments where they are maintained, and it seeds the company row when it is missing so the tree always has a company root.

The cache is refreshed from that current-state hierarchy source so the published tree stays aligned with the latest company-through-subclass structure in Merchandising.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_MERCH_HIER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Rebuild marks cache rows deleted when the hierarchy row no longer exists in V_MERCHAPI_MERCH_HIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

Delta processing is driven by ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, and ICL_RMS_SUBCLASS.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_MERCH_HIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_OMNI_MERCH_HIER_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/merchhier

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/merchhier

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-20 PageResultsItemsVMerchapiMerchHier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-21 VMerchapiMerchHier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for merchandising hierarchy.
nodeid	No	Number(10,0)	This column contains the Merchandising Hierarchy ID.
hierarchylevel	No	String (10)	This column contains the Merchandising Hierarchy Level.

Table 5-21 (Cont.) VMerchapiMerchHier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nodename	No	String (120)	This column contains the name of the Merchandising Hierarchy.
parentlevel	No	String (10)	This column contains the Parent Merchandising Hierarchy Level.
parentnodeid	No	Number(10,0)	This column contains the Parent Merchandising Hierarchy ID.
merchdisplayid	No	Number(10,0)	This column contains the merchandising hierarchy display ID.
buyer	No	Number(4,0)	This column contains the number of the buyer associated with the merchandising hierarchy.

Table 5-22 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "nodeid": 1,
      "hierarchylevel": "COMPANY",
      "nodename": "Retailers Ltd",
      "parentlevel": "COMPANY",
      "parentnodeid": 1,
      "merchdisplayid": 1,
      "buyer": 1002
    }
  ],
}
```

```

"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "UPDATE",
      "nodeid": 1,
      "hierarchylevel": "COMPANY",
      "nodename": "Retailers Ltd",
      "parentlevel": null,
      "parentnodeid": null,
      "merchdisplayid": 1,
      "buyer": null
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {

```

```

        "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/merchhier?
before=2021-05-12T20%3A43%3A04.296066%2B05%3A30&limit=1&since=1970-01-01T00%3A
00%3A00.001Z",
        "rel": "self"
    },
    {
        "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/location/warehouse?
before=2021-05-12T20%3A15%3A13.253929%2B05%3A30&limit=1&offsetkey=2%2Cvirtual&
since=1970-01-01T00%3A00%3A00.001Z ",
        "rel": "next"
    }
]
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_MERCH_HIER	Yes	Yes	Yes	Yes
V_MERCHAPI_MERCH_HIER	Yes	No	No	No
V_MERCHAPI_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

REST Endpoint for Omni channel/Org hier descendant's details

Endpoint

MerchIntegrations/services/foundation/omnichannel/orghier/descendant

Functional Area

Organizational Hierarchy

Business Overview

This service retrieves organization hierarchy relationships for the levels below the provided hierarchy value (including warehouses). For example, if a region is sent in as the input information, the service will return all of the districts and stores in the provided region. No details about each hierarchy value are provided, only the hierarchy tree relationships are returned. In order to fetch new updates pass the parameter of since/before timestamp.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: CHAIN, AREA, REGION, DISTRICT, STORE, WH, ADDR

Additional business tables: COMPHEAD, CODE_DETAIL

JSON cache table: MERCHAPI_ORG_HIER (shared with foundation/omnichannel/orghier)

JSON generation view: V_ORG_HIER_FULL_VW (shared cache source used by foundation/omnichannel/orghier)

This service is the descendant projection over the same organizational hierarchy cache table, MERCHAPI_ORG_HIER.

It uses the same underlying hierarchy data built from V_ORG_HIER_FULL_VW, but functionally it returns the current non-deleted descendants beneath a supplied organization node rather than the broader change-tracked relationship feed.

Because it is a child merchapi of foundation/omnichannel/orghier, it does not have its own separate cache-population process.

Initial Seeding and post enablement rebuild/repair

This service relies on the shared initial-seeding and rebuild flow used by foundation/omnichannel/orghier.

That shared rebuild uses the all-level org-hierarchy refresh path.

ICL (Integration Change Log) table: ICL_RMS_CHAIN, ICL_RMS_AREA, ICL_RMS_REGION, ICL_RMS_DISTRICT, ICL_RMS_STORE, ICL_RMS_WH, ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

This child service relies on the shared org-hierarchy delta flow, with address deltas scoped to modules WH, ST, and WFST.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_ORG_HIER_STR_WH

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_ORG_HIER_STR_WH_ADHOC_PROCESS ->
MERCHAPI_OMNI_ORG_HIER_STR_WH_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/orghier/descendant

Special handling

Additional GET query parameters: nodelevel, nodeid.

Supported nodelevel values for this service are COMPANY, CHAIN, AREA, REGION, DISTRICT.

Foundation/omnichannel/orghier/descendant is a child merchapi of foundation/omnichannel/orghier.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/orghier/descendant

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-23 PageResultsItemsVMerchApiOrgHier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-23 (Cont.) PageResultsItemsVMerchApiOrgHier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-24 VMerchApiOrgHier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchylevel	No	String (10)	This field specifies Organization Hierarchy Level.
hierarchynodeid	No	Number(10,0)	This field specifies Organization Hierarchy ID.
hierarchynodename	No	String (120)	This field specifies the name of the Organization Hierarchy ID.
parentlevel	No	String (10)	This field specifies the Parent Organization Hierarchy Level.
parentid	No	Number(10,0)	This field specifies Parent Organization Hierarchy ID.
levelorder	No	Number(4,0)	This field specifies the hierarchy level order.

Table 5-25 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-25 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "hierarchylevel": "CHAIN",
      "hierarchynodeid": 100,
      "hierarchynodename": null,
      "parentlevel": null,
      "parentid": 1,
      "levelorder": 10
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.

Elements	Description
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "hierarchylevel": "COMPANY",
      "hierarchynodeid": 9118,
      "hierarchynodename": "Retailers Ltd",
      "parentlevel": null,
      "parentid": null,
      "levelorder": 1
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/orghier/descendant?nodelevel=COMPANY&limit=1&nodeid=9118",
      "rel": "self"
    },
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/orghier/descendant?
nodelevel=COMPANY&limit=1&nodeid=9118&offsetkey=1%2C9118",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
V_ICL_RMS_ORGHIER	Yes	No	No	No
ICL_RMS_CHAIN	Yes	Yes	Yes	Yes
ICL_RMS_AREA	Yes	Yes	Yes	Yes
ICL_RMS_REGION	Yes	Yes	Yes	Yes
ICL_RMS_DISTRICT	Yes	Yes	Yes	Yes
ITEM_LOC	Yes	No	No	No
MERCHAPI_ORG_HIER	Yes	Yes	Yes	Yes
V_ORG_HIER_FULL_VW	Yes	No	No	No
V_MERCHAPI_ORG_HIER	Yes	No	No	No

REST Endpoint for Omni channel/Org hierarchy details

Endpoint

MerchIntegrations/services/foundation/omnichannel/orghier

Functional Area

Organizational Hierarchy

Business Overview

This service publishes organizational hierarchy relationships to downstream consuming systems. It expands the company, chain, area, region, district, store, and warehouse node set into hierarchy relationship rows so consuming systems can retrieve descendants beneath a selected parent node while each row still carries its immediate parent information.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: CHAIN, AREA, REGION, DISTRICT, STORE, WH, ADDR

Additional business tables: COMPHEAD, CODE_DETAIL

JSON cache table: MERCHAPI_ORG_HIER

JSON generation view: not applicable; source hierarchy view is V_ORG_HIER_FULL_VW

That source view builds company, chain, area, region, district, store, and warehouse nodes from COMPHEAD, CHAIN, AREA, REGION, DISTRICT, STORE, WH, and CODE_DETAIL.

Functionally, V_MERCHAPI_ORG_HIER expands those nodes into ancestor-to-descendant relationship rows, so consumers can retrieve the hierarchy beneath a selected parent node while each row still carries its immediate parent information.

The shared package processes higher organization levels directly and then invokes the retail-store and warehouse loaders so the broader org-hierarchy, store, and warehouse caches stay aligned.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_ORG_HIER_FULL_VW. No additional filter criteria were identified beyond the cache-build logic.

Foundation/omnichannel/orghier rebuilds all hierarchy levels.

ICL (Integration Change Log) table: ICL_RMS_CHAIN, ICL_RMS_AREA, ICL_RMS_REGION, ICL_RMS_DISTRICT, ICL_RMS_STORE, ICL_RMS_WH, ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

Delta processing uses ICL_RMS_CHAIN, ICL_RMS_AREA, ICL_RMS_REGION, ICL_RMS_DISTRICT, ICL_RMS_STORE, ICL_RMS_WH, and ICL_RMS_ADDR, with address deltas scoped to modules WH, ST, and WFST.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_ORG_HIER_STR_WH

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_ORG_HIER_STR_WH_ADHOC_PROCESS ->
MERCHAPI_OMNI_ORG_HIER_STR_WH_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/orghier

Special handling

Additional GET query parameters: nodelevel, nodeid.

Supported nodelevel values for this service are COMPANY, CHAIN, AREA, REGION, DISTRICT, STORE.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/orghier

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)

Parameter Name	Required	Data Type	Description
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-26 PageResultsItemsVMerchApiOrgHierDetail - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-27 VMerchApiOrgHierDetail - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for organization hierarchy detail.

Table 5-27 (Cont.) VMerchApiOrgHierDetail - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchylevel	No	String (10)	This field specifies Organization Hierarchy Level.
hierarchynodeid	No	Number(10,0)	This field specifies Organization Hierarchy ID.
hierarchynodename	No	String (120)	This field specifies the name of the Organization Hierarchy ID.
parentlevel	No	String (10)	This field specifies the Parent Organization Hierarchy Level.
parentid	No	Number(10,0)	This field specifies Parent Organization Hierarchy ID.
mgrname	No	String (120)	This field specifies the name of the store manager.
stockholding	No	String (1)	This field specifies whether the store can hold stock. This field holds value only for stores.
integratedpos	No	String (1)	This field indicates whether or not the POS system at the store is integrated. This field holds value only for stores.

Table 5-28 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
```

```
{
  "action": null,
  "hierarchylevel": "CHAIN",
  "hierarchynodeid": 100,
  "hierarchynodename": null,
  "parentlevel": null,
  "parentid": 1,
  "mgrname": null,
  "stockholding": null,
  "integratedpos": null
},
{
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than limit defined then hasMore will be set to true else false.
Limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
Links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "action": "UPDATE",
  "hierarchylevel": "AREA",
  "hierarchynodeid": 1212,
  "hierarchynodename": "Area 12121",
  "parentlevel": "CHAIN",
```

```

        "parentid": 1212,
        "mgrname": "V",
        "stockholding": null,
        "integratedpos": null
    }
],
"hasMore": true,
"limit": 1,
"count": 1,
"links": [
    {
        "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/orghier?
before=2021-05-11T15%3A20%3A25.711306%2B00%3A00&nodelevel=COMPANY&limit=1&node
id=9118&since=1970-01-01T00%3A00%3A00.001Z",
        "rel": "self"
    },
    {
        "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/orghier?
before=2021-05-11T15%3A20%3A25.711306%2B00%3A00&nodelevel=COMPANY&limit=1&node
id=9118&offsetkey=AREA%2C1212&since=1970-01-01T00%3A00%3A00.001Z",
        "rel": "next"
    }
]
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
V_ICL_RMS_ORGHIER	Yes	No	No	No
ICL_RMS_CHAIN	Yes	Yes	Yes	Yes
ICL_RMS_AREA	Yes	Yes	Yes	Yes
ICL_RMS_REGION	Yes	Yes	Yes	Yes
ICL_RMS_DISTRICT	Yes	Yes	Yes	Yes
ITEM_LOC	Yes	No	No	No
MERCHAPI_ORG_HIER	Yes	Yes	Yes	Yes
V_ORG_HIER_FULL_VW	Yes	No	No	No
V_MERCHAPI_ORG_HIER	Yes	No	No	No

REST Endpoint for Omni channel/refresh Date

Endpoint

MerchIntegrations/services /omnichannel/refreshdate

Functional Area

Administration - Operations

Business Overview

This is a metadata publish service that exposes the list of enabled integration services together with cache-refresh and latest-change information for each service. It is intended as a lightweight pre-check for clients that call multiple integration APIs as part of the same batch or polling process. By calling this service first, a client can identify which APIs have had a real data change and avoid redundant requests to services whose published data has not changed.

Cache refresh timestamp: The timestamp when the JSON cache for a service was initially loaded or most recently fully refreshed. This helps determine when a cache was last rebuilt and whether a downstream consumer should perform a bulk reload or continue with delta retrieval. This information is sourced from MERCHAPI_CONFIG.

Latest data update timestamp: The most recent data-update timestamp for the service. This helps reduce unnecessary calls across multiple integration services by allowing a client to identify, in a single request, which services have had recent data changes. This information is derived by querying the applicable underlying table for each individual service.

Implementation usage: Where a client batch process calls multiple publish APIs in sequence, this service can be used as a gating step to determine which APIs need to be called in that cycle and which can be skipped because no new published data is available.

XStore usage: XStore relies extensively on this service to keep integration calls from XOCS to MFCS relevant by helping avoid unnecessary requests to APIs whose published data has not changed.

Performance note: This service relies on the LAST_UPDATE_DATETIME index being available on each JSON cache table. If any API is running initial seeding with the drop-index option enabled and the index rebuild has not yet completed, performance of this service for that table can degrade.

Service Type

GET

ReST URL

MerchIntegrations/services/omnichannel/refreshdate

Input Parameters

Parameter Name	Required	Data Type	Description
entitytype	No	String	entity type
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-29 PageResultsItemsVMerchApiRefreshDate - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-30 VMerchApiRefreshDate - Object. See list of elements for detail

Element Name	Required	Data Type	Description
entity	No	String (100)	This field gives the service name which is defined in merchapi_config table.

Table 5-30 (Cont.) VMerchApiRefreshDate - Object. See list of elements for detail

Element Name	Required	Data Type	Description
refreshdate	No	dateTime	This field gives the refresh date which is the time when the initial loading of the service completes and the API is enabled in merchapi_config table. This field can be used to identify if a bulk process (api enablement or refresh with the option truncate and load) was done for the api. A change of refresh date indicates that the entire dataset is available for subscribe for the publish API. Note: XOCS heavily relies on this field during every 30 min polling request to identify if the fetch for an API will be delta increment or a full download for an API and accordingly handle the call differently.
latestupdatetime	No	dateTime	This field provides the most recent update timestamp for the dataset applicable for the API. The purpose of this field is to provide additional visibility of the data changes across all the API in a single place to have the client calling the differnet publish API at regular frequency to optimize the number of service calls.

Table 5-31 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-31 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "entity": "ItemImage",
      "refreshdate": "2001-12-31T23:59:59.000Z",
      "latestupdate": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.

Elements	Description
links	“self” : This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "entity": "COFutureAvail",
      "refreshdate": "2020-12-17T10:03:12.324664000-06:00",
      "latestupdatetime": "2022-04-25T11:26:01.714875000+00:00"
    }
  ],
  "hasMore": false,
  "limit": 10000,
  "count": 15,
  "links": [
    {
      "href": "http://<hostname> /MerchIntegrations/services/omnichannel/refreshdate?before=2021-05-06T21%3A00%3A34.238393%2B08%3A00&limit=10000&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "self"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
V_CACHE_REFRESH_DATE	Yes	No	No	No

REST Endpoint for Omni channel/Warehouse details

Endpoint

MerchIntegrations/services/foundation/omnichannel/location/warehouse

Functional Area

Organizational Hierarchy - Warehouses

Business Overview

This service publishes warehouse details to the Omnichannel application and can also be consumed by downstream consuming systems. It publishes warehouse records with the primary warehouse address maintained in Merchandising and derives the published warehouse type as physical or virtual from the warehouse relationship data.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: WH, ADDR

Additional business tables: CHANNELS, ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, COUNTRY

JSON cache table: MERCHAPI_WH

JSON generation view: V_MERCHAPI_WH_JSON

The JSON view is built from WH, CHANNELS, ADDR, ADD_TYPE_MODULE, STATE, COUNTRY, and V_ADD_TYPE_TL.

Functionally, it publishes warehouse records with their primary warehouse address (ADDR_TYPE = 1 and PRIMARY_ADDR_IND = 'Y') and derives the published warehouse type as physical or virtual from the PHYSICAL_WH relationship.

The warehouse cache shares its loader with the omnichannel organization hierarchy and retail-store caches through the shared API_OMNI_ORG_HIER_STR_WH process.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_WH_JSON and publish warehouse records with the primary warehouse address where ADDR_TYPE = 1 and PRIMARY_ADDR_IND = 'Y'.

ICL (Integration Change Log) table: ICL_RMS_WH, ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

Warehouse changes come from ICL_RMS_WH, and address-only changes come from ICL_RMS_ADDR for module WH.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_ORG_HIER_STR_WH

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_ORG_HIER_STR_WH_ADHOC_PROCESS ->
MERCHAPI_OMNI_ORG_HIER_STR_WH_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/location/warehouse

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/location/warehouse

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-32 PageResultsItemsVMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-32 (Cont.) PageResultsItemsVMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-33 VMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for warehouse details.
warehouseid	No	Number(10,0)	This field uniquely identifies the warehouse.
type	No	String (8)	This field determines which warehouses are physical and which are virtual.
warehousename	No	String (150)	This field specifies the name of the warehouse which, along with the warehouse number, identifies the warehouse.
whnamessecondary	No	String (150)	This field specifies the secondary name of the warehouse.
contactname	No	String (120)	This field specifies the name of the contact at this location.
phonenumber	No	String (20)	This field specifies the phone number of the contact person at this location.
faxnumber	No	String (20)	This field specifies the fax number of the contact person at this location.
email	No	String (100)	This field specifies the email of the contact person at this location.

Table 5-33 (Cont.) VMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencycode	No	String (3)	This field specifies the currency code under which the warehouse operates.
physicalwh	No	Number(10,0)	This field specifies the number of the physical warehouse that is assigned to the virtual warehouse.
vatregion	No	Number(4,0)	This field specifies the unique identifying number for the VAT region where this warehouse is located.
orghiertype	No	Number(4,0)	This field specifies the organization type that will be used for reporting purposes for the warehouse. The type comes from the organizational hierarchy.
orghiervalue	No	Number(10,0)	This field specifies the code associated with the specific organizational hierarchy type.
orgentitytype	No	String (1)	This field specifies if the warehouse is a legal entity (importer, exporter) or a regular warehouse.
channelid	No	Number(4,0)	This field specifies the channel for which the virtual warehouse will be assigned.
channelname	No	String (120)	This field specifies the name of the channel for which the virtual warehouse will be assigned.
address1	No	String (240)	This field specifies the first line of the address.
address2	No	String (240)	This field specifies the second line of the address.
address3	No	String (240)	This field specifies the third line of the address.

Table 5-33 (Cont.) VMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
city	No	String (120)	This field specifies the name of the city that is associated with the address.
state	No	String (3)	This field specifies the state abbreviation for the address.
country	No	String (3)	This field specifies the country where the address exists.
postalcode	No	String (30)	This field specifies the zip code for the address.

Table 5-34 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": null,
      "warehouseid": 6000,
      "type": "physical",
      "warehousename": null,
      "whnamessecondary": null,
      "contactname": "Laura Johnson",
      "phonenumber": "1800800800",
      "faxnumber": "617-897-0902",
      "email": "someone@email.com",
      "currencycode": "USD",
      "physicalwh": 6000,
      "vatregion": 1000,
      "orghiertype": 50,
      "orghiervalue": 1,
      "orgentitytype": "R",
      "channelid": 1,
      "channelname": null,

```

```

"address1": "101 First Street",
"address2": "201 Second Street",
"address3": null,
"city": "Minneapolis",
"state": "MN",
"country": "US",
"postalcode": "55555-1234"
},
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "warehouseid": 2,
      "type": "virtual",
      "warehousename": "N. America Central",
      "whnamesecondary": null,
      "contactname": "Sue Glass",

```

```

        "phonenumber": "3122222473",
        "faxnumber": null,
        "email": null,
        "currencycode": "USD",
        "physicalwh": 2,
        "vatregion": 1000,
        "orghiertype": null,
        "orghiervalue": null,
        "orgentitytype": "R",
        "channelid": null,
        "channelname": null,
        "address1": "123 Street",
        "address2": "Anytown",
        "address3": null,
        "city": "Anycity",
        "state": "MN",
        "country": "US",
        "postalcode": "50250"      }
    ],
    "hasMore": true,
    "limit": 1,
    "count": 1,
    "links": [
        {
            "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/location/warehouse?
before=2021-05-12T20%3A15%3A13.253929%2B05%3A30&limit=1&since=1970-01-01T00%3A
00%3A00.001Z ",
            "rel": "self"
        },
        {
            "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/merchhier?
before=2021-05-12T21%3A00%3A13.036021%2B05%3A30&limit=1&offsetkey=10%2C1&since
=1970-01-01T00%3A00%3A00.001Z ",
            "rel": "next"
        }
    ]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_WH	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes
ADDR	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_WH	Yes	Yes	Yes	Yes
MERCHAPI_ORG_HIER	No	No	Yes	No
V_MERCHAPI_WH	Yes	No	No	No
V_MERCHAPI_WH_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

REST Endpoint for Omnichannel/related Item

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/relateditem

Functional Area

Items - Related Items

Business Overview

This service publishes related-item relationships to the Omnichannel application and can also be consumed by downstream consuming systems. It publishes relationships between a primary item and related items only while both item records remain approved in Merchandising, so relationships tied to non-approved primary or related items are not returned.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: RELATED_ITEM_HEAD, RELATED_ITEM_DETAIL, ITEM_MASTER

Additional business tables: none

JSON cache table: MERCHAPI_RELATED_ITEM

JSON generation view: V_MERCHAPI_RELATED_ITEM_JSON

The JSON view joins RELATED_ITEM_HEAD, RELATED_ITEM_DETAIL, and ITEM_MASTER for both the primary and related items.

Functionally, the main item and the related item must both still be approved (ITEM_MASTER.STATUS = 'A'), so the service publishes only active relationships between approved items.

The cache is refreshed from the approved relationship set in V_MERCHAPI_RELATED_ITEM_JSON and is republished when qualifying relationship or item approval changes occur.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_RELATED_ITEM_JSON and apply the rule that both item rows must have STATUS = 'A'.

ICL (Integration Change Log) table: ICL_RMS_RELATED_ITEM (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on RELATED_ITEM_HEAD, RELATED_ITEM_DETAIL, ITEM_MASTER.

Item Approval Handling

On item approval, the main item publish flow writes related-item head and detail rows where the approved item is the primary item to ICL_RMS_RELATED_ITEM so qualifying relationships can publish when the item becomes approved.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_RELATED_ITEM

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_RELATED_ITEM_ADHOC_PROCESS ->
MERCHAPI_OMNI_RELATED_ITEM_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/item/relateditem

Special handling

Additional GET query parameters: nodelevel, nodeid.

Service Types

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/relateditem

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-35 PageResultsItemsVMerchApiRelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-36 VMerchApiRelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for related item.
relationshipid	No	Number(20,0)	This column contains the Relationship ID.
item	No	String (25)	This column contains the item.

Table 5-36 (Cont.) VMerchApiRelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshiptype	No	String (6)	This column contains the relationship type.
mandatoryind	No	String (1)	This column contains the mandatory indicator.
relateditem	No	String (25)	This column contains the related item.
relateditemaction	No	String (6)	Action for related item detail.
startdate	No	date	This column contains the start date.
enddate	No	date	This column contains the end date.
priority	No	Number(4,0)	This column contains the priority.

Table 5-37 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "relationshipid": 1,
      "item": null,
      "relationshiptype": null,
      "mandatoryind": null,
      "relateditem": null,
      "relateditemaction": null,
      "startdate": "2001-12-31",
      "enddate": "2001-12-31",
      "priority": 1
    }
  ],
  "hasMore": true,
}
```

```

"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "relationshipid": 120001,
      "item": "111150001",
      "relationshiptype": "CRSL",
      "mandatoryind": "N",
      "relateditem": "111051290",
      "relateditemaction": "INSERT",
      "startdate": null,
      "enddate": null,
      "priority": null
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [

```

```

    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/item/relateditem?
before=2021-05-06T18%3A17%3A30.285106%2B08%3A00&limit=1&since=1970-01-01T00%3A
00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http://localhost:8080/MerchIntegrations/services/foundation/
omnichannel/item/relateditem?
before=2021-05-06T18%3A17%3A30.285106%2B08%3A00&limit=1&offsetkey=120001%2C111
051290&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_RELATED_ITEM	Yes	Yes	Yes	Yes
MERCHAPI_RELATED_ITEM	Yes	Yes	Yes	Yes
V_MERCHAPI_RELATED_ITEM_JSON	Yes	No	No	No
V_MERCHAPI_RELATED_ITEM	Yes	Yes	Yes	Yes

REST Endpoint for Omnichannel/Retail Store details

Endpoint

MerchIntegrations/services/foundation/omnichannel/location/retailstore

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service publishes retail store details to the Omnichannel application and can also be consumed by downstream consuming systems. It publishes each store together with its company-through-district hierarchy placement and the primary address maintained for the store or workforce-store address modules in Merchandising.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: STORE, ADDR

Additional business tables: STORE_HIERARCHY, ADD_TYPE_MODULE

JSON cache table: MERCHAPI_RETAIL_STORE

JSON generation view: V_MERCHAPI_RETAIL_STORE_JSON

The JSON view joins STORE, STORE_HIERARCHY, ADDR, and ADD_TYPE_MODULE, so each published row carries the store's company-through-district placement together with the primary address maintained for store modules ST and WFST.

The store cache shares its loader with the omnichannel organization hierarchy and warehouse caches through the shared API_OMNI_ORG_HIER_STR_WH process.

Address publication is limited to the primary address rows for store modules ST and WFST.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_RETAIL_STORE_JSON and publish store hierarchy placement together with the primary address rows for modules ST and WFST.

Rebuild deletes or marks rows when the store no longer exists in the source view.

ICL (Integration Change Log) table: ICL_RMS_STORE, ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

Store changes come from ICL_RMS_STORE, and address-only changes come from ICL_RMS_ADDR for ST and WFST addresses when no store ICL row exists.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_ORG_HIER_STR_WH

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_ORG_HIER_STR_WH_ADHOC_PROCESS ->
MERCHAPI_OMNI_ORG_HIER_STR_WH_ADHOC_JOB

Webhook configuration api name: foundation/omnichannel/location/retailstore

Special handling

Additional GET query parameters: nodelevel, nodeid.

Supported nodelevel values for this service are COMPANY, CHAIN, AREA, REGION, DISTRICT, STORE.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/location/retailstore

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-38 PageResultsItemsVMerchapiRetailStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-38 (Cont.) PageResultsItemsVMerchapiRetailStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-39 VMerchapiRetailStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for the store.
storeid	No	Number(10,0)	This column contains the number which uniquely identifies the store.
storename	No	String (150)	This column contains the name of the store which, along with the store number, identifies the store.
storename10	No	String (10)	This column contains a ten-character abbreviation of the store name.
manager	No	String (120)	This column contains the name of the store manager.
phonenumber	No	String (20)	This column contains the phone number for the store.
faxnumber	No	String (20)	This column contains the fax number for the store.
email	No	String (100)	This column contains the email address for the location.
totalsqfeet	No	Number(8,0)	This column contains the total square footage of the store.
sellingsqfeet	No	Number(8,0)	This column contains the total square footage of the store's selling area.

Table 5-39 (Cont.) VMerchapiRetailStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencycode	No	String (3)	This column contains the currency code under which the store operates.
vatregion	No	Number(4,0)	This column contains the number of the Value Added Tax region in which this store is contained.
stockholding	No	String (1)	This column indicates whether the store can hold stock. In a non-multichannel environment this will always be Y.
integratedpos	No	String (1)	This column indicates whether or not the POS system at the store is integrated.
contactname	No	String (120)	This column contains the name of the contact for the store at this address.
address1	No	String (240)	This column contains the first line of the store address.
address2	No	String (240)	This column contains the second line of the store address.
address3	No	String (240)	This column contains the third line of the store address.
city	No	String (120)	This column contains the name of the city that is associated with the store address.
state	No	String (3)	This column contains the state abbreviation for the store address.
country	No	String (3)	This column contains the country where the store address exists.
postalcode	No	String (30)	This column contains the zip code for the store address.

Table 5-40 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "storeid": 1521,
      "storename": "Albuquerque",
      "storename10": "Albuquerqu",
      "manager": "Paul Petros",
      "phonenumber": "505-388-5577",
      "faxnumber": "458121212",
      "email": "someone@oracle.com",
      "totalsqfeet": 4500,
      "sellingsqfeet": 4000,
      "currencycode": "USD",
      "vatregion": 1000,
      "stockholding": "Y",
      "integratedpos": "Y",
      "contactname": "Sue Glass",
      "address1": "123 Street",
      "address2": "Anaheim",
      "address3": "California",
      "city": "Anaheim",
      "state": "MN",
      "country": "US",
      "postalcode": "50250"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

```

}
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "storeid": 1111,
      "storename": "Rebuild with refresh N NO ICL Charlotte **",
      "storename10": "Charlotte",
      "manager": "Don Wright",
      "phonenum": "704-555-2121",
      "faxnumber": null,
      "email": null,
      "totalsqfeet": 64400,
      "sellingsqfeet": 59500,
      "currencycode": "USD",
      "vatregion": 1000,
      "stockholding": "Y",
      "integratedpos": "Y",
      "contactname": "Sue Glass",
      "address1": "123 Street",
      "address2": "Anytown",
      "address3": null,
      "city": "Anycity",
      "state": "MN",
      "country": "US",
      "postalcode": "50250"
    }
  ],
  "hasMore": true,

```

```

    "limit": 1,
    "count": 1,
    "links": [
      {
        "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/location/retailstore?
before=2021-05-12T17%3A30%3A45.385517%2B05%3A30&nodelevel=COMPANY&limit=1&node
id=1&since=1970-01-01T00%3A00%3A00.001Z",
        "rel": "self"
      },
      {
        "href": "http://<hostname>:<portnumber>/MerchIntegrations/
services/foundation/omnichannel/location/retailstore?
before=2021-05-12T17%3A30%3A45.385517%2B05%3A30&nodelevel=COMPANY&limit=1&node
id=1&offsetkey=1111&since=1970-01-01T00%3A00%3A00.001Z",
        "rel": "next"
      }
    ]
  }
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_STORE	Yes	Yes	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes
ADDR	Yes	No	No	No
STORE	Yes	No	No	No
MERCHAPI_RETAIL_STORE	Yes	Yes	Yes	Yes
MERCHAPI_ORG_HIER	No	No	Yes	No
V_MERCHAPI_RETAIL_STORE_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

REST Endpoint for Omnichannel/Dimension Type details

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/dimensiontype

Functional Area

Items - Item Definition

Business Overview

This service publishes dimension-type rows to the Omnichannel application and can also be consumed by downstream consuming systems. Functionally, the published dimensionSystem is resolved as the item itself, the item parent, or the item grandparent depending on item level, and the service publishes dimension-type rows for differentiator group-level data or for ID-level data when the item resolves to a style dimension system.

This service retrieves the item differentiator types 1, 2 and 3 (referred to as dimension types in XStore). For example, it might return that an item is differentiated by 'color' (diff 1 type), 'waist size' (diff 2 type) and 'inseam' (diff 3 type). While items can be defined with up to four differentiators, this service only returns up to 3 because the Omnichannel cloud services only support up to 3 dimensions. In order to fetch new updates pass the parameter of since/before timestamp.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

The data processing for this service is controlled by the main 'foundation/omnichannel/item' service. Refer to API documentation for additional batch configuration details.

Core business tables: ITEM_MASTER, DIFF_GROUP_HEAD, DIFF_GROUP_DETAIL, DIFF_IDS

Additional business tables: none

JSON cache table: MERCHAPI_ITEM_DIFF_TYPE

JSON generation view: not applicable; cache rows are derived from the shared V_MERCHAPI_ITEM_JSON item source

The shared item source already resolves item level, transaction level, item parent, item grandparent, differentiator level, differentiator type, and differentiator description for DIFF_1 through DIFF_3.

The dimension-type cache unpivots those DIFF_1 through DIFF_3 columns, resolves dimensionSystem as item, item parent, or item grandparent based on item level, and publishes GROUP differentiator rows plus ID differentiator rows only when the derived item type resolves to STYLE.

Qualifying diff-group and diff-ID update flows can also republish affected dimension-type rows after resolving the impacted items from the shared item source.

Webhook configuration api name: foundation/omnichannel/item/dimensiontype

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/dimensiontype

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
limit	No	String (6)	Pagination limit. Default value is 10000.
offsetkey	No	String	Offset key
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-41 PageResultsItemsVMerchapiDimType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch

Table 5-41 (Cont.) PageResultsItemsVMerchapiDimType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-42 VMerchapiDimType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for the Dimension Type group.
itemaction	No	String (6)	Action for the Item changes related to Dimension Type Group.
dimensionsystem	No	String (25)	This column contains the dimension system. This contains the item, item parent or the item grandparent depending on the item level.
dimension	No	String (6)	This column contains the dimension. This is the differentiator type of the item.
description	No	String (120)	This field contains the description of the dimension.
seq	No	Number(1,0)	This column contains the sequence number of the dimension. Valid values are 1 for Diff 1, 2 for Diff 2 and 3 for Diff 3.

Table 5-43 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-43 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "itemaction": null,
      "dimensionsystem": "1000000010",
      "dimension": "MT",
      "description": "Mens TShirt Small",
      "seq": 1
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
Items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.

Elements	Description
Limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
Links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "action": "INSERT",
      "itemaction": null,
      "dimensionsystem": "1000023212",
      "dimension": "C_3815",
      "description": "Smoke Diff Group Clr",
      "seq": 1
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/
dimensiontype?
before=2021-05-11T17%3A41%3A38.052402%2B00%3A00&limit=1&since=1970-01-01T00%3A00%3A00.001
Z",
      "rel": "self"
    },
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/
dimensiontype?
before=2021-05-11T17%3A41%3A38.052402%2B00%3A00&limit=1&offsetkey=0430.PD.1%2C1%2CC_3815&
since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_ITEM_DIFF_TYPE	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_JSON	Yes	No	No	No
V_MERCHAPI_DIM_TYPE	Yes	No	No	No

REST Endpoint for Omnichannel/Dimension Value details

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/dimensionvalue

Functional Area

Items - Item Definition

Business Overview

This service publishes dimension-value rows to the Omnichannel application and can also be consumed by downstream consuming systems. Functionally, the published dimensionSystem is resolved as the item itself, the item parent, or the item grandparent depending on item level, and the cache publishes differentiator values only for rows where DIFF_LEVEL = 'ID'. The display sequence is derived from the matching differentiator-group detail.

This service retrieves item differentiators (referred to as 'dimensions' in XStore). Only differentiators that are associated with approved items are returned.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

The data processing for this service is controlled by the main 'foundation/omnichannel/item' service. Refer to API documentation for additional batch configuration details.

Core business tables: ITEM_MASTER, DIFF_IDS

Additional business tables: DIFF_GROUP_DETAIL

JSON cache table: MERCHAPI_ITEM_DIFF_VALUE

JSON generation view: not applicable; cache rows are derived from the shared V_MERCHAPI_ITEM_JSON item source

The shared item source already resolves item level, transaction level, item parent, item grandparent, differentiator level, differentiator type, and differentiator description for DIFF_1 through DIFF_3.

The dimension-value cache unpivots those DIFF_1 through DIFF_3 columns, resolves dimensionSystem as item, item parent, or item grandparent based on item level, keeps only ID differentiator rows, and derives displayseq from DIFF_GROUP_DETAIL for the matching differentiator group and differentiator value combination.

Qualifying diff-ID update flows can also republish affected dimension-value rows after resolving the impacted items from the shared item source.

Webhook configuration api name: foundation/omnichannel/item/dimensionvalue

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/dimensionvalue

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
limit	No	String (6)	Pagination limit. Default value is 10000.
offsetkey	No	String	Offset key
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-44 PageResultsItemsVMerchapiDimValue - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-44 (Cont.) PageResultsItemsVMerchapiDimValue - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-45 VMerchapiDimValue - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for the Dimension Value group.
itemaction	No	String (9)	Action for the item changes related to the Dimension Value group.
dimensionsystem	No	String (25)	This column contains the dimension system. This contains the item, item parent or the item grandparent depending on the item level.
dimension	No	String (6)	This column contains the dimension. This is the differentiator type of the item.
value	No	String (10)	This column contains the value of the dimension.
description	No	String (120)	This field contains the description of the dimension.
displayseq	No	Number(4,0)	This column contains the display sequence number of the dimension.

Table 5-46 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "itemaction": null,
      "dimensionsystem": "1000000010",
      "dimension": "MT",
      "value": "MT-Small",
      "description": "Mens TShirt Small",
      "displayseq": 1
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
Items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.

Elements	Description
Limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
Links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "action": "INSERT",
      "itemaction": "NO_CHANGE",
      "dimensionsystem": "1000012345",
      "dimension": "C_3815",
      "value": "BLACK_8742",
      "description": "Auto Black BLACK_8742",
      "displayseq": 1
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/
dimensionvalue?
before=2021-05-11T17%3A51%3A17.671921%2B00%3A00&limit=1&since=1970-01-01T00%3A00%3A00.001
Z",
      "rel": "self"
    },
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/
dimensionvalue?
before=2021-05-11T17%3A51%3A17.671921%2B00%3A00&limit=1&offsetkey=0430.PD.1%2CC_3815%2CBL
ACK_8742&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCH_BATCH_PARAM	Yes	No	No	No
MERCHAPI_ITEM_DIFF_VALUE	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_JSON	Yes	No	No	No
V_MERCHAPI_DIM_VALUE	Yes	No	No	No

REST Endpoint for Omnichannel/future Inventory

Endpoint

MerchIntegrations/services/inventory/omnichannel/inventory/futureinventory

Functional Area

Inventory

Business Overview

This service publishes future inventory to the Omnichannel application and can also be consumed by downstream consuming systems. It includes only the future inventory that is relevant to customer-order fulfillment for active, transaction-level, orderable items that are backorder-enabled at stockholding customer-order locations, together with the backorder, open-order, and received quantities that still contribute to fulfillment. The calculation excludes non-customer-order allocation branches from open order quantity and applies the source pack rules so warehouse sellable packs publish at pack level while store sellable packs and non-sellable packs publish through component items.

This service retrieves future inventory information aggregated at an item level. The service returns on-order, received, and back-ordered quantity against an item.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: STORE, WH, ITEM_LOC_SOH, ITEM_MASTER, ITEM_LOC_TRAITS, ALLOC_HEADER

Additional business tables: ALLOC_DETAIL, ORDHEAD, ORDLOC, PERIOD, V_PACKSKU_QTY

JSON cache table: MERCHAPI_FUTURE_AVAIL_CO_INV

JSON generation view: V_BDI_CO_FUTURE_AVAIL

The source view pulls future availability from STORE, WH, ITEM_LOC_SOH, ITEM_MASTER, ITEM_LOC_TRAITS, ALLOC_HEADER, ALLOC_DETAIL, ORDHEAD, ORDLOC, PERIOD, and V_PACKSKU_QTY.

Functionally, it publishes only the future inventory that is relevant to customer-order fulfillment: active transaction-level orderable items that are backorder-enabled at stockholding customer-

order locations, together with the backorder, open-order, and received quantities that still contribute to fulfillment.

The calculation excludes non-customer-order allocation branches from open order quantity and applies pack rules so sellable warehouse packs publish at pack level while store sellable packs and non-sellable packs publish through their component items.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_BDI_CO_FUTURE_AVAIL. No additional filter criteria were identified beyond the cache-build logic.

Mark rows deleted when the item is no longer backorderable in ITEM_LOC_TRAITS.

ICL (Integration Change Log) table: none; delta processing compares V_BDI_CO_FUTURE_AVAIL directly to MERCHAPI_FUTURE_AVAIL_CO_INV

No ICL table is used; both delta and refresh compare V_BDI_CO_FUTURE_AVAIL directly to MERCHAPI_FUTURE_AVAIL_CO_INV.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

This service does bulk heavy data processing for the entire dataset and hence this job should be run once or twice per day. This should not be configured to run hourly or multiple times in an hour.

Process configuration name: API_OMNI_FUTURE_AVL_CO_INV

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_FUT_AVL_CO_INV_ADHOC_PROCESS ->
MERCHAPI_OMNI_FUTURE_AVL_CO_INV_ADHOC_JOB

Webhook configuration api name: inventory/omnichannel/inventory/futureinventory

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/omnichannel/inventory/futureinventory

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
offsetkey	No	String	Offset key
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-47 PageResultsItemsVMerchapiFutureAvailColnv - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-48 VMerchapiFutureAvailColnv - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Item Future Available Inventory.
item	No	String (25)	This column contains the item.
onorderquantity	No	Number(12,4)	This column contains the on order quantity of the item.

Table 5-48 (Cont.) VMerchapiFutureAvailColnv - Object. See list of elements for detail

Element Name	Required	Data Type	Description
receivedquantity	No	Number(12,4)	This column contains the received quantity of the item.
backorderquantity	No	Number(12,4)	This column contains the backorder quantity of the item.

Table 5-49 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "item": "100001000",
      "onorderquantity": 10,
      "receivedquantity": 10,
      "backorderquantity": 10
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "action": "UPDATE",
      "item": "105700020",
      "onorderquantity": "0",
      "receivedquantity": "0",
      "backorderquantity": "30"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/inventory/
omnichannel/inventory/futureinventory?
before=2021-05-06T20%3A56%3A51.316158%2B08%3A00&limit=1&since=1970-01-01T00%3A
00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http:// <hostname>/MerchIntegrations/services/inventory/
omnichannel/inventory/futureinventory?
before=2021-05-06T20%3A56%3A51.316158%2B08%3A00&limit=1&offsetkey=105700020&si
nce=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
MERCHAPI_BOOKMARK	Yes	Yes	Yes	Yes
V_BDI_CO_FUTURE_AVAIL	Yes	No	No	No
MERCHAPI_FUTURE_AVAIL_CO_INV	Yes	Yes	Yes	Yes
V_MERCHAPI_FUTURE_AVAIL_CO_INV	Yes	No	No	No

REST Endpoint for Omnichannel/item Image

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/image

Functional Area

Items - Item Definition

Business Overview

This service publishes item images to the Omnichannel application and can also be consumed by downstream consuming systems. The source view publishes images only for currently approved items, and when callers supply nodelevel and nodeid the service returns images only for items that also have a matching MERCHAPI_ITEM_LOC row at the requested store or warehouse.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_IMAGE

Additional business tables: ITEM_MASTER

JSON cache table: MERCHAPI_ITEM_IMAGE

JSON generation view: V_MERCHAPI_ITEM_IMAGE_JSON

The JSON view is built from ITEM_IMAGE and ITEM_MASTER, and it explicitly filters to ITEM_MASTER.STATUS = 'A', so only images for currently approved items are published from the source view.

When callers supply `nodelevel` and `nodeid`, the REST query further requires a matching `MERCHAPI_ITEM_LOC` row for the item at the requested store or warehouse, which scopes the published images to items active at that node.

When `nodelevel` and `nodeid` are supplied, the service requires a matching `MERCHAPI_ITEM_LOC` row for the item at the requested node.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use `V_MERCHAPI_ITEM_IMAGE_JSON` and apply `ITEM_MASTER.STATUS = 'A'`.

ICL (Integration Change Log) table: `ICL_RMS_ITEM_IMAGE` (ICL consumer name = `MERCHAPI`)

ICL entries are created using item-image changes, while `ITEM_MASTER.STATUS = 'A'` is enforced through the source view when rows are rebuilt or merged.

Item Approval Handling

On item approval, the main item publish flow writes existing `ITEM_IMAGE` rows for the item to `ICL_RMS_ITEM_IMAGE` so image data is also queued when the item becomes approved.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: `API_OMNI_ITEM_IMAGE`

To enable via Background engine set `BACKGROUND_RUN_FLAG` to Y and review the `BACKGROUND_RUN_INTERVAL_SECOND` parameter.

POM Batch schedule process and job:
`MERCHAPI_DELTA_OMNI_ITEM_IMAGE_ADHOC_PROCESS ->`
`MERCHAPI_OMNI_ITEM_IMAGE_ADHOC_JOB`

Webhook configuration api name: `foundation/omnichannel/item/image`

Special handling

Additional GET query parameters: `nodelevel`, `nodeid`.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately.

Multi-threading and commit control

This service batch processing for delta and initial seeding is single threaded.

The delivered setting for `API_COMMIT_MAX_COUNTER` is currently 1000 and it can be extend to 10,000.

REST fetch considerations

REST page size should be selected based on average payload size so the average response remains below about 10 MB per call. Common page-limit values are 1,000 or 10,000 rows.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the LAST_UPDATE_DATETIME index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping BUILD_HISTOGRAM enabled allows the service to support recovery through smaller, more selective polling windows using the optimalBefore query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides optimalBefore and leaves before empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups LAST_UPDATE_DATETIME values into buckets of 10,000 rows in MERCHAPI_TIMESTAMP_HISTOGRAM using the shared HISTOGRAM_ROW_COUNT configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/image

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-50 PageResultsItemsVMerchapiltemImage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-51 VMerchapiltemImage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for the Item Image.
item	No	String (25)	This column contains the unique alphanumeric identifier for the item.
imagename	No	String (120)	This column contains the name of the image of the item.
imageaddr	No	String (255)	This field contains the actual path where the file of the image of the item is stored.

Table 5-51 (Cont.) VMerchapiItemImage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageuri	No	String (375)	This column contains the URI of the image of the item.
imagedesc	No	String (40)	This field contains the description associated with the image of the item.
imagetype	No	String (6)	This field contains the type of the image of the item. Valid values are defined as members of IITD code type.
primaryind	No	String (1)	This field indicates whether this record is the primary image of the item or not. Valid values are Y and N. It defaults to N if left blank or set to null.
displaypriority	No	Number(4,0)	This field specifies the display sequence order of images associated with the item by priority.

Table 5-52 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "item": "1000000010",
      "image name": "imageName1",
      "image addr": "http://www.oracle.com/",
      "image uri": "http://www.oracle.com/imageName1",

```



```

"imagedesc": "imageDesc1",
"imagetype": "T",
"primaryind": "Y",
"displaypriority": 1
},
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "item": "100200042",
      "image name": "image1",
      "imageaddr": "http://www.oracle.com/",
      "imageuri": "http://www.oracle.com/image1",
      "imagedesc": "image1Desc",
      "imagetype": "T",
      "primaryind": "Y",
      "displaypriority": 1
    }
  ],

```

```

    "hasMore": true,
    "limit": 1,
    "count": 1,
    "links": [
      {
        "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/
image?
before=2021-05-06T17%3A32%3A13.265046%2B08%3A00&limit=1&since=1970-01-01T00%3A00%3A00.001
Z",
        "rel": "self"
      },
      {
        "href": "http:// <hostname>/MerchIntegrations/services/foundation/omnichannel/item/
image?
before=2021-05-06T17%3A32%3A13.265046%2B08%3A00&limit=1&offsetkey=100200042%2Cimage1&sinc
e=1970-01-01T00%3A00%3A00.001Z",
        "rel": "next"
      }
    ]
  }
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
MERCHAPI_BOOKMARK	Yes	Yes	Yes	Yes
ICL_RMS_ITEM_IMAGE	Yes	Yes	Yes	Yes
ITEM_MASTER	Yes	No	No	No
ITEM_IMAGE	Yes	No	No	No
MERCHAPI_ITEM_IMAGE	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_IMAGE_JSON	Yes	No	No	No
V_MERCHAPI_ITEM_IMAGE	Yes	No	No	No

REST Endpoint for Omnichannel/Item Initial Price details

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/initialprice

Functional Area

Items - Item Definition

Business Overview

This service should not be used for external consumption. This service is very specific for XStore consumption and the roadmap plan is to deprecate this service in future once XStore dependency is removed. The information provided by this service is same as 'foundation/omnichannel/item/itemlocation'. The term 'initial price' for this service is misleading for external consumption because this service provides the current price for the item/location combination.

If Xstore is part of the implementation, refer to the technical details for API documentation for 'foundation/omnichannel/item/itemlocation' because the underlying components are the same.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/initialprice

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 10000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-53 PageResultsItemsVMerchApiInitialItemPrice - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-54 VMerchApiInitialItemPrice - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Item details.
loctype	No	String (1)	This field specifies the type of location in the location field. Valid values are S (store) and W (warehouse).
location	No	Number(10,0)	This field contains the numeric identifier of the location in which the item is to be found. This field may contain a store or warehouse.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.
pricetype	No	String (7)	This field specifies the price type. Valid value is INITIAL.

Table 5-54 (Cont.) VMerchApiInitialItemPrice - Object. See list of elements for detail

Element Name	Required	Data Type	Description
price	No	Number(20,4)	This field specifies the initial unit retail price in the standard unit of measure for the item/location combination. This field is stored in the local currency.
effective	No	date	This field specifies the date on which the unit retail price became or will become active. This field is always null in the current implementation.
source	No	String (6)	This field contains the source of the initial unit retail price of each item. This field is always direct in the current implementation.
eventid	No	String	This field specifies the event ID. This field is always null in the current implementation.
resetind	No	String	This field specifies the reset indicator. This field is always null in the current implementation.
clearanceind	No	Number(1,0)	This field specifies if the item is on clearance at the store. A value of 1 indicates clearance.

Table 5-55 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "loctype": "S",
      "location": 1521,
      "item": "100012012",
      "pricetype": "INITIAL",
      "price": 19.5,
      "effective": "2001-12-31",
      "source": "direct",
      "eventid": null,
      "resetind": null,
      "clearanceind": 1
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	“self” : This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "loctype": "S",
      "location": 121,
      "item": "100000104",
      "pricetype": "INITIAL",
      "price": 12.1,
      "effective": null,
      "source": "direct",
      "eventid": null,
      "resetind": null,
      "clearanceind": 0
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/initialprice?before=2021-05-11T17%3A23%3A22.208547%2B00%3A00&nodelevel=COMPANY&limit=1&nodeid=9118&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "self"
    },
    {
      "href": "http://<hostname>/MerchIntegrations/services/foundation/omnichannel/item/initialprice?before=2021-05-11T17%3A23%3A22.208547%2B00%3A00&nodelevel=COMPANY&limit=1&nodeid=9118&offsetkey=S%2C121%2C100000104&since=1970-01-01T00%3A00%3A00.001Z",
      "rel": "next"
    }
  ]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_ITEM_LOC	Yes	Yes	Yes	Yes
ITEM_MASTER	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
MERCHAPI_ITEM_LOC	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_LOC_JSON	Yes	No	No	No
V_MERCHAPI_INITIAL_ITEM_PRICE	Yes	No	No	No

REST Endpoint for Omnichannel/Item Location details

Endpoint

MerchIntegrations/services/foundation/omnichannel/item/itemlocation

Functional Area

Items - Item Definition

Business Overview

This service publishes the item/location payload to the Omnichannel application and can also be consumed by downstream consuming systems. It serves the store and warehouse projection of the shared item/location cache with 'foundation/item/location' service and returns the location-specific selling, sourcing, VAT-code, and tax-group details used by omnichannel selling flows. Although the shared cache also supports broader location coverage for other services, this omnichannel projection does not return external-finisher rows.

The shared source view behind the cache is limited to items whose current ITEM_MASTER.STATUS is A or D. This service then applies its own store or warehouse query scope and enriches matching rows with tax-group data when corresponding GTS cache data exists in the requested window.

Refer to the api documentation for 'foundation/item/location' for technical and performance consideration for this service as these two share the same underlying component.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/item/itemlocation

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
nodelevel	No	String	Node Level
nodeid	No	String (10)	Node Id
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 10000.

Parameter Name	Required	Data Type	Description
changeType	No	String	Filter records that were created or deleted within since/ before time range. Valid values create or delete
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-56 PageResultsItemsVMerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-57 VMerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for item details.
loctype	No	String (1)	This field specifies the type of location in the location field. Valid values are S (store) and W (warehouse).
location	No	Number(10,0)	This field contains the numeric identifier of the location in which the item is to be found. This field may contain a store or warehouse.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.
sellingunitretail	No	Number(20,4)	This field specifies the unit retail price in the selling unit of measure for the item/location combination. This field is stored in the local currency.
sellinguom	No	String (4)	This field specifies the selling unit of measure for an item's single-unit retail.
sellingUomDecimalPrecision	No	Number(1,0)	This field defines the number of decimal places to be considered for the specific selling UOM in XOCS.
taxableind	No	String (1)	This field indicates if the item is taxable at the store.
localitemdesc	No	String (250)	This field specifies the local description of the item. This field will default to the item's description but will be overridable. This value will be downloaded to the POS.
status	No	String (1)	This field specifies the current status of the item at the store.

Table 5-57 (Cont.) VMerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
qtykeyoptions	No	String (6)	This field specifies whether the qty key on a POS should be used for this item at the location. Valid values are in the code_type RPO. Current values include R - required, P - prohibited, and O - optional. Additional values can be added to the code type as needed. This value will be downloaded to the POS as needed.
manualpriceentry	No	String (6)	This field specifies whether the price can or should be entered manually on a POS for this item at the location.
foodstampind	No	String (1)	This field specifies whether the item is approved for food stamps at the location. This value will be downloaded to the POS.
fixedtarevalue	No	Number(12,4)	This field holds the value associated with the packaging in items sold by weight at the location. Fixed tare is the tare of the packaging used. The only processing RMS does involving the fixed tare value is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to RMS. This kind of processing is the responsibility of the client and should occur before sales are sent to any Oracle Retail application.

Table 5-57 (Cont.) VMerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fixedtareuom	No	String (4)	This field holds the unit of measure value associated with the tare value. The only processing RMS does involving the fixed tare value and UOM is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to RMS. This kind of processing is the responsibility of the client and should occur before sales are sent to any Oracle Retail application.
stopsaleind	No	String (1)	This field specifies whether sale of the item should be stopped immediately at the location (for example, in case of recall). This value will be downloaded to the POS.
returnableind	No	String (1)	This field contains a value of Yes when the item can be returned to the location.
backorderind	No	String (1)	This field contains a value of Yes when the item can be back ordered to the location.
merchandiseind	No	String (1)	This field indicates if the item is a merchandise item.
clearanceind	No	String (1)	This field indicates if the item is on clearance at the store.
crosssell	No	String (1)	This field indicates if a substitute item exists for the main item.
attacheditem	No	String (1)	This field indicates if cross-sell or up-sell items are attached to the main item.

Table 5-57 (Cont.) VMerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatcodes	No	String (4000)	This field specifies VAT code information for the item.
localshortdescription	No	String (120)	This field contains the local short description of the item. This field will default to the item's short description but will be overridable. This value will be downloaded to the POS.
sourcemethod	No	String (1)	This value will be used to specify how the ad-hoc PO or TSF creation process should source the item/location request. If the value is Warehouse, the process will attempt to fill the request by creating a transfer from the warehouse mentioned in the source_wh field. If this warehouse does not have enough inventory to fill the request, a purchase order will be created for the item/location's primary supplier. For warehouses, it is used by Oracle Retail Allocation to determine the valid sources and destinations for warehouse-to-warehouse allocations.
sourcewarehouse	No	Number(10,0)	This value will be used by the ad-hoc PO or transfer creation process to determine which warehouse to fill the store's request from. It will also be used by the Allocation process to support warehouse-to-warehouse allocations. A value will be required in this field if the sourcing method is Warehouse.

Table 5-57 (Cont.) VMerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
taxGroupId	No	String (60)	Tax group ID.

Table 5-58 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "loctype": "S",
      "location": 1521,
      "item": "100100076",
      "sellingunitretail": 19.5,
      "sellinguom": "EA",
      "sellingUomDecimalPrecision": 1,
      "taxableind": "Y",
      "localitemdesc": "localItemDesc",
      "status": null,
      "qtykeyoptions": null,
      "manualpriceentry": null,
      "foodstampind": null,
      "fixedtarevalue": 1.0,
      "fixedtareuom": null,
      "stopsaleind": null,
      "returnableind": null,
      "backorderind": null,
      "merchandiseind": "Y",
      "clearanceind": null,
      "crosssell": null,
      "attacheditem": null,
      "vatcodes": null,
      "localshortdescription": "localShortDesc",
      "sourcemethod": null,
      "sourcewarehouse": 1,
      "taxGroupId": null
    }
  ],
}
```

```

"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

vatcodes JSON Output

Name	Data Type	Description
code	String	This field contains vat code which is used to uniquely identify a VAT rate.
date	Date	This field contains date on which VAT rate become active. Format: YYYY-MM-DD

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "loctype": "S",
      "location": 1111,
      "item": "100750001",
      "sellingunitretail": "132.13",

```

```

        "sellinguom": "EA",
        "taxableind": "Y",
        "localitemdesc": "100750001",
        "status": "A",
        "qtykeyoptions": null,
        "manualpriceentry": null,
        "foodstampind": null,
        "fixedtarevalue": null,
        "fixedtareuom": null,
        "stopsaleind": null,
        "returnableind": null,
        "backorderind": null,
        "merchandiseind": "Y",
        "clearanceind": "N",
        "crosssell": "N",
        "attacheditem": "N",
        "vatcodes": "[{\"code\":\"S\",\"date\":\"2020-07-25\"}]",
        "localshortdescription": "New Component Item",
        "sourcemethod": "W",
        "sourcewarehouse": 10001,
        "taxGroupId": null
    }
],
"hasMore": true,
"limit": 1,
"count": 1,
"links": [
    {
        "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/item/itemlocation?
before=2021-05-03T15%3A50%3A19.189562000%2B00%3A00&nodelevel=COMPANY&limit=2&n
odeid=1&offsetkey=S%2C1111%2C100550000%26since%3D1970-01-01T00%3A00%3A00.001Z&
since=1970-01-01T00%3A00%3A00.001000000%2B00%3A00",
        "rel": "self"
    },
    {
        "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/item/itemlocation?
before=2021-05-03T15%3A50%3A19.189562000%2B00%3A00&nodelevel=COMPANY&limit=2&n
odeid=1&offsetkey=S%2C1151%2C1000000067%26since%3D1970-01-01T00%3A00%3A00.001Z&
since=1970-01-01T00%3A00%3A00.001000000%2B00%3A00",
        "rel": "next"
    }
]
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_ITEM_LOC	Yes	Yes	Yes	Yes
ITEM_MASTER	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
MERCHAPI_ITEM_LOC	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_LOC_JSON	Yes	No	No	No
V_MERCHAPI_ITEM_LOC	Yes	No	No	No
MERCHAPI_GTS_TAX_LOC_ITEM	Yes	No	No	No
MERCHAPI_GTS_TAX_LOC	Yes	No	No	No

REST Endpoint for Omnichannel/vat

Endpoint

MerchIntegrations/services/foundation/omnichannel/vat

Functional Area

Financials - Taxes

Business Overview

This service publishes VAT definitions to the Omnichannel application and can also be consumed by downstream consuming systems. It publishes VAT region and VAT code rate definitions for non-exempt tax regions only, and it derives the published end date from the next more recent effective row for the same VAT region and VAT code combination.

This includes data such as VAT code, active date, and VAT rate, and is applicable for VAT implementations only (Default Tax Type system option is 'Simple VAT'). This service does not support incremental date updates, it always returns all VAT related information, even if only one VAT definition was modified.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: VAT_CODES

Additional business tables: VAT_CODE_RATES, VAT_REGION

JSON cache table: MERCHAPI_VAT

JSON generation view: V_MERCHAPI_VAT_JSON

The JSON view is built from VAT_CODES, VAT_CODE_RATES, and VAT_REGION.

Functionally, it publishes VAT-region and VAT-code combinations for non-exempt tax regions only (VAT_REGION.VAT_CALC_TYPE <> 'E'), ranks ACTIVE_DATE descending within each (vat_region, vat_code) pair, and derives endDate from the next newer effective row for that same region/code combination.

This API is unusual compared with the rest of the omnichannel caches because delta maintenance compares the current VAT source directly to the cache table rather than using row-level ICL driven change capture.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_VAT_JSON and apply VAT_REGION.VAT_CALC_TYPE <> 'E'.

ICL (Integration Change Log) table: none; refresh and delta processing compare V_MERCHAPI_VAT_JSON directly to MERCHAPI_VAT

No ICL table is used; fine-grained delete and update publishing is intentionally not implemented because omnichannel VAT uses kill-and-fill behavior.

Related bulk publish handling for item/location cache

Tomorrow-effective VAT rate activation through VAT code-rate maintenance using batch job VATDLXPL and its prepost (post) can insert a record into MERCHAPI_ASYNC_REQUEST for foundation/omnichannel/item/itemlocation with REQUEST_TYPE = DATA_REFRESH and REFRESH_TYPE = REBUILD when the VAT change can affect a broad item/location population. This bulk publish handling is driven through the Vat Rate Change Explode (VATDLXPL) flow as part of prepost.pc.

If the future-effective change is narrower and item-specific, targeted item/location delta rows are staged instead of the broader rebuild request.

Refer to foundation/item/location for the detailed item/location operational and performance implications of this bulk publish handling.

Batch Configuration. This API does not have a dedicated delivered asynchronous schedule configuration.

Process configuration name: API_OMNI_VAT

No POM batch schedule entry was identified for this API.

No API-specific BACKGROUND_RUN_FLAG or BACKGROUND_RUN_INTERVAL_SECOND rows were found for this API.

Webhook configuration api name: foundation/omnichannel/vat

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/omnichannel/vat

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-59 PageResultsItemsVMerchApiVat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-60 VMerchApiVat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for VAT details.
vatregion	No	Number(4,0)	This field gives the VAT region.
vatregionname	No	String (120)	This field gives the VAT region name.
vatcode	No	String (6)	This field gives the VAT code.
vatcodedesc	No	String (120)	This field gives the VAT code description.
activatedate	No	date	This field gives the active date.
vatrate	No	Number(20,10)	This field gives the VAT rate.
enddate	No	date	This field gives the end date.

Table 5-61 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": null,
      "vatregion": 1000,
      "vatregionname": null,
      "vatcode": "S",
      "vatcodedesc": null,
      "activatedate": "2001-12-31",
      "vatrate": 1.0,
      "enddate": "2001-12-31"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
}
```

```

"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```

" {
  "items": [
    {
      "action": "INSERT",
      "vatregion": 1000,
      "vatregionname": "Vat Region 1000",
      "vatcode": "E",
      "vatcodedesc": "Exempt",
      "activedate": "2019-02-10T00:00:00",
      "vatrate": null,
      "enddate": "2019-02-18T00:00:00"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "http:// <hostname> /MerchIntegrations/services/foundation/
omnichannel/vat?
before=2021-05-06T19%3A34%3A38.158110%2B08%3A00&limit=1&since=1970-01-01T00%3A

```

```

00%3A00.001Z",
    "rel": "self"
  },
  {
    "href": "http://<hostname>/MerchIntegrations/services/foundation/
omnichannel/vat?
before=2021-05-06T19%3A34%3A38.158110%2B08%3A00&limit=1&offsetkey=1000%2CE%2C2
019-02-10T00%3A00%3A00&since=1970-01-01T00%3A00%3A00.001Z",
    "rel": "next"
  }
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
VAT_CODES	Yes	No	No	No
VAT_CODE_RATES	Yes	No	No	No
VAT_ITEM	Yes	No	No	No
MERCHAPI_VAT	Yes	Yes	Yes	Yes
V_MERCHAPI_VAT_JSON	Yes	No	No	No
V_MERCHAPI_VAT	Yes	No	No	No

Merchandising Services

The following services are included in this functional area:

- [Administration](#)
 - [Data Privacy](#)
 - * [Data Privacy Download Service](#)
 - * [Get Personally Identifiable Information](#)
 - * [Data Privacy Upload Service](#)
 - * [Delete Personally Identifiable Information](#)
 - * [Security Pre-requisites/Considerations for accessing Data Privacy ReST Services](#)
 - [Operations](#)
 - * [Background Job Configuration Upload Service](#)
 - * [Get Batch Run Status Services](#)
 - * [Update Background Job Configuration](#)
 - * [BI Publisher Report URL Download Service](#)
 - * [Get BI Publisher Report URL](#)

- * [Calendar Download Service](#)
 - * [Get 4-5-4 Calendar](#)
- * [Codes and Descriptions Download Service](#)
 - * [Get Codes and Descriptions](#)
 - * [Get Codes and Descriptions for All or Provided Code Type](#)
- * [POM End of Week Signal Download Service](#)
 - * [Get POM End of Week Signal Detail](#)
- * [Service Status Download Service](#)
 - * [Get ReST Service Status](#)
- * [Virtual Date Download Service](#)
 - * [Get Current Virtual Date in Date Format](#)
 - * [Get Current Virtual Date in Long and Date Format](#)
- * [Webhook Configuration Download Service](#)
- * [Webhook Configuration Upload Service](#)
- [System Options](#)
 - * [System Options Download Service](#)
 - * [Get Functional System Options](#)
 - * [Get Inventory Movement System Options](#)
 - * [Get Procurement System Options](#)
- [Cost](#)
 - [Average Cost](#)
 - * [Average Cost Upload Service](#)
 - * [Update Average Cost](#)
 - [Cost Changes](#)
 - * [Cost Change Upload Service](#)
 - * [Create Cost Change Effective Immediately](#)
 - * [Manage Cost Changes](#)
 - [Pricing Cost](#)
 - * [Pricing Cost Download Service](#)
 - * [Get Pricing Cost](#)
- [Customer Orders](#)
 - [Customer Order Available Inventory Download Service](#)
 - * [Get Available Inventory for Customer Orderable Items](#)
 - [Customer Order Inventory Backorder Upload Service](#)
 - * [Create Inventory Backorder](#)
 - [Customer Order Item Substitution Upload Service](#)
 - * [Create Customer Order Item Substitution](#)
 - [Customer Order Upload Service](#)

- * [Cancel Fulfillment Order](#)
 - * [Create Fulfillment Order](#)
- [Deals](#)
 - [Deals Download Service](#)
 - * [Get Item Location Deals](#)
- [Financials](#)
 - [Budgets](#)
 - * [Half Data Budget Upload Service](#)
 - [Currency](#)
 - * [Currency Download Service](#)
 - * [Currency Exchange Rate Download Service](#)
 - * [Currency Exchange Rate Upload Service](#)
 - [General Ledger](#)
 - * [General Ledger Chart of Accounts Upload Service](#)
 - [Terms](#)
 - * [Freight Term Upload Service](#)
 - * [Payment Term Upload Service](#)
 - [Transaction Data](#)
 - * [Transaction Data Upload Service](#)
- [Foundation](#)
 - [Banners and Channels Download Service](#)
 - * [Get Banners](#)
 - * [Get Channels](#)
 - [Location List Upload Service](#)
 - * [Manage Location Lists](#)
 - [Location Trait Upload Service](#)
 - * [Create Location Trait](#)
 - * [Delete Location Trait](#)
 - * [Update Location Trait](#)
 - [Suppliers and Partners](#)
 - * [Partner Download Service](#)
 - * [Supplier and Partner Upload Service](#)
 - * [Supplier Download Service](#)
 - * [Supplier Upload Service](#)
 - [Tax Rules](#)
 - * [Tax Rule Download Service](#)
 - * [Get Item Location Tax Rules](#)
 - * [Get Store Tax Location](#)

- * [Get Tax Rule Groups](#)
 - * [Tax Rule Upload Service](#)
- [Franchise Management](#)
 - [Customer Credit Status Upload Service](#)
 - * [Update Customer Credit Status](#)
 - [Franchise Order Upload Service](#)
 - * [Create Franchise Orders](#)
 - * [Manage Franchise Orders](#)
- [Import Management](#)
 - [Actual Landed Cost](#)
 - * [Actual Landed Cost Upload Service](#)
 - * [Finalize Actual Landed Cost](#)
 - [Obligations](#)
 - * [Obligations Upload Services](#)
- [Inventory](#)
 - [Appointments](#)
 - * [Appointment Upload Service](#)
 - * [Create Appointment](#)
 - * [Delete Appointment](#)
 - * [Delete Appointment Details](#)
 - * [Manage Appointment Details](#)
 - * [Update Appointment](#)
 - [Inventory Download Service](#)
 - * [Get Item Available Inventory at Customer Orderable Locations](#)
 - * [Get Item Location Inventory](#)
 - [Inventory Adjustments](#)
 - * [Inventory Adjustment Upload Service](#)
 - * [Create Inventory Adjustment for Items at a Location](#)
 - * [Create Inventory Adjustments](#)
 - * [Item Transformation Upload Service](#)
 - * [Create Item Transformations](#)
 - [Returns To Vendor](#)
 - * [Return to Vendor Download Service](#)
 - * [RTV Publish Services](#)
 - * [Return to Vendor Upload Service](#)
 - * [Manage Returns to Vendor](#)
 - [Shipments and Receipts](#)
 - * [Receiver Unit Adjustment Download Service](#)

- * [Get Receiver Unit Adjustment](#)
- * [Shipment and Receipt Download Service](#)
 - * [Get Shipment Detail](#)
- * [Shipment and Receipt Upload Service](#)
 - * [Create DSD Receipts](#)
 - * [Create Outbound ASNs](#)
 - * [Delete Inbound ASNs](#)
 - * [Manage Inbound ASN](#)
 - * [Manage Purchase Order and Stock Order Receipts](#)
 - * [Manage Purchase Order Receipts](#)
 - * [Manage Stock Order Receipts](#)
 - * [Update DSD Receipts](#)
- [Stock Counts](#)
 - * [Stock Count Download Service](#)
 - * [Get Stock Count Detail](#)
 - * [Stock Count Upload Service](#)
 - * [Create Stock Count Schedules](#)
 - * [Delete Stock Count Schedule Locations](#)
 - * [Delete Stock Count Schedules](#)
 - * [Update Stock Count Schedules](#)
- [Transfers and Allocations](#)
 - * [Allocation Download Service](#)
 - * [Get Allocation Details for Provided Allocation](#)
 - * [Get Details for an Allocation](#)
 - * [Allocation Number Download Service](#)
 - * [Get Next Allocation Numbers Service](#)
 - * [Allocation Upload Service](#)
 - * [Close Allocations](#)
 - * [Create Allocation Details](#)
 - * [Create Allocations](#)
 - * [Delete Allocation Details](#)
 - * [Update Allocation Details](#)
 - * [Update Allocations](#)
 - * [Mobile Workflow Transfer Download Service](#)
 - * [Create Transfer - Get From Locations for Transfer](#)
 - * [Create Transfer - Get Items for Transfer](#)
 - * [Create Transfer - Get Next Transfer Numbers](#)
 - * [Create Transfer - Get To Locations for Transfer](#)

- * [Create Transfer - Refresh Transfer Items](#)
- * [Create Transfer - Refresh Transfer Locations](#)
- * [Mobile Workflow - Get Locations on Existing Transfers](#)
- * [Mobile Workflow - Get Transfer Statuses](#)
- * [Mobile Workflow - Get Transfer Summary](#)
- * [Mobile Workflow - Get Transfer Types](#)
- * [Mobile Workflow - Get Transfer Users](#)
- * [Mobile Workflow - Get Transfers](#)
- * [Mobile Workflow - Refresh Transfer Locations](#)
- * [Mobile Workflow Transfer Upload Service](#)
 - * [Create Transfer](#)
 - * [Mobile Workflow - Update Transfer Status](#)
- * [Stock Order Status Upload Service](#)
 - * [Manage Stock Order Status](#)
- * [Transfer Detail Service](#)
- * [Transfer Download Service](#)
 - * [Get Transfer](#)
 - * [Get Transfer Details for Provided Transfer](#)
- * [Transfer Upload Service](#)
 - * [Create Book Transfer](#)
 - * [Create Transfer Details](#)
 - * [Create Transfers](#)
 - * [Delete Transfer Details](#)
 - * [Delete Transfers](#)
 - * [Update Transfer Details](#)
 - * [Update Transfers](#)
- * [Work Order Status Upload Service](#)
 - * [Update Work Orders Status](#)
- * [Work Order Download Service](#)
 - * [Purchase Order Work Order Publish Services](#)
 - * [Transfer Work Order Publish Services](#)
- [Items](#)
 - [Item Definition](#)
 - * [Item AI Services](#)
 - * [Item Attribute Extraction Services](#)
 - * [Get Item Attributes Base](#)
 - * [Get Item Attributes Customized](#)
 - * [Update Item Attributes Customized](#)

- * [Item Download Service](#)
 - * [Get Item Details](#)
 - * [Get Item Details for All or Provided Items](#)
 - * [Get Item Details for Provided Item](#)
 - * [Get Item Location Details](#)
 - * [Get Item VAT Details](#)
 - * [Get Item VAT Details by Item Number](#)
- * [Item ELC Upload Service](#)
 - * [Create Item Conditional Tariff Treatments](#)
 - * [Create Item Expenses](#)
 - * [Create Item HTS](#)
 - * [Create Item HTS Assessments](#)
 - * [Delete Item Conditional Tariff Treatments](#)
 - * [Delete Item Expenses](#)
 - * [Delete Item HTS](#)
 - * [Delete Item HTS Assessments](#)
 - * [Update Item Expenses](#)
 - * [Update Item HTS](#)
 - * [Update Item HTS Assessments](#)
- * [Item Location Upload Service](#)
 - * [Create Item Locations](#)
 - * [Manage Item Location Status Update Requests](#)
 - * [Update Item Locations](#)
- * [Item Number Reservation Upload Service](#)
 - * [Generate and Reserve Item Numbers](#)
- * [Item Placeholder Upload Service](#)
 - * [Create Placeholder Items](#)
- * [Item Sourcing Upload Service](#)
 - * [Create Item Supplier Countries](#)
 - * [Create Item Supplier Countries of Manufacture](#)
 - * [Create Item Supplier Country Dimensions](#)
 - * [Create Item Supplier Country Locations](#)
 - * [Create Item Supplier Translations](#)
 - * [Create Item Suppliers](#)
 - * [Delete Item Supplier Countries](#)
 - * [Delete Item Supplier Countries of Manufacture](#)
 - * [Delete Item Supplier Country Dimensions](#)
 - * [Delete Item Supplier Country Locations](#)

- * [Delete Item Supplier Translations](#)
- * [Delete Item Suppliers](#)
- * [Update Item Supplier Countries](#)
- * [Update Item Supplier Countries of Manufacture](#)
- * [Update Item Supplier Country Dimensions](#)
- * [Update Item Supplier Country Locations](#)
- * [Update Item Supplier Translations](#)
- * [Update Item Suppliers](#)
- * [Item Taxes Upload Service](#)
 - * [Create Item Taxes](#)
 - * [Delete Item Taxes](#)
- * [Item Up Charge Download Service](#)
 - * [Get Item Up Charges for Future Date](#)
- * [Item Upload Service](#)
 - * [Create Item Description Translations](#)
 - * [Create Item Image Translations](#)
 - * [Create Item Images](#)
 - * [Create Item Seasons](#)
 - * [Create Item Tickets](#)
 - * [Create Item UDA Free Form Translations](#)
 - * [Create Item UDAs](#)
 - * [Create Item Up Charges](#)
 - * [Create Items](#)
 - * [Create Required Documents](#)
 - * [Delete Item Description Translations](#)
 - * [Delete Item Image Translations](#)
 - * [Delete Item Images](#)
 - * [Delete Item Seasons](#)
 - * [Delete Item Tickets](#)
 - * [Delete Item UDA Free Form Translations](#)
 - * [Delete Item UDAs](#)
 - * [Delete Item Up Charges](#)
 - * [Delete Items](#)
 - * [Delete Required Documents](#)
 - * [Update Item Description Translations](#)
 - * [Update Item Image Translations](#)
 - * [Update Item Images](#)
 - * [Update Item Tickets](#)

- * [Update Item UDA Free Form Translations](#)
- * [Update Item UDAs](#)
- * [Update Item Up Charge Details](#)
- * [Update Items](#)
- * [Related Items Upload Service](#)
 - * [Create Related Items](#)
 - * [Create Related Item Translations](#)
 - * [Delete Related Items](#)
 - * [Delete Related Item Translations](#)
 - * [Update Related Items](#)
 - * [Update Related Item Translations](#)
- [Item Foundation](#)
 - * [Brand Download Service](#)
 - * [Get Brands](#)
 - * [Brand Upload Service](#)
 - * [Manage Brands](#)
 - * [Differentiator Download Service](#)
 - * [Get Differentiator Details](#)
 - * [Get Differentiator Details for Provided Differentiator](#)
 - * [Get Differentiator Details for Single Differentiator](#)
 - * [Differentiator Group Download Service](#)
 - * [Get Differentiator Groups](#)
 - * [Get Differentiators for Provided Group](#)
 - * [Differentiator Group Upload Service](#)
 - * [Create Differentiator Group Details](#)
 - * [Create Differentiator Groups](#)
 - * [Delete Differentiator Group Details](#)
 - * [Delete Differentiator Groups](#)
 - * [Update Differentiator Group Details](#)
 - * [Update Differentiator Groups](#)
 - * [Differentiator Type Download Service](#)
 - * [Get Differentiator Type Details for Provided Type](#)
 - * [Get Differentiator Types](#)
 - * [Differentiator Upload Service](#)
 - * [Create Differentiators](#)
 - * [Delete Differentiators](#)
 - * [Update Differentiators](#)
 - * [Seasons and Phases Download Service](#)

- * [Get Seasons and Phases Service](#)
 - * [User Defined Attribute \(UDA\) Download Service](#)
 - * [REST Endpoint for UDA details](#)
 - * [REST Endpoint to Fetch UDA by UDA ID](#)
- [Merchandise Hierarchy](#)
 - [Class Download Service](#)
 - * [Get Class Details for Provided Class](#)
 - * [Get Classes](#)
 - [Class Upload Service](#)
 - * [Create Class](#)
 - * [Delete Class](#)
 - * [Update Class](#)
 - [Company Upload Service](#)
 - * [Update Company](#)
 - [Department Download Service](#)
 - * [Get Department Details](#)
 - * [Get Department Details for Provided Department](#)
 - * [Get Department Name](#)
 - * [Get Department Tax Details](#)
 - * [Get Departments](#)
 - [Department Upload Service](#)
 - * [Create Department](#)
 - * [Create Department Taxes](#)
 - * [Create Department Up Charge](#)
 - * [Delete Department](#)
 - * [Delete Department Up Charge](#)
 - * [Update Department](#)
 - * [Update Department Taxes](#)
 - * [Update Department Up Charge](#)
 - [Division Download Service](#)
 - * [Get Division Details for Provided Division](#)
 - * [Get Divisions](#)
 - [Division Upload Service](#)
 - * [Create Division](#)
 - * [Delete Division](#)
 - * [Update Division](#)
 - [Group Download Service](#)
 - * [Get Group for Provided Group](#)

- * [Get Groups](#)
 - [Group Upload Service](#)
 - * [Create Group](#)
 - * [Delete Group](#)
 - * [Update Group](#)
 - [Merchandise Hierarchy Download Service](#)
 - * [Get Merchandise Hierarchy](#)
 - [Subclass Download Service](#)
 - * [Get Subclass Details for Provided Subclass](#)
 - * [Get Subclasses](#)
 - [Subclass Upload Service](#)
 - * [Create Subclass](#)
 - * [Delete Subclass](#)
 - * [Update Subclass](#)
 - [Reclassification](#)
 - * [Item Reclassification Download Service](#)
 - * [Item Reclassification Upload Service](#)
 - * [Merchandise Hierarchy Reclassification Upload Service](#)
- [Organizational Hierarchy](#)
 - [Organizational Hierarchy Download Service](#)
 - * [Get Organizational Hierarchy for Provided Level](#)
 - * [Get Organizational Hierarchy for Provided Level and Hierarchy Identifier](#)
 - * [Get Organizational Hierarchy](#)
 - [Organizational Hierarchy Upload Service](#)
 - * [Create Organizational Hierarchy](#)
 - * [Delete Organizational Hierarchy](#)
 - * [Update Organizational Hierarchy](#)
 - [Stores](#)
 - * [Store Download Service](#)
 - * [Get Basic Store Details](#)
 - * [Get Store Details](#)
 - * [Get Store Details for Provided Store](#)
 - * [Store Upload Service](#)
 - * [Create Store](#)
 - * [Create Store Address](#)
 - * [Create Store Department Up Charges](#)
 - * [Create Store Hours](#)
 - * [Create Store Location Trait](#)

- * [Create Walk-Through Store](#)
 - * [Delete Store](#)
 - * [Delete Store Address](#)
 - * [Delete Store Department Up Charges](#)
 - * [Delete Store Hours](#)
 - * [Delete Store Location Trait](#)
 - * [Delete Walk-Through Store](#)
 - * [Update Store](#)
 - * [Update Store Address](#)
 - * [Update Store Department Up Charges](#)
 - * [Update Store Hours](#)
- * [Stores by Hierarchy Level Upload Service](#)
 - * [Create Store Location Trait by Hierarchy Level](#)
 - * [Delete Store Location Trait by Hierarchy Level](#)
- [Warehouse](#)
 - * [Warehouse Download Service](#)
 - * [Get Basic Warehouse Details](#)
 - * [Get Warehouse Details](#)
 - * [Get Warehouse Details for Provided Warehouse](#)
- [Purchase Orders](#)
 - [Mobile Workflow Purchase Order Download Service](#)
 - * [Create Purchase Order - Get Items for Purchase Orders](#)
 - * [Create Purchase Order - Get Locations for Purchase Order](#)
 - * [Create Purchase Order - Get Next Purchase Order Numbers](#)
 - * [Create Purchase Order - Get Purchase Order Supplier Terms](#)
 - * [Create Purchase Order - Get Suppliers for Purchase Order](#)
 - * [Create Purchase Order - Refresh Purchase Order Item Locations](#)
 - * [Create Purchase Order - Refresh Purchase Order Items](#)
 - * [Create Purchase Order - Refresh Purchase Order Suppliers](#)
 - * [Get Purchase Order Detail - Simple](#)
 - * [Manage Purchase Orders - Get Open Purchase Orders](#)
 - * [Manage Purchase Orders - Get Purchase Order Cancel Reasons](#)
 - * [Manage Purchase Orders - Get Purchase Order Header](#)
 - * [Manage Purchase Orders - Get Purchase Order Items for an Order](#)
 - * [Manage Purchase Orders - Get Purchase Order Locations for an Order/Item](#)
 - * [Manage Purchase Orders - Get Purchase Order Origins](#)
 - * [Manage Purchase Orders - Get Purchase Order Statuses](#)
 - * [Manage Purchase Orders - Get Users Who Have Created Purchase Orders](#)

- [Mobile Workflow Purchase Order Upload Service](#)
 - * [Create Purchase Order - Simple](#)
 - * [Manage Purchase Orders - Approve Purchase Orders](#)
 - * [Manage Purchase Orders - Cancel Purchase Orders](#)
 - * [Manage Purchase Orders - Reject Purchase Orders](#)
 - * [Manage Purchase Orders - Update Purchase Order Dates](#)
- [Purchase Order Download Service](#)
 - * [Get Purchase Order Details](#)
 - * [Get Purchase Order Details by Order Number](#)
- [Purchase Order Upload Service](#)
 - * [Create Pre-issued Order Numbers](#)
 - * [Create Purchase Order Details](#)
 - * [Create Purchase Order Expenses](#)
 - * [Create Purchase Order HTS](#)
 - * [Create Purchase Order HTS Assessments](#)
 - * [Create Purchase Order Letters of Credit](#)
 - * [Create Purchase Orders](#)
 - * [Delete Purchase Order Details](#)
 - * [Delete Purchase Order Expenses](#)
 - * [Delete Purchase Order HTS](#)
 - * [Delete Purchase Order HTS Assessments](#)
 - * [Delete Purchase Order Letters of Credit](#)
 - * [Delete Purchase Orders](#)
 - * [Update Purchase Order Details](#)
 - * [Update Purchase Order Expenses](#)
 - * [Update Purchase Order HTS](#)
 - * [Update Purchase Order HTS Assessments](#)
 - * [Update Purchase Order Letters of Credit](#)
 - * [Update Purchase Orders](#)
- [Replenishment](#)
 - [Inventory Request Upload Service](#)
 - * [Create Inventory Requests](#)
 - [Replenishment Scheduled Update Upload Service](#)
- [Sales](#)
 - [Sales Download Service](#)
 - * [Get Item Location Weekly Sales \(deprecated\)](#)
 - * [Get Item Location Weekly Sales](#)
 - [Store Day User Upload Service](#)

Administration

The following services are included in this functional area:

- [Data Privacy](#)
 - [Data Privacy Download Service](#)
 - * [Get Personally Identifiable Information](#)
 - [Data Privacy Upload Service](#)
 - * [Delete Personally Identifiable Information](#)
 - [Security Pre-requisites/Considerations for accessing Data Privacy ReST Services](#)
- [Operations](#)
 - [Background Job Configuration Upload Service](#)
 - * [Get Batch Run Status Services](#)
 - * [Update Background Job Configuration](#)
 - [BI Publisher Report URL Download Service](#)
 - * [Get BI Publisher Report URL](#)
 - [Calendar Download Service](#)
 - * [Get 4-5-4 Calendar](#)
 - [Codes and Descriptions Download Service](#)
 - * [Get Codes and Descriptions](#)
 - * [Get Codes and Descriptions for All or Provided Code Type](#)
 - [External Reference Service](#)
 - * [Manage External Cross Reference Identifier](#)
 - [POM End of Week Signal Download Service](#)
 - * [Get POM End of Week Signal Detail](#)
 - [Service Status Download Service](#)
 - * [Get ReST Service Status](#)
 - [Virtual Date Download Service](#)
 - * [Get Current Virtual Date in Date Format](#)
 - * [Get Current Virtual Date in Long and Date Format](#)
 - [Webhook Configuration Download Service](#)
 - [Webhook Configuration Upload Service](#)
- [System Options](#)
 - [System Options Download Service](#)
 - * [Get Functional System Options](#)
 - * [Get Inventory Movement System Options](#)
 - * [Get Procurement System Options](#)

Data Privacy

The following services are included in this functional area:

- [Data Privacy Download Service](#)
 - [Get Personally Identifiable Information](#)
- [Data Privacy Upload Service](#)
 - [Delete Personally Identifiable Information](#)
- [Security Pre-requisites/Considerations for accessing Data Privacy ReST Services](#)

Data Privacy Download Service

Get Personally Identifiable Information

This section describes the Data Privacy Access service for Merchandising and Sales Audit.

Business Overview

This query service provides access to data stored in Merchandising and Sales Audit that contains personally identifiable information.

Service Type

GET

ReST URL

```
https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace>/
RetailAppsDataPrivServicesRESTApp/rest/privatedata/<entityName>?
customer_id={entityName}::{entityType}::{entityId}::{fullName}:: {phone}::
{email}
```

Accept

- application/json
- application/xml

Query Parameters

- customer_id (required): The customer ID string containing the parameters to be used in looking up data. The format of this string is as follows:
 - {entity name}::{entity type}::{entity id}::{full name}::{phone}::{email}

Path Parameters

Parameter	Description
Entity Name	The query group type for which data is to be retrieved. The available group types for Merchandising are: • BUYER (Buyer) • MERCHANT (Merchant) • STORE (Store) • WAREHOUSE (Warehouse) • SUPPLIER (Supplier) • PARTNER (Partner) • OUTLOC (Outside Location) • ORDER_CUSTOMER (Merchandising Customer) The available group types for Sales Audit are: • EMPLOYEE (Employee) • CUSTOMER (Customer) Note: The supported entity names can also be retrieved by accessing the below URL either in the browser or through a REST client like Postman, cURL and so on: <code>https://rex.retail.<Region Name>.ocs.oraclecloud.com/ <Customer_Subnamespace/RetailAppsDataPrivServicesRESTApp/ rest/privatedata/config/access</code> Sample Response: <pre>{ "types": ["PARTNER", "OUTLOC", "WAREHOUSE", "getPersonalInfo", "EMPLOYEE", "raf", "STORE", "BUYER", "CUSTOMER", "SUPPLIER", "MERCHANT", "ORDER_CUSTOMER"] }</pre>

In the above response, `getPersonalInfo` and `raf` are only for internal consumption and not to be used. The ones that are permitted are the other entity names.

Parameter	Description
Entity Type	In case of Merchandising, this parameter is used if the entity name is PARTNER or OUTLOC. The value here should indicate the type of partner or outside location being queried. Valid values for this input can be found on the Codes table for each type: - Partner Types = PTNR - Outside Location Types = LOCT In case of Sales Audit, this parameter is used if the entity name is CUSTOMER. The value here should indicate the type of customer being queried. Valid values for this input can be found on the Codes table where code type is CIDD. (Customer Types)
Entity ID	The ID of the entity being queried. For example, the supplier ID (in Merchandising) or employee ID (in Sales Audit).
Full Name	The full name to be searched for the selected entity.
Phone	The phone number to be searched for the selected entity.
Email	The email to be searched for the selected entity.

Default Response

The response will return all instances of the data being searched that occur in the requested entity. For example, if the entity requested was BUYER, all instances where the buyer, name, and phone match the data sent will be returned. If any of these parameters are not sent (for example, email), then it will not be used as part of the search. The following data is included in the response:

Parameter	Description
Entity Name	For Merchandising: - BUYER (Buyer) - MERCHANT (Merchant) - STORE (Store) - WAREHOUSE (Warehouse) - SUPPLIER (Supplier) - PARTNER (Partner) - OUTLOC (Outside Location) - ORDER CUSTOMER (Merchandising Customer) For Sales Audit: - EMPLOYEE (Employee) - CUSTOMER (Customer)
Entity Type	In the case of Merchandising, if the entity name is PARTNER or OUTLOC, the value here indicates the type of partner or outside location. Valid values can be found on the Codes table for each type: - Partner Types = PTNR - Outside Location Types = LOCT For other entity types, this will be null. In the case of Sales Audit, if the entity name is CUSTOMER, the value here indicates the type of customer. Valid values can be found on the Codes table where code type is CIDD. For other entity types, this will be null.
Entity ID	The ID of the entity where the data was found.
Full Name	The name associated with the entity.
Phone	The phone number associated with the entity.

Parameter	Description
Fax	The fax number associated with the entity.
Telex	The telex number associated with the entity.
Pager	The pager number associated with the entity.
Email	The email address associated with the entity.

Sample Response

```
{
  "Buyer Information": {
    "list": [],
    "Get Buyer Information": {
      "list": [
        [
          {
            "ENTITY_NAME": "BUYER",
            "ENTITY_TYPE": "null",
            "ENTITY_ID": "1002",
            "FULL_NAME": "Matt Wilsman",
            "PHONE": "6125251034",
            "FAX": "6125259800",
            "TELEX": "null",
            "PAGER": "null",
            "EMAIL": "test.abd.com"
          }
        ]
      ]
    }
  }
}
```

Response Codes and Error Messages

- 200 - Success
- 400 - Bad Request - for the following situations:
 - Customer ID does not match the required format.
 - Invalid input type
 - Missing customer ID
 - Invalid JSON Format
- 500 - Internal Server Errors - for all other types of errors (for example, configuration errors, SQL errors, and so on).

Success Payloads

- When Accept=application/json, this API will return data in JSON format
- When Accept=application/xml, this API will return data formatted as an HTML page

The below table gives a sample of URLs for invoking the GET service for the different entity names.

Note

- The entity ID should be replaced with the appropriate valid ID based on the customer environment.
- The sample response formats shown here are with using a REST client like PostMan and with the Header Accept value set to application/json

Entity Name	Reference URL	Sample Response
BUYER	<code>https:// rex.retail.<Region Name>.ocs.oraclec loud.com/ <Customer Subnamespace/ RetailAppsDataPri vServicesRESTApp/ rest/privatedata/ BUYER? customer_id=BUYER ::3933:::</code>	<pre>{ "Buyer Information": { "list": [], "Get Buyer Information": { "list": [[{ "ENTITY_NAME": "BUYER", "ENTITY_TYPE": "null", "ENTITY_ID": "3933", "FULL_NAME": "RMS_Buyer", "PHONE": "999-888-7777", "FAX": "666-555-4444", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }]] } } }</pre>

Entity Name	Reference URL	Sample Response
MERCHANT	https:// rex.retail.<Region Name>.ocs.oraclecl oud.com/<Customer Subnamespace / RetailAppsDataPriv ServicesRESTApp/ rest/privatedata/ MERCHANT? customer_id=MERCHA NT::2114:::	{ "Merchant Information": { "list": [], "Get Merchant Information": { "list": [{ "ENTITY_NAME": "MERCHANT", "ENTITY_TYPE": "null", "ENTITY_ID": "2114", "FULL_NAME": "DC Merchandise_2114", "PHONE": "999-666-5555", "FAX": "999-888-2222", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }] } } }

Entity Name	Reference URL	Sample Response
STORE	<code>https:// rex.retail.<Region Name>.ocs.oraclec loud.com/ <Customer Subnamespace/ RetailAppsDataPri vServicesRESTApp/ rest/privatedata/ STORE? customer_id=STORE ::7143876752:::: :</code>	<pre>{ "Store Information": { "list": [], "Get Store Information": { "list": [{ "ENTITY_NAME": "STORE", "ENTITY_TYPE": "null", "ENTITY_ID": "7143876752", "FULL_NAME": "Scott", "PHONE": "763-888-4455", "FAX": "1-763-959-4477", "TELEX": "null", "PAGER": "null", "EMAIL": "Oracle@Automail.com" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "50003", "FULL_NAME": "Oracle", "PHONE": "111", "FAX": "111", "TELEX": "null", "PAGER": "null", "EMAIL": "John.Doe@Oracle.com" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "50004", "FULL_NAME": "Oracle", "PHONE": "111", "FAX": "111", "TELEX": "null", "PAGER": "null", "EMAIL": "John.Doe@Oracle.com" }] } } }</pre>

Entity Name	Reference URL	Sample Response
WAREHOUSE		
E	<code>https:// rex.retail.<Region >.ocs.oraclecloud.com/ <Customer Subnamespace / RetailAppsDataPriv ServicesRESTApp/ rest/privatedata/ WAREHOUSE? customer_id=WAREH OUSE::9993::</code>	<pre>{ "Warehouse Information": { "list": [], "Get Warehouse Information": { "list": [{ "ENTITY_NAME": "WAREHOUSE", "ENTITY_TYPE": "null", "ENTITY_ID": "9993", "FULL_NAME": "null", "PHONE": "null", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "Eric@manhattan.com" }] } } }</pre>

Entity Name	Reference URL	Sample Response
SUPPLIER	<a href="https://rex.retail.<Region>.ocs.oraclecloud.com/<CustomerSubnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/SUPPLIER?customer_id=SUPPLIER::2001:::">https://rex.retail.<Region>.ocs.oraclecloud.com/<CustomerSubnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/SUPPLIER?customer_id=SUPPLIER::2001::::	<pre>{ "Supplier Information": { "list": [], "Get Supplier Information": { "list": [[{ "ENTITY_NAME": "SUPPLIER", "ENTITY_TYPE": "null", "ENTITY_ID": "2001", "FULL_NAME": "James Moriarty", "PHONE": "6128059160", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "105001", "FULL_NAME": "null", "PHONE": "null", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "105002", "FULL_NAME": "null", "PHONE": "null", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "105003", "FULL_NAME": "null", "PHONE": "null", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }]] } } }</pre>

Entity Name	Reference URL	Sample Response
		<pre>"ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "105004", "FULL_NAME": "null", "PHONE": "null", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }] } }</pre>

Entity Name	Reference URL	Sample Response
PARTNER	<a href="https://rex.retail.<Region>.ocs.oraclecloud.com/<CustomerSubnamespace>/RetailAppsDataPrivServicesRESTApp/rest/privatedata/PARTNER?customer_id=PARTNER::::464694358::::">https://rex.retail.<Region>.ocs.oraclecloud.com/<CustomerSubnamespace>/RetailAppsDataPrivServicesRESTApp/rest/privatedata/PARTNER?customer_id=PARTNER::::464694358::::	<pre>{ "Partner Information": { "list": [], "Get Partner Information": { "list": [{ "ENTITY_NAME": "PARTNER", "ENTITY_TYPE": "BK", "ENTITY_ID": "464694358", "FULL_NAME": "XXXXX", "PHONE": "XXXXX", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "270001", "FULL_NAME": "B", "PHONE": "123-654-8888", "FAX": "123-654-9999", "TELEX": "null", "PAGER": "null", "EMAIL": "oracle@automail.com" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "270002", "FULL_NAME": "P", "PHONE": "123-654-8888", "FAX": "123-654-9999", "TELEX": "null", "PAGER": "null", "EMAIL": "oracle@automail.com" }, { "ENTITY_NAME": "ADDRESS", "ENTITY_TYPE": "null", "ENTITY_ID": "270003", "FULL_NAME": "null", "PHONE": "null", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }] } } }</pre>

Entity Name	Reference URL	Sample Response
		<pre>] } } }</pre>
OUTLOC	<pre>https:// rex.retail.<Region Name>.ocs.oraclec loud.com/ <Customer Subnamespace / RetailAppsDataPri vServicesRESTApp/ rest/privatedata/ OUTLOC? customer_id=OUTLO C:::97285:::</pre>	<pre>{ "Out Loc Information": { "list": [], "Get OutLoc Information": { "list": [{ "ENTITY_NAME": "OUTLOC", "ENTITY_TYPE": "RL", "ENTITY_ID": "97285", "FULL_NAME": "Alexandar Hamilton", "PHONE": "56565-22-22", "FAX": "56565-22-22", "TELEX": "56565-22-22", "PAGER": "null", "EMAIL": "alloy.automation@oracle.com" }] } } }</pre>

Entity Name	Reference URL	Sample Response
ORDER_CUSTOMER	https:// rex.retail.<Region Name>.ocs.oraclecloud.com/ <Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/ rest/privatedata/ ORDER_CUSTOMER? customer_id=ORDER_CUSTOMER::90003: :::	{ "Order Customer Information": { "list": [], "Order Customer Information": { "list": [{ "ENTITY_NAME": "ORDER_CUSTOMER", "ENTITY_TYPE": "null", "ENTITY_ID": "90003", "FULL_NAME": "John Scott", "PHONE": "1800800800", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "null" }] } } }
EMPLOYEE	https:// rex.retail.<Region Name>.ocs.oraclecloud.com/ <Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/ rest/privatedata/ EMPLOYEE? customer_id=EMPLOYEE::SAMMYSMITH: :::	{ "Employee Information": { "list": [], "Employee Information": { "list": [{ "ENTITY_NAME": "EMPLOYEE", "ENTITY_TYPE": "null", "ENTITY_ID": "SAMMYSMITH", "FULL_NAME": "PS", "PHONE": "990-888-8989", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "PS@GMAIL.COM" }] } } }

Entity Name	Reference URL	Sample Response
CUSTOMER	<code>https:// rex.retail.<Region >.ocs.oraclecloud.com/ <Customer Subnamespace/ RetailAppsDataPriv ServicesRESTApp/ rest/privatedata/ CUSTOMER? customer_id=CUSTO MER::CUSTID::1234 567:::~::~:</code>	<pre>{ "Customer Information": { "list": [], "Customer Information": { "list": [{ "ENTITY_NAME": "CUSTOMER", "ENTITY_TYPE": "CUSTID", "ENTITY_ID": "1234567", "FULL_NAME": "alice", "PHONE": "876543", "FAX": "null", "TELEX": "null", "PAGER": "null", "EMAIL": "lice@gmail.com" }] } } }</pre>

Data Privacy Upload Service

Delete Personally Identifiable Information

This section describes the Data Privacy Forget service for Merchandising and Sales Audit.

Business Overview

This service supports updating personal information stored in Merchandising and Sales Audit. When the service is invoked with mask strings as inputs, it overwrites the fields with mask strings, which effectively removes the personal information from the system.

Service Type

DELETE

ReST URL

`https://<host:port>/RetailAppsDataPrivServicesRESTApp/rest/privatedata/
<updateEntityName>?customer_id={entityName}::{entityType}::{entityId}::
{fullName}:: {phone}::{fax}::{telex}::{pager}::{email}::{addr1}::{addr2}::
{addr3}::{county}::{city}::{state}::{countryId}::{postalCode}`

Accept

- application/json
- application/xml

Query Parameters

- customer_id (required): The customer ID string containing the parameters to be used in updating data. The format of this string is as follows:
 - {entityName}::{entityType}::{entityId}::{fullName}::{phone}::{fax}::{telex}::{pager}::{email}::{addr1}::{addr2}::{addr3}::{county}::{city}::{state}::{countryId}::{postalCode}

Path Parameters

Parameter	Description
Update Entity Name	The update group type for which data is to be updated. The available group types for Merchandising are: - updateAddressInformation - updateOrderCustomerInformation - updateSupplierInformation - updateOutLocInformation - updateBuyerInformation - updateMerchantInformation - updateStoreInformation - updateWarehouseInformation - updatePartnerInformation The available group types for Sales Audit are: - updateEmployeeInformation - updateCustomerInformation Note: To view the supported forget PII data types, the below URL can be invoked with a GET request <a href="https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/config/forget">https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/config/forget
Entity Name (required)	The group type for which data is to be updated. The available group types for Merchandising are: - BUYER (Buyer) - MERCHANT (Merchant) - STORE (Store) - WAREHOUSE (Warehouse) - SUPPLIER (Supplier) - PARTNER (Partner) - OUTLOC (Outside Location) - ORDER CUSTOMER (Merchandising Customer) The available group types for Sales Audit are: - EMPLOYEE (Employee) - CUSTOMER (Customer)
Entity Type	In case of Merchandising, this parameter is used if the entity name is PARTNER or OUTLOC. The value here should indicate the type of partner or outside location. Valid values for this input can be found on the Codes table for each type: - Partner Types = PTNR - Outside Location Types = LOCT In case of Sales Audit, this parameter is used if the entity name is CUSTOMER. The value here should indicate the type of customer. Valid values for this input can be found on the Codes table where code type is CIDT (Customer Types).
Entity ID (required)	The ID of the entity to be updated. For example, the supplier ID (in Merchandising) or employee ID (in Sales Audit).

Parameter	Description
Full Name	The value to update the full name with. If the value is null and this is a required field in the entity, XXXXX will be used.
Phone	The value to update the phone number with. If the value is null and this is a required field in the entity, XXXXX will be used.
Fax	The value to update the fax number with.
Telex	The value to update the telex number with.
Pager	The value to update the pager number with.
Email	The value to update the email address with.
Addr1	The value to update the address 1 with.
Addr2	The value to update the address 2 with.
Addr3	The value to update the address 3 with.
County	The value to update the county with.
City	The value to update the city with.
State	The value to update the state with.
Country	The value to update the country with.
Postal Code	The value to update the postal code with.

Default Response

This service only returns a response code to signify if the request is successful or not. If no record is updated, the service returns an error.

Response Codes and Error Messages

- 200 - Success
- 400 - Bad Request - for the following situations:
 - Customer ID does not match the required format
 - Invalid input type
 - Missing customer ID
 - Invalid JSON Format
- 500 - Internal Server Errors - for all other types of errors (for example, configuration errors, SQL errors, and so on).

Success Payloads

N/A

The below section depicts the sample URL for Delete API for PII (Personally Identifiable Information) for the various entity names.

Entity Name	Reference URL
BUYER	<a href="https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateBuyerInformation?customer_id=BUYER::3933:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateBuyerInformation?customer_id=BUYER::3933:::
SUPPLIER	<a href="https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateSupplierInformation?customer_id=SUPPLIER::2001:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateSupplierInformation?customer_id=SUPPLIER::2001:::

Entity Name	Reference URL
MERCHANT	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateMerchantInformation? customer_id=MERCHANT::2114:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateMerchantInformation? customer_id=MERCHANT::2114:::
STORE	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/ RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateStoreInformation? customer_id=STORE::7143876752:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/ RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateStoreInformation? customer_id=STORE::7143876752:::
WAREHOUSE	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateWarehouseInformation? customer_id=WAREHOUSE::9993:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateWarehouseInformation? customer_id=WAREHOUSE::9993:::
PARTNER	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updatePartnerInformation? customer_id=PARTNER:::464694358:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updatePartnerInformation? customer_id=PARTNER:::464694358:::
OUTLOC	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateOutLocInformation? customer_id=OUTLOC:::97285:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateOutLocInformation? customer_id=OUTLOC:::97285:::
CUSTOMER	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/ RetailAppsDataPrivServicesRESTApp/rest/privatedata/CUSTOMER? customer_id=CUSTOMER::CUSTID::1234567:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace/ RetailAppsDataPrivServicesRESTApp/rest/privatedata/CUSTOMER? customer_id=CUSTOMER::CUSTID::1234567:::
ORDER_CUST OMER	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/ updateOrderCustomerInformation?customer_id=ORDER_CUSTOMER::90003:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/ updateOrderCustomerInformation?customer_id=ORDER_CUSTOMER::90003:::
EMPLOYEE	<a href="https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateEmployeeInformation? customer_id=EMPLOYEE::SAMMYSMITH:::">https:// rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace / RetailAppsDataPrivServicesRESTApp/rest/privatedata/updateEmployeeInformation? customer_id=EMPLOYEE::SAMMYSMITH:::

Security Pre-requisites/Considerations for accessing Data Privacy ReST Services

The preferred access method for these services is through OAuth2–based security. For OAuth credential configuration, see the section titled [OAuth for REST Service Authentication](#) in this document.

In the rare event of using basic Auth for accessing these services, ensure the user accessing these services has the necessary IDCS groups associated as listed below:

- For Production environments,
 - DATAPRIV_ADMINISTRATOR_REST_API_ROLE
 - DATA_PRIVACY_ADMINISTRATOR_JOB
- For Non-Prod environments,
 - DATAPRIV_ADMINISTRATOR_REST_API_ROLE_PREPROD
 - DATA_PRIVACY_ADMINISTRATOR_JOB_PREPROD

Operations

The following services are included in this functional area:

- [Background Job Configuration Upload Service](#)
 - [Get Batch Run Status Services](#)
 - [Update Background Job Configuration](#)
- [BI Publisher Report URL Download Service](#)

- [Get BI Publisher Report URL](#)
- [Calendar Download Service](#)
 - [Get 4-5-4 Calendar](#)
- [Codes and Descriptions Download Service](#)
 - [Get Codes and Descriptions](#)
 - [Get Codes and Descriptions for All or Provided Code Type](#)
- [External Reference Service](#)
 - [Manage External Cross Reference Identifier](#)
- [POM End of Week Signal Download Service](#)
 - [Get POM End of Week Signal Detail](#)
- [Service Status Download Service](#)
 - [Get ReST Service Status](#)
- [Virtual Date Download Service](#)
 - [Get Current Virtual Date in Date Format](#)
 - [Get Current Virtual Date in Long and Date Format](#)
- [Webhook Configuration Download Service](#)
- [Webhook Configuration Upload Service](#)

Upload RDE Batch Status for Business Date

Functional Area

Administration - Operations

Business Overview

This service can be used by RDE to send job status to MFCS. It inserts a record in RMS_RDE_BATCH_STATUS with job name, business date and status from the payload.

When the service is called with job name = 'RDE_COMPLETION', the relevant business date, and status = 'C' (completed), it serves as a signal to MFCS that RDE has successfully extracted data for a specific business date.

Service Type

POST

ReST URL

MerchIntegrations/services/private/administration/operations/rdeBatchDependency/create

Input Payload Details

Create - Object. See list of elements for detail

Sample Input Message

```
{
  "jobName": "RDE_COMPLETION",
  "businessDate": "2001-12-31",
```

```
"status": "C"
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Background Job Configuration Upload Service

The following services are included in this functional area:

- [Get Batch Run Status Services](#)
- [Update Background Job Configuration](#)

Get Batch Run Status Services

Endpoints

/MerchIntegrations/services/administration/operations/batchRunStatus

Functional Area

Administration - Operations

Business Overview

This service returns the current running status of nightly batch. During the nightly batch window, certain REST services may be unavailable either for the entire duration or for a portion of the night, especially when inventory or costing batch flows are in progress. This API enables clients to monitor the status of batch processes, allowing them to optimize their service calls by pausing or resuming dependent REST services accordingly.

Service Type

GET

ReST URL

/MerchIntegrations/services/administration/operations/batchRunStatus

Response Code: 200 (Success)

Table 5-62 BatchStatus - Object. See list of elements for detail

Element Name	Required	Data Type	Description
batchRunningInd	Yes	String	This field indicates if Nightly batch jobs are running. It is set to 'Y' at the beginning of the RMS nightly batch cycle and set to 'N' at the end of the cycle.
inventoryBatchRunningInd	Yes	String	This field indicates whether inventory batch jobs are running. Valid values are Y or N.
costingBatchRunningInd	Yes	String	This field indicates whether costing batch jobs are running. Valid values are Y or N.
batchStartWarningInd	Yes	String	Indicates whether the nightly batch is about to start, allowing clients to pause operations that are not permitted during the batch window. Valid values are Y or N.

Sample Response Message

```
{
  "batchRunningInd": "Y",
  "inventoryBatchRunningInd": "N",
  "costingBatchRunningInd": "N",
  "batchStartWarningInd": "N"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

```

    }
  }

```

Update Background Job Configuration

This section describes the Background Process Configuration.

Business Overview

This service is used to update the configuration for each background jobs in Merchandising.

Service Type

Post

ReST URL

processes/update/process_config/execution

Input Parameters

Parameter Name	Required	Description
JobName	Yes	Job Name
numThreads	No	Maximum number of threads the job will execute
numDataToProcess	No	Number of records a jobs will process each run.
commitMaxCtr	No	Max number of records processed before a commit is issued.
archiveInd	No	This field will be used to determine if associated tables for this job needs to be archived to history or not.

Output

N/A

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
B8D_PROCESS_CONFIG	No	No	Yes	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

BI Publisher Report URL Download Service

The following services are included in this functional area:

- [Get BI Publisher Report URL](#)

Get BI Publisher Report URL

Functional Area

Administration - Operations

Business Overview

This service is used by Oracle Retail Financial Integration (RFI) to retrieve the URL of a BI Publisher report from Merchandising or Sales Audit that can be invoked from the PeopleSoft

Financials General Ledger based on a particular journal entry. The report URL that will be returned will be different based on the ID sent in the service call. Based on that ID, Merchandising will determine if it was a Merchandising, Sales Audit, or Invoice Matching ID and return a URL for the appropriate report.

The possible reports for Merchandising and Sales Audit are:

- GL Fixed Deal Data Report
- GL Item level Data Report
- GL Item Rollup Daily Data Report
- GL Item Rollup Monthly Data Report
- GL Sales Audit Data Report

Also, for Invoice Matching, one of the following reports might be returned:

- Merchandise Invoice Document Report
- Non-Merchandise Invoice Document Report
- Credit Note Document Report
- Credit Memo Cost Document Report
- Credit Memo Quantity Document Report
- Debit Memo Cost Document Report
- Debit Memo Quantity Document Report
- Debit Memo VAT Document Report
- Receipt Write Off Document Report

For cloud service implementations, this configuration should be done if you are working with PeopleSoft Financials. For on premise implementations, you may need to configure this in the RETAIL_SERVICE_REPORT_URL table. For cloud service implementations, this configuration can be done by the Oracle Cloud Operations team.

The report is similar functionality to the Drill Forward and Drill Back functionality available in the Merchandising Transaction Data and Fixed Deal pages and the Sales Audit General Ledger Transaction page. See also Get Drill Back Forward URL Service for more on the APIs that support this functionality.

Service Type

POST

ReST URL

MerchIntegrations/services/administration/operations/biPublisherReport/url/get

Input Payload Details

Table 5-63 Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reportReferenceKey	Yes	String (32)	Holds the unique reference trace ID that is used to match to the report URL.

Sample Input Message

```
{
  "reportReferenceKey": "String"
}
```

Response Code: 200 (Success)

Table 5-64 GetResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reportReferenceKey	Yes	String (32)	Holds the unique reference trace ID that is used to match to the report URL.
reportUrl	Yes	String (256)	Contains the report URL with parameters for financial integration.

Sample Response Message

```
{
  "reportReferenceKey": "String",
  "reportUrl": "String"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Calendar Download Service

The following services are included in this functional area:

- [Get 4-5-4 Calendar](#)

Get 4-5-4 Calendar

Endpoints

/MerchIntegrations/services/administration/operations/calendar

Functional Area

Administration – Operations

Business Overview

This service retrieves 4-5-4 calendar details. This service takes an optional query parameter of year to retrieve details for that year

Service Type

GET

ReST URL

MerchIntegrations/services/administration/operations/calendar

Input Parameters

Parameter Name	Required	Data Type	Description
year	No	String (4)	Year

Response Code: 200 (Success)

Table 5-65 Calendar - Object. See list of elements for detail

Element Name	Required	Data Type	Description
year	Yes	Number	Contains the year. The 4-5-4 year begins on the first day of the first 4-5-4 month regardless of when the calendar year begins.
month	Yes	Number	Contains the number ranging from 1 to 12 which indicates the 4-5-4 month.
noOfWeeks	Yes	Number	Indicates whether there are 4 or 5 calendar weeks in the 4-5-4 month.
firstDay	Yes	date	Contains the first day of the 4-5-4 month in YYYY-MM-DD format

Sample Response Message

```
[
{
  "year": 2020,
  "month": 2,
  "noOfWeeks": 4,
```

```
"firstDay": "2001-12-31"
}
]
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Codes and Descriptions Download Service

The following services are included in this functional area:

- [Get Codes and Descriptions](#)
- [Get Codes and Descriptions for All or Provided Code Type](#)

Get Codes and Descriptions**Business Overview**

Code Detail service allows user to retrieve code details for a selected code and code type.

Service Type

GET

ReST URL

CodeDetail/codeDetails?code={ }&codeType={ }

Input Parameters

Parameter Name	Required	Description	Valid values
Code	Yes	Code	NA
Code Type	Yes	Code Type	NA

Output

RestCodeDetailRecRDO

Parameter Name	Data Type
codeDesc	String
requiredInd	String
codeSeq	BigDecimal
codeType	String
codeTypeDesc	String
code	String

JSON Structure

```
{
  "codeDesc": null,
  "requiredInd": null,
  "codeSeq": null,
  "codeType": null,
  "codeTypeDesc": null,
  "code": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Codes and Descriptions for All or Provided Code Type**Endpoints**

/MerchIntegrations/services/administration/operations/codes

Functional Area

Administration – Operations

Business Overview

This service returns full dataset of code and code detail. This service supports pagination and the page size of each response is controlled by the query paramter 'limit'. This service optionally takes code type as input for returning back the code details for a single code type.

Table 5-66 Input Parameters

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.

Table 5-66 (Cont.) Input Parameters

Parameter Name	Required	Data Type	Description
offsetkey	No	String	Offset key (Code Type)
codeType	No	String	Code Type
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URL

MerchIntegrations/services/administration/operations/codes

Input Payload Details

Table 5-67 Input Parameters for MerchIntegrations/services/administration/operations/codes

Parameter Name	Required	Data Type	Description
offsetkey	No	String	Offset Key. Valid Value is code type
limit	No	Number	Pagination limit. Default Value is 1000.
codeType	No	String	Code Type. Searches for a specific code type

Response Code: 200 (Success)

Table 5-68 PageResultItemsCode - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch

Table 5-68 (Cont.) PageResultItemsCode - Object. See list of elements for detail

Element Name	Required	Data Type	Description
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-69 Code - Object. See list of elements for detail

Element Name	Required	Data Type	Description
codeType	No	String	This field contains the code type which will serve as a grouping mechanism for the multiple codes.
description	No	String	This field will contain the description of the code type.
details	No	Collection of Object	List of Code against this codeType

Table 5-70 CodeDetail - Object. See list of elements for detail

Element Name	Required	Data Type	Description
code	No	String (6)	This field contains the code used in the application.
codeDescription	No	String	This field contains the description associated with the code and code type.
sequence	No	String	This is a number used to order the elements so that they appear consistently when using to populate a list or display in user interface.
usedInd	No	String	Indicate if the code is in use or not. Records set to N will not display as possible options for users.

Table 5-71 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-71 (Cont.) LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "codeType": "LOC",
      "description": "Location Types",
      "details": [
        {
          "code": "S",
          "codeDescription": "Store",
          "sequence": "1",
          "usedInd": "Y"
        }
      ]
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?limit=1000",
      "rel": "self"
    }
  ]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No

External Reference Service

The following services are included in this functional area:

- [Manage External Cross Reference Identifier](#)

Manage External Cross Reference Identifier

Functional Area

Administration - Operations

Business Overview

This is a helper service to bulk update external reference id column of few selected tables that can be used during migration activity, like financial integration migration. This only supports the following entities - Supplier, Partner, Payment Terms, Freight Terms, Organization Unit, Language and State. This service allows the client to publish Merchandising internal entity id, optional additional sub-identifier key, and its corresponding external reference id and this service will update the column external_ref_id column in the table. This service does basic validation that the merchandising entity id is valid and exists in the corresponding table. For the external reference id field, there is no validation and it expects the caller to pass in the correct value. If the external reference id is left blank in the request, it will be nulled out in the table.

Service Type

PUT

ReST URL

MerchIntegrations/services/administration/operations/externalCrossReference/
manage

Input Payload Details

Manage - Object. See list of elements for detail

Items - Object. See list of elements for detail

ManageError - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

ManageError - Object. See list of elements for detail

POM End of Week Signal Download Service

The following services are included in this functional area:

- [Get POM End of Week Signal Detail](#)

Get POM End of Week Signal Detail

Business Overview

This service is used to determine whether the Vdate is an End of Week Date or Not.

If Vdate is at EOW, it returns 'Y', else 'N'.

Service Type

Get

ReST URL

EowRunSignal/EowRunSignalDetail

Input Parameters

N/A

Output

RestEowRDO

Parameter Name	Data Type
eow	Char('Y' or 'N')

JSON Structure

```
{
  "eow": "N",
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PERIOD	Yes	No	No	No
SYSTEM_VARIABLES	Yes	No	No	No

Service Status Download Service

Get ReST Service Status

Functional Area

Administration - Operations

Business Overview

This service allows the querying of the status of ReST web service call using an X-Correlation-ID. The correlation id that was included in the request HTTP header can be used as the input, or if one was not passed one in the original request, the one from the response HTTP header can be used as the input. This service will return all the entries in service_metrics table against that correlation ID. The response support pagination and if there are multiple pages of response, use the next link to call subsequent pages. This service additionally checks the input correlation ID against the request_id field to handle the scenario where the x_correlation_id is null in the service_metrics table because it was not included in request HTTP header.

An additional query parameter of includePayload, supports a value of yes ('Y') or no ('N') can be used to retrieve the request and response payload for that request. In the case of a service error (response code of 300 or greater) the response payload will be included irrespective of includePayload parameter.

Service Metrics

Merchandising ReST Service framework keeps a log of every service call in the service_metrics table. When the service request is received at the application server, it makes an entry to service_metrics table using a unique request ID and the time of the call. Once the service request has been processed and the response is ready to be served back, the

application server updates the service metrics table to capture the run time and response code.

The application server optionally saves the payload in `service_payload` table. The payloads are only captured for POST, PUT or DELETE service calls. The payload for GET service calls are captured if service call had an error (response code is 300 or greater) or while running in debugging mode.

The service metrics and payload entry are held for short period and get purged after configurable (default 3 months) purge window.

X-Correlation-ID

Merchandising ReST Service framework supports a custom HTTP Header named as X-Correlation-ID and it is recommended to include this in every ReST web service call for traceability purpose.

The request X-Correlation-ID is copied and is returned in the Response HTTP Header X-Correlation-ID field. This is also captured in the `service_metrics` table against that request.

In the case where, X-Correlation-ID was not included in the request header, the internally generated unique request ID is populated in the Response HTTP Header X-Correlation-ID field. The response X-Correlation-ID should be logged if one is not passed in the request.

It is recommended that a unique X-Correlation-ID is always passed as part of every request (including retry) to trace each unique call. The validation for unique value is not enforced.

A log of X-Correlation-IDs should be maintained for traceability purpose and troubleshooting exceptions.

Service Metrics

- Merchandising ReST Service framework keeps a log of every service call in the `service_metrics` table. When the service request is received at the application server, it makes an entry to `service_metrics` table using a unique request ID and the time of the call. Once the service request has been processed and the response is ready to be served back, the application server updates the service metrics table to capture the run time and response code.
- The application server optionally saves the payload in `service_payload` table. The payloads are only captured for POST, PUT or DELETE service calls. The payload for GET service calls are captured if service call had an error (response code is 300 or greater) or while running in debugging mode.
- The service metrics and payload entry are held for short period and get purged after configurable (default 3 months) purge window.

X-Correlation-ID

- Merchandising ReST Service framework supports a custom HTTP Header named as X-Correlation-ID and it is recommended to include this in every ReST web service call for traceability purpose.
- The request X-Correlation-ID is copied and is returned in the Response HTTP Header X-Correlation-ID field. This is also captured in the `service_metrics` table against that request.
- In the case where, X-Correlation-ID was not included in the request header, the internally generated unique request ID is populated in the Response HTTP Header X-Correlation-ID field. The response X-Correlation-ID should be logged if one is not passed in the request.
- It is recommended that a unique X-Correlation-ID is always passed as part of every request (including retry) to trace each unique call. The validation for unique value is not enforced.

- A log of X-Correlation-IDs should be maintained for traceability purpose and troubleshooting exceptions.

Service Type

GET

ReST URL

MerchIntegrations/services/administration/operations/restService/status

Input Parameters

Parameter Name	Required	Data Type	Description
xCorrelationId	Yes	String	X-Correlation-ID
includePayload	No	String	Include Payload Y/N. Default value is N.
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Response Code: 200 (Success)

Table 5-72 PageResultItemsServiceMetrics - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-73 ServiceMetrics - Object. See list of elements for detail

Element Name	Required	Data Type	Description
requestId	Yes	String (36)	This field contains unique Request id.
xCorrelationId	No	String (36)	This field contains the X-Correlation_ID in the response http header. If the client passed a X-Correlation_ID in the request HTTP Header, the same is returned or else a generated UUID (captured in request_id) is returned back
method	Yes	String (10)	Contains the HTTP Method like GET, PUT, POST, DELETE.
path	Yes	String (255)	Contains the Rest Service path.
responseCode	No	String (10)	This field contains the HTTP Code for the response. This will hold a value after the server processed the request and responded. Example 200, 400, 500
requestTimestamp	Yes	dateTime	This field hold the timestamp in UTC when the request was received by the server.
responseTimestamp	No	dateTime	This field hold the timestamp in UTC when the request processing completes and server is ready to transmit the response back.
durationMillisecond	No	Number	This field contains the millisecond to process the request. This is populated after server completes the request.
requestSizeByte	No	Number	This field contains the bytes in request payload.
responseSizeByte	No	Number	This field contains the bytes in the response payload. This is populated after server completes the request.

Table 5-73 (Cont.) ServiceMetrics - Object. See list of elements for detail

Element Name	Required	Data Type	Description
clientName	Yes	String (255)	This field contains the OAuth Client name used to make the request
serviceUrl	Yes	String (2000)	This field the complete path for the service call
requestPayload	No	String	This field contains the request payload if query parameter of includePayload is set to Y.
responsePayload	No	String	This field contains the response payload if query parameter of includePayload is set to Y or there was an exception in the request and the response code is 300 or greater.

Table 5-74 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "requestId": "6ccc5287-a078-4aae-b697-9ade58d70bd1",
      "xCorrelationId": "034b9672-4818-40b4-916d-210a21e0de84",
      "method": "POST",
      "path": "/MerchIntegrations/services/foundation/item",
      "responseCode": "200",
      "requestTimestamp": "2001-12-31T23:59:59.000Z",
      "responseTimestamp": "2001-12-31T23:59:59.000Z",
      "durationMillisecond": 450,
      "requestSizeByte": 720,

```

```
"responseSizeByte": 18240,
"clientName": "xocs_client",
"serviceUrl": "https://<server>/MerchIntegrations/services/administration/
operations/service/status?xCorrelationId=36c6b984-8e16-4efb-
b73f-5984357c1d72",
"requestPayload": "{}",
"responsePayload": "{}"
}
],
"hasMore": true,
"limit": 1,
"count": 1,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?limit=1000",
"rel": "self"
}
]
}
```

Virtual Date Download Service

The following services are included in this functional area:

- [Get Current Virtual Date in Date Format](#)
- [Get Current Virtual Date in Long and Date Format](#)

Get Current Virtual Date in Date Format

Business Overview

Deprecated in v23.1.401.0: Scheduled for Removal in 2 Years. Use RmsReSTServices/services/private/Common/vDate

This service retrieves the Merchandising virtual business date (vdate).

Service Type

GET

ReST URL

Vdate/vdateDetail

Input Parameters

N/A

Output

Table 5-75 RestVdateRecRDO

Parameter Name	Data Type
vdate	Date

JSON Structure

```
[
  {
    "vdate": null,
    "links": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PERIOD	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYL OAD	No	Yes	No	No

Get Current Virtual Date in Long and Date Format

Business Overview

Retrieve Merchandising Vdate.

Service Type

GET

ReST URL

/Common/vDate

Input Parameters

N/A

Output

Table 5-76 Vdate in Long and Date Format

Parameter Name	Data Type
Vdate	Long
Vdate	Date

JSON Structure

```
{
  "vdateDisplay": "01-Jul-2019",
  "vdate": 1561939200000,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```



```

    }
  }

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PERIOD	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Webhook Configuration Download Service

Get Webhook Configuration and Status

Functional Area

Administration - Operations

Business Overview

This service retrieves the configured Webhook Listeners and any unused credential names. The response contains three types of information:

- **Webhook Configurations:** Provides details about the current webhook configuration, including the listener URL and interval seconds. This information is retrieved from the `webhook_config` table.
- **Current Status:** Indicates the current status of the webhook, including the since timestamp to be used in the next call and an estimate of when the next heartbeat or polling process will be triggered. If the most recent heartbeat encountered a failure, the error message will be included. This information is retrieved from the `webhook_status` table.
- **Metrics Summary for today and yesterday:** Included only if the query parameter `includeMetrics` is set to Y. The metrics summary is at day level and is limited to the current day (starting from midnight GMT) and yesterday (GMT timezone) and provides total counts of webhook calls and heartbeats triggered on the day, along with the number of successful calls. It additionally contains detailed activity log message of the last webhook and heartbeat processes executed for the day to assist with troubleshooting if webhook publish is unsuccessful. This data is retrieved from the `webhook_metrics` table. Note: The number of rows in this table can grow very large over time and fetching the metrics summary will be a performance-intensive operation and service may take additional time compared to a call with `includeMetrics = N`.

Service Type

GET

ReST URL

MerchIntegrations/services/administration/operations/webhooks

Input Parameters

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
offsetkey	No	String	Offset key
service	No	String	Service
includeMetrics	No	String	Include metrics summary for last two day - Y/N. Default N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Response Code: 200 (Success)

Table 5-77 PageResultItemsWebhookConfig - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-78 WebhookConfig - Object. See list of elements for detail

Element Name	Required	Data Type	Description
credentialName	Yes	String (100)	This field contains the Credential name.
credentialType	Yes	String (30)	This field contains the Credential type.
service	No	String (100)	This field contains the service name.

Table 5-78 (Cont.) WebhookConfig - Object. See list of elements for detail

Element Name	Required	Data Type	Description
listenerUrl	No	String (255)	This field contains the listener URL.
enabled	No	String (1)	This field indicates if the service is enabled. Default is 'Y'.
intervalSeconds	No	Number	This field contains the frequency in seconds webhook will poll for new events. Default is 900 seconds.
includePayload	No	String (1)	This field indicates if data should be included in the webhook call. Default is 'N'.
pageLimit	No	Number	This field contains the number of record within a payload during pagination and is used to control the message size. Default is 1000.
queryParameter	No	String	This field contains the configured query parameter (include or exclude) for this api.
heartbeatIntervalHours	No	Number	This field frequency in hours on which heartbeat will be pinged to listener. Default is 6 hours.
sinceTimestamp	No	dateTime	This field hold the configured since-timestamp for this API.
status	No	Object	This node hold the current status of the webhook for reference

Table 5-78 (Cont.) WebhookConfig - Object. See list of elements for detail

Element Name	Required	Data Type	Description
metrics	No	Collection of Object	This will be only populated if query parameter includeMetrics is Y. This node hold the summary view of webhook activities at day level since midnight (UTC) yesterday and can hold upto two entries - one for today and one for yesterday. The record for today or yesterday is populated only if some webhook or heartbeat activity happened for the day.

Table 5-79 Status - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nextPollTime	No	dateTime	This field holds the timestamp when the next internal polling to check for delta change is due.
nextPollSinceTimestamp	No	dateTime	This field holds the since-timestamp that will be used in the next internal poll. This field is updated after every successful publish and holds the pointer of the last successful publish.
nextPollOffsetKey	No	String	This field holds a value only when multi-page publish is in progress. During a multi-page publish, this helps to identify the key of the last record of the last page published. After the last page is published, it is reset to blank.

Table 5-79 (Cont.) Status - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nextHeartbeatTime	No	dateTime	This field holds the timestamp when the heartbeat is due. One the heartbeat is due, before any subsequent webhook publish, an heartbeat will be fired.
heartbeatError	No	String	This field will hold the error message if the previous heartbeat was a failure.

Table 5-80 Metrics - Object. See list of elements for detail

Element Name	Required	Data Type	Description
date	No	dateTime	This field holds either today or yesterday date for which the subsequent metrics information is applicable.
count	No	Number	This field contains the number of calls (excluding heartbeat) made to the listener for the date
countSuccess	No	Number	This field contains the number of successful calls (excluding heartbeat) made to the listener for the date
lastPublishTime	No	dateTime	This field holds the timestamp when the webhook engine initiated the last publish for the date.
lastPublishActivityLog	No	String	This field holds the log messages when the webhook engine initiated the last publish call for the date.
countHeartbeat	No	Number	This field contains the number of heartbeat calls made to the listener for the date
countSuccessHeartbeat	No	Number	This field contains the number of successful heartbeat calls made to the listener for the date

Table 5-80 (Cont.) Metrics - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lastHeartbeatTime	No	dateTime	This field holds the timestamp when the webhook engine initiated the last heartbeat call for the date.
lastHeartbeatActivityLog	No	String	This field holds the log messages when the webhook engine initiated the last heartbeat call for the date.

Table 5-81 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "credentialName": "APP_DEV1",
      "credentialType": "OAUTH",
      "service": "foundation/item",
      "listenerUrl": "https://apigateway.myretailer.com/webhook/mfcs/item",
      "enabled": "Y",
      "intervalSeconds": 900,
      "includePayload": "N",
      "pageLimit": 1000,
      "queryParameter": "include=items.item,items.status",
      "heartbeatIntervalHours": 6,
      "sinceTimestamp": "2001-12-31T23:59:59.000Z",
      "status": {
        "nextPollTime": "2001-12-31T23:59:59.000Z",
        "nextPollSinceTimestamp": "2001-12-31T23:59:59.000Z",
        "nextPollOffsetKey": null,
        "nextHeartbeatTime": "2001-12-31T23:59:59.000Z",
        "heartbeatError": null
      }
    },
  ],
}
```

```

"metrics": [
{
"date": "2001-12-31T23:59:59.000Z",
"count": 7,
"countSuccess": 6,
"lastPublishTime": "2001-12-31T23:59:59.000Z",
"lastPublishActivityLog": "<Timestamp>: Calling url for payload...",
"countHeartbeat": 3,
"countSuccessHeartbeat": 3,
"lastHeartbeatTime": "2001-12-31T23:59:59.000Z",
"lastHeartbeatActivityLog": "<Timestamp>: Calling url for heartbeat..."
}
]
},
"hasMore": true,
"limit": 1,
"count": 1,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?limit=1000",
"rel": "self"
}
]
}

```

Webhook Configuration Upload Service

Manage Webhook Credential

Functional Area

Administration - Operations

Business Overview

This service maintains credentials for the Webhooks listener and allows for setting up new credentials, along with updating or deleting existing credentials. Credentials must be set up in advance of webhook listener setup. A credential cannot be deleted if it is attached to a listener.

The credentials are securely stored in OCI S3. The supported credentials are Basic Auth, OAuth2 (Client Credentials) and API key (Request Parameter - X-API-Key). Based on the type of credential, the corresponding field should be provided. In case of OAuth, the URL to retrieve the OAuth token should be allowlisted and should have a publicly trusted SSL/TLS certificate.

Service Type

PUT

ReST URL

MerchIntegrations/services/administration/operations/webhook/credential/manage

Input Payload Details

Table 5-82 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
credentialName	Yes	String (100)	Unique Credential name for storing credential details for webhook listener endpoint.
credentialType	No	String (30)	Type of credential supported by listener. Valid options are OAUTH, BASICAUTH or APIKEY
username	No	String (100)	Contains the username for basic auth credential
password	No	String (100)	Contains the password for basic auth credential
apikey	No	String (2000)	Contains apikey. API Key will be passed in Request Parameter X-API-Key
oauthClientId	No	String (255)	Contains the client id for Oauth - Client Credential authentication
oauthClientSecret	No	String (255)	Contains the client secret for Oauth - Client Credential authentication
oauthTokenUrl	No	String (255)	Contains the OAuth Token URL for Oauth - Client Credential authentication
oauthScope	No	String (255)	Contains the scope for Oauth - Client Credential authentication. Scope is optional.
heartbeatIntervalHours	No	Number (2)	Controls the frequency in which heartbeat will be published to the listener URL
deleteInd	No	String (1)	Pass a value of Y to delete the credential. To delete a credential, none of the listener should be using this credential. Supported values as Y or N. The default if left blank is N.

Sample Input Message

```

{
  "credentialName": "APP_DEV1",
  "credentialType": "OAUTH",
  "username": "",
  "password": "",
  "apikey": "",
  "oauthClientId": "oracle_mfcs_client_dev1",
  "oauthClientSecret": "6c2c0acb-88a1-4067-865b-5eb75b2989c3",
  "oauthTokenUrl": "https://iam-6c2c0acb.iamprovider.com:443/oauth2/v1/token",
  "oauthScope": "mfcs-webhook-dev1",
  "deleteInd": ""
}

```


Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Create record was successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` is present when the input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Manage Webhooks

Functional Area

Administration - Operations

Business Overview

This service is used to register and manage webhook listeners. To register a listener, the credential for the listener should be set up first using the Manage Webhook Credential service.

Webhook Overview

The listener URL should be an allowlisted https ReST endpoint supporting THE POST method and having publicly trusted SSL/TLS certificate. a Webhook can be configured for THE Publish API and THE name of the service should match with the value in THE `merchapi_name` column from the `merchapi_config` table. Once a webhook is successfully registered, it is added to the server threadpool after a short interval and, thereafter, runs based on the configured interval.

Three types of publish are available:

- **HEARTBEAT** - The frequency (in hours) of this can be defined using the field `heartbeatIntervalHours`. A ping call is made to the listener endpoint to validate the connectivity, credential, and to ensure that the webhook server is up and running. If the heartbeat call fails, the webhook call with the event or payload is put on hold until the heartbeat connectivity is established. Once the heartbeat connectivity is established, the event or payload publish starts.
- **EVENT** - A webhook call of type Event indicates to the listener that there are one or more functional data packets that have been created or updated since the last webhook call, and the listener should poll the Publish API to publish the change. A webhook of type Event is

the default webhook configuration and should be the preferred configuration to keep the data traffic through the webhook to a minimum. If the field `includePayload` is left blank or is set to `N`, this indicates that the webhook is configured for event publishing.

- **PAYLOAD** - A webhook of type `Payload` includes the data as part of the webhook publish. The data is published in JSON format and is enclosed in the attribute named `payload`. To enable payload publish, set the value of `includePayload` to `Y`. The payload supports pagination to keep the individual message size in control. Use the field `pageLimit` to define the number of records included in the call. If `pageLimit` is not defined, it is set to 1000. For a payload publish, the server posts each subsequent page immediately upon receiving the success response from the client for the previous page.

The processing of a webhook is asynchronous. Using a background micro batch engine, the webhook server checks for new events in `merchapi_json` tables for changes. The frequency at which the server polls the `merchapi_json` table is configured using the field `intervalSeconds`. Every `<interval>` seconds, the server polls the table to check for new entries and, if there are updates, it makes a webhook call to the listener URL to publish the event or payload depending on the webhook configuration. If there are no new entries, it does not make an empty webhook call.

To keep track of the records that have been published, and have the subsequent webhook publish only include the newer records since the last call, the timestamp of the webhook pool is persisted by the server and is named the `since_timestamp`. When the webhook is registered for the first time, the `since_timestamp` is defaulted to Unix time zero (1970) unless a `since_timestamp` was provided as part of the registration. The server polls the `json` table to check for newer entries having timestamp greater than `since_timestamp`.

- For a webhook of type `Event`, the `since_timestamp` is included in the publish to inform the listener that there are records having a timestamp greater than the `since_timestamp` and they should poll the Publish API using a query parameter of `since_timestamp`.
- For a webhook of type `Payload`, the server internally calls the Publish API with the `since_timestamp` and using `before` as the current server time and the configured pagination limit. The server redirects the response from the Publish API to the webhook listener. Depending on the number of records and the pagination size, more than one call can be made to the listener URL. Once the last page is published, the server replaces the `since_timestamp` with the server time when the first poll is performed. Subsequent polling uses the new `since_timestamp` to check for newer entries.
- If there are no new entries, the value of `since_timestamp` is not changed and this value can be seen as a the timestamp when the last publish of type `Event` or `Payload` was made.

The `since_timestamp` is available as part of webhook configuration and this can be updated for existing webhooks. This can be useful to request a full download or to replay all changes since, for example, last week for data corruption scenarios.

Enable/Disable The flag `Enabled` takes a value of `Y` or `N`, and can be used to activate or deactivate a webhook. If the webhook is disabled, it pauses all polls and even heartbeat calls to the listener. Once the webhook is enabled, it starts polling using the `since_timestamp` of the last publish.

Query Parameter This optional field enables Dynamic Payload Filtering, allowing clients to fetch selective fields or exclude fields based on the listener application's needs. Several webhook-supported APIs accept a query parameter of `include` or `exclude`, with values representing JSON field names using dot notation (for example, `include=items.item,items.status,itemDescription`). When provided, this query parameter is appended to the internal service call URL when fetching the payload. The only supported query parameters through webhooks are `include` and `exclude`, and it is recommended to test the query parameter values before configuring them in a webhook.

Update Webhook Configuration

To update the configuration of a webhook listener, include the fields that need to be updated and their values will be replaced. The fields left blank will not be changed. A configuration change can have an impact on the current publish process (multi-page payload publish).

- **Listener URL** change will be applied immediately. A change of listener causes a heartbeat to be fired and, once the heartbeat is successful, subsequent calls are published to the new URL. If the change happened in between pagination publishes, the remaining pages are published to the new URL. A listener change is a major change and, if the intention for the new listener URL is to get the entire dataset or to start with a particular date and time, adjust the `since timestamp` accordingly to get the complete/required payload.
- A change of **credential** takes effect immediately. A change of credential causes a heartbeat check to be fired before the next event or payload publish.
- **Since Timestamp** change is applied immediately. This will terminate any in-progress pagination call and publish will restart with the new since timestamp.
- **Page limit** change takes effect immediately for the subsequent page publish. In case of an in-progress pagination publish, the remaining pages publish happens with the new page limit.
- **Interval second** change takes effect a few minutes after the webhook engine syncs up with update configuration.
- **Disable** webhook - This takes effect immediately and may pause a pagination call in between. The remaining pagination pages are published when the webhook is enabled. **Enabling** of the webhook takes few minutes for the server engine to include this webhook in its thread pool.
- A change of **heartbeat interval hour** takes effect after the next heartbeat is fired.
- **IncludePayload** - This change takes effect immediately and stops any pagination call. It uses the currently active since timestamp to check for changes, unless the since timestamp was also updated as part of the update request.
- **Query Parameter** change will take into effect immediately for the subsequent page publish. To remove a previously set query parameter, provide blank spaces.

Delete Webhook

To delete a registered webhook, set the flag `deleteInd` to Y. That immediately stops subsequent calls and purges all configurations for the webhook.

Service Type

PUT

ReST URL

MerchIntegrations/services/administration/operations/webhooks/manage

Input Payload Details

Table 5-83 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of webhook listeners

Table 5-84 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
service	Yes	String (100)	Service name for which webhook listener has to be configured. The service name should match with merchapi_name in merchapi_config table. On the api's having the json_table column populated are eligible for webhook.
credentialName	No	String (100)	Registered credential name that was setup using the Manage Webhook Credential service. For update, this can be left blank.
listenerUrl	No	String (255)	A whitelisted listener endpoint which is running a post service to consume the webhook call and returns 200 for success.
includePayload	No	String (1)	Indicates if the webhook publish will contain the data or just the event. Takes a value of Y or N. A value of blank is defaulted to N during initail registration. In case of update of already registered webhook, leave it blank to keep the value unchanged.
intervalSeconds	No	Number (6)	Defines the polling seconds for the webhook server to poll to json table for new events. This value is in seconds.
pageLimit	No	Number (6)	If the webhook is registered to include payload, this defines the maximum number of records in one publish call. If the number of records to be published are greater than page limit, additional pagination call will be made to publish the entire data set. Default value is 1000.
sinceTimestamp	No	date-time	Allows the listener to define if the publish should include events that happens after a particular time. This also allows for replaying already published daata/event by setting this to a earlier time.
enabled	No	String (1)	Takes a value of Y (default if null) or N. If the webhook is disabled, it will pause polling and publish of webhook (inlcuding heartbeat)
heartbeatIntervalHours	No	Number (2)	Controls the frequency in which heartbeat will be published to the listener URL
deleteInd	No	String (1)	Use this to delete an existing listener by setting the value as Y. the default if left blank is N.

Table 5-84 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
queryParameter	No	String (2000)	This field can be used to apply API-supported query parameter of include or exclude when an internal service is invoked to fetch a payload. It is highly recommended to test the query parameter by calling the GET APIs to avoid unexpected errors at runtime. To remove an earlier provided query parameter value, submit the value with an empty space.

Table 5-85 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
service	Yes	String (100)	Service name for which the registration failed
errorMessage	Yes	String (2000)	Error message

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "service": "foundation/item",
      "credentialName": "APP_DEV1",
      "listenerUrl": "https://apigateway.myretailer.com/webhook/mfcs/item",
      "includePayload": null,
      "intervalSeconds": 900,
      "pageLimit": 1000,
      "sinceTimestamp": "2001-12-31T23:59:59.123Z",
      "enabled": null,
      "heartbeatIntervalHours": 6,
      "deleteInd": null,
      "queryParameter": "include\u003ditems.item,items.status"
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service. The element `businessError` will be present if the payload passes schema validation but an exception is caught while processing the business logic.

Table 5-86 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
service	Yes	String (100)	Service name for which the registration failed
errorMessage	Yes	String (2000)	Error message

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "service": "foundation/items/manage",
      "errorMessage": "Invalid API"
    }
  ]
}
```

System Options

The following services are included in this functional area:

- [System Options Download Service](#)
 - [Get Functional System Options](#)
 - [Get Inventory Movement System Options](#)
 - [Get Procurement System Options](#)

System Options Download Service

The following services are included in this functional area:

- [Get Functional System Options](#)
- [Get Inventory Movement System Options](#)
- [Get Procurement System Options](#)

Get Functional System Options

Business Overview

Retrieve Merchandising's Functional Config Options.

Service Type

GET

ReST URL

/Common/FuncSysOps

Input Parameters

N/A

Output

Table 5-87 FunctionalConfigRDO

Parameter Name	Data Type
importInd	String
orgUnitInd	String
supplierSitesInd	String
contractInd	String
elcInd	String

JSON Structure:

```
"{"  
  "links": [],  
  "importInd": "Y",  
  "orgUnitInd": "Y",  
  "supplierSitesInd": "Y",  
  "contractInd": "Y",  
  "elcInd": "Y",  
  "hyperMediaContent": {  
    "linkRDO": []  
  }  
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
FUNCTIONAL_CONFIG_OPTIONS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Inventory Movement System Options

Business Overview

Retrieve Merchandising's Inventory Movement Unit Options.

Service Type

GET

ReST URL

/Common/InvMovSysOps

Input Parameters

N/A

Output

Table 5-88 InvMoveUnitOptRDO

Parameter Name	Data Type
allocMethod	String
applyProfPresStock	String
autoRcvStore	String
closeOpenShipDays	BigDecimal
costMoney	BigDecimal
costOutStorage	BigDecimal
costOutStorageMeas	String
costOutStorageUom	String
costWhStorage	BigDecimal
costWhStorageMeas	String
costWhStorageUom	String
defaultAllocChrgInd	String
defaultOrderType	String
defaultSizeProfile	String
deptLevelTransfers	String
distributionRule	String
duplicateReceivingInd	String
increaseTsfQtyInd	String
intercompanyTransferBasis	String
invHistLevel	String
locActivityInd	String
locDlvryInd	String
lookAheadDays	BigDecimal
maxWeeksSupply	BigDecimal
ordWorksheetCleanUpDelay	BigDecimal
racRtvTsfInd	BigDecimal
rejectStoreOrdInd	String
replOrderDays	String
rtvNadLeadTime	BigDecimal
rtvUnitCostInd	BigDecimal
shipRcvStore	String
shipRcvWh	String
storageType	String
storePackCompRcvInd	String
wfDefaultWh	String
targetRoi	BigDecimal
tsfAutoCloseStore	BigDecimal

Table 5-88 (Cont.) InvMoveUnitOptRDO

Parameter Name	Data Type
tsfAutoCloseWh	String
tsfCloseOverdue	String
simForceCloseInd	String
tsfForceCloseInd	String
tsfOverReceiptInd	String
tsfMdStoreToStoreSndRcv	String
tsfMdStoreToWhSndRcv	String
tsfMdWhToStoreSndRcv	String
tsfMdWhToWhSndRcv	String
tsfPriceExceedWacInd	String
ssAutoCloseDays	String
wsAutoCloseDays	BigDecimal
swAutoCloseDays	BigDecimal
wwAutoCloseDays	BigDecimal
wfOrderLeadDays	BigDecimal
whCrossLinkInd	BigDecimal
wrongStReceiptInd	String

JSON Structure:

```
{
  "links": [],
  "allocMethod": "P",
  "applyProfPresStock": "N",
  "autoRcvStore": "Y",
  "closeOpenShipDays": 3,
  "costMoney": 7.5,
  "costOutStorage": 1.5,
  "costOutStorageMeas": "P",
  "costOutStorageUom": null,
  "costWhStorage": 1.5,
  "costWhStorageMeas": "P",
  "costWhStorageUom": null,
  "defaultAllocChrgInd": "Y",
  "defaultOrderType": "WAVE",
  "defaultSizeProfile": "N",
  "deptLevelTransfers": "Y",
  "distributionRule": "PRORAT",
  "duplicateReceivingInd": "N",
  "increaseTsfQtyInd": "N",
  "intercompanyTransferBasis": "T",
  "invHistLevel": "A",
  "locActivityInd": "Y",
  "locDlvryInd": "Y",
  "lookAheadDays": 7,
  "maxScalingIterations": null,
```

```

    "maxWeeksSupply": 5,
    "ordWorksheetCleanUpDelay": 1,
    "racRtvTsfInd": "A",
    "rejectStoreOrdInd": "N",
    "replOrderDays": 3,
    "rtvNadLeadTime": 1,
    "rtvUnitCostInd": "A",
    "shipRcvStore": "Y",
    "shipRcvWh": "Y",
    "storageType": "W",
    "storePackCompRcvInd": "Y",
    "wfDefaultWh": 1212,
    "targetRoi": 7,
    "tsfAutoCloseStore": "Y",
    "tsfAutoCloseWh": "Y",
    "tsfCloseOverdue": "Y",
    "simForceCloseInd": "NL",
    "tsfForceCloseInd": "SL",
    "tsfOverReceiptInd": "NL",
    "tsfMdStoreToStoreSndRcv": "S",
    "tsfMdStoreToWhSndRcv": "S",
    "tsfMdWhToStoreSndRcv": "S",
    "tsfMdWhToWhSndRcv": "S",
    "tsfPriceExceedWacInd": "Y",
    "ssAutoCloseDays": 1,
    "wsAutoCloseDays": 1,
    "swAutoCloseDays": 1,
    "wwAutoCloseDays": 1,
    "wfOrderLeadDays": null,
    "whCrossLinkInd": "Y",
    "wrongStReceiptInd": "Y",
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
INV_MOVE_UNIT_OPTIONS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Procurement System Options

Business Overview

Retrieve Merchandising's Procurement Unit Options.

Service Type

GET

ReST URL

/Common/POSysOps

Input Parameters

N/A

Output

ProcurementUnitOptionsRDO

Parameter Name	Data Type
backpostRcaRuaInd	String
calcNegativeIncome	String
copyPoCurrRate	String
costLevel	String
creditMemoLevel	String
dealAgePriority	String
dealLeadDays	BigDecimal
dealTypePriority	String
deptLevelOrders	String
ediCostOverrideInd	String
expiryDelayPreIssue	BigDecimal
genConsignmentInvcFreq	String
genConInvcItmSupLocInd	String
latestShipDays	BigDecimal
ordApprCloseDelay	BigDecimal
ordApprAmtCode	String
ordAutoClosePartRcvdInd	String
ordPartRcvdCloseDelay	BigDecimal
orderBeforeDays	BigDecimal
orderExchInd	String
otbSystemInd	String
rcvCostAdjType	String
reclassApprOrderInd	String
redistFactor	BigDecimal
softContractInd	String
wacRecalcAdjInd	String

JSON Structure:

```
{
  "links": [],
  "backpostRcaRuaInd": "N",
  "billToLoc": "1000",
  "calcNegativeIncome": "N",
  "copyPoCurrRate": null,
  "costLevel": "DNN",
  "creditMemoLevel": "D",
  "dealAgePriority": "O",
  "dealLeadDays": 1,
  "dealTypePriority": "P",
```

```

    "deptLevelOrders": "N",
    "ediCostOverrideInd": "Y",
    "expiryDelayPreIssue": 30,
    "genConsignmentInvFreq": "M",
    "genConInvItmSupLocInd": "I",
    "latestShipDays": 30,
    "ordApprCloseDelay": 1,
    "ordApprAmtCode": "C",
    "ordAutoClosePartRcvdInd": "N",
    "ordPartRcvdCloseDelay": 1,
    "orderBeforeDays": 5,
    "orderExchInd": "N",
    "otbSystemInd": "N",
    "rcvCostAdjType": "F",
    "reclassApprOrderInd": "Y",
    "redistFactor": 2,
    "softContractInd": "Y",
    "wacRecalcAdjInd": "N",
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PROCUREMENT_UNIT_OPTIONS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Cost

The following services are included in this functional area:

- [Average Cost](#)
 - [Average Cost Upload Service](#)
 - * [Update Average Cost](#)
- [Cost Changes](#)
 - [Cost Change Upload Service](#)
 - * [Create Cost Change Effective Immediately](#)
 - * [Manage Cost Changes](#)
- [Pricing Cost](#)
 - [Pricing Cost Download Service](#)
 - * [Get Pricing Cost](#)

Average Cost

The following services are included in this functional area:

- [Average Cost Upload Service](#)
 - [Update Average Cost](#)

Average Cost Upload Service

The following services are included in this functional area:

- [Update Average Cost](#)

Update Average Cost

Functional Area

Cost - Average Cost

Business Overview

This service can be used to update the weighted average cost (WAC) from an external system for one or more item/location combinations. It also creates a transaction data record posting with transaction code 70 for the difference in cost, based on the owned inventory at the location at the time the cost change is applied.

The web service can be called with the following details:

- Item
- Location
- Location type
- New average cost (must be greater than 0)

Service Type

PUT

ReST URL

MerchIntegrations/services/cost/averageCost/update

Input Payload Details

Table 5-89 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of items and locations for which weight average cost needs to be updated.

Table 5-90 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the item number for which the average cost has to be updated
location	Yes	Number (10)	Store or virtual warehouse location number.
locationType	Yes	String (1)	Location Type. Valid values are 'S' (store) and 'W' (warehouse).

Table 5-90 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
averageCost	Yes	Number (20,4)	New average cost.

Sample Input Message

```
{
  "items": [
    {
      "item": "101450060",
      "location": 6000,
      "locationType": "S",
      "averageCost": 19.50
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Cost Changes

The following services are included in this functional area:

- [Cost Change Upload Service](#)
 - [Create Cost Change Effective Immediately](#)
 - [Manage Cost Changes](#)

Cost Change Upload Service

The following services are included in this functional area:

- [Create Cost Change Effective Immediately](#)
- [Manage Cost Changes](#)

Create Cost Change Effective Immediately

Functional Area

Cost - Cost Changes

Business Overview

This service allows external systems to update unit costs within Merchandising. All cost changes that are sent through this service are executed immediately. This service creates both the cost change events with an effective date of the current date, as well as updates unit costs for item/locations that already exist in Merchandising. It does not create or delete item/location relationships in Merchandising. Cost changes can be performed at the item level, or at the following levels of the organization hierarchy - chain, area, region, district or store. Unit costs are updated for all stores within the location group. Warehouses are only impacted by cost changes applied at the warehouse level, since they are not part of the organization hierarchy. Since the user has the flexibility to have child items with the same or different purchase type compared to the parent item, any cost change created at the parent item level will impact only those child items having the same ownership as the parent. Cost changes can be created for individual child items as well.

The service takes in a collection of cost changes and will return success and failure through the service response object. This service validates that all the required fields are provided and also checks the supplier's currency and the item status. In case differentiator IDs are passed in the message, it verifies that they are valid for the provided item. The service also retrieves the following:

- Transaction level items, if the passed in item is an item parent.
- All locations based on the passed in hierarchy type and value, if provided.
- All item/location combinations where the passed in supplier/country is the primary supplier/country at an item location.
- All orderable buyer packs that the passed-in item or its children, if above transaction level.
- All item/locations on approved (and worksheet) order, if the recalculate order indicator is set to Yes.

This service performs the following actions:

- Creates a cost change event in the 'Executed' status, with the current date set as the effective date.

- Updates the unit cost in Merchandising for all item/supplier/country and item/supplier/country/locations based on the information provided.
- Creates price history records for all item/locations that get updated as part of the cost change.
- If the recalculate order indicator is set to Yes, it updates all relevant order/item/locations unit cost in Merchandising.

It is important to note that cost changes sent through this service do not include estimated landed costs. The cost updated here is the default purchase cost, before any deals, that will be used for purchase orders created in Merchandising, similar to cost changes initiated in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/cost/costChange/immediate/create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-91 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-92 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item to which the cost change applies. It can be a parent item, but its item level cannot be greater than its transaction level. This cannot be a buyer pack.
supplier	Yes	Number (10)	The ID of the supplier that sources the items the cost change affects. This value must be predefined on the SUPS table. This can be a primary or non-primary supplier.
originCountry	Yes	String (3)	The identifier of the origin country of the item supplier to which the cost change will apply. This value must be predefined on the COUNTRIES table. This can be the primary or a non-primary country.

Table 5-92 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	No	String (10)	The identifier for a differentiator. This can be used with a parent or grandparent item. The cost change will explode to all the parent's transaction level items with the differentiator. This value must be predefined on the DIFF_IDS table.
unitCost	Yes	Number (20,4)	The new unit cost of the item in the currency specified on the message. Tables item_supp_country_loc, item_loc_soh, and ordloc will also be updated with this value, depending on the values in tags hier_level and hier_value.
recalculateOrderInd	Yes	String (1)	This field will indicate if orders in approved status for items on the cost change will be recalculated with the new cost. Valid values will be Y (yes) and N (no).
currencyCode	Yes	String (3)	The currency code of the unit cost. This value must be predefined on the currencies table and must be the currency of the supplier.
hierarchyLevel	No	String (2)	The hierarchy level of the hierarchy value. This field will indicate the level of the organizational hierarchy to which the cost change applies. Valid values are CH (chain), AR (area), RE (region), DI (district), S (store), W (warehouse).
details	No	Collection of Object	References a collection of hierarchy details
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 5-93 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The ID of the specified hierarchy level which encompasses the locations affected by the cost change.

Table 5-94 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "items": [
    {
      "item": "1009000006",
      "supplier": 2400,
      "originCountry": "US",
      "diffId": "String",
      "unitCost": 9.65,
      "recalculateOrderInd": "N",
      "currencyCode": "USD",
      "hierarchyLevel": "S",
      "details": [
        {
          "hierarchyValue": 1521
        }
      ],
      "customFlexAttribute": [
        {
          "name": "REVIEW_DATE",
          "value": null,
          "valueDate": "2001-12-31"
        }
      ],
      "purchaseRate": 18.00
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
```

```
"message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Manage Cost Changes

Functional Area

Cost – Cost Change

Business Overview

This service is used by an external source, usually a supplier, to create and maintain future-dated cost changes in Merchandising. The service accepts a collection of cost changes and returns success or failure through the service response object. This service uses the cost change induction staging tables, populates those tables, then invokes the existing cost change induction API that validates the data and populates the base tables.

Service Type

Post

ReST URL

RmsRestServices/services/private/costChange

Input Parameters – Cost Change Header Record

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
action	No	No	No	string	Action type for Cost Change. Valid values are NEW(for create), MOD (for update) or DEL (for delete)

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
costChange	Yes	Yes	Yes	number	Holds the ID that uniquely identifies the cost change. The ID should not overlap with cost changes created through other modes such as induction or the Merchandising UI.
costChangeDescription	Yes	Yes	No	string	Cost change description. This along with the cost change number identifies the cost change for modification and deletion.
reason	Yes	Yes	No	string	Indicates the reason code for the cost change. Valid values for this input can be found in the COST_CHG_REASON table.
activeDate	Yes	Yes	No	date	Indicates the date on which the cost change will become active. The date format should be YYYY-MM-DD
status	Yes	Yes	No	string	Contains the status of the cost change. Valid values are W = Worksheet, D = Deleted (to be deleted), C = Canceled, A = Approved, E = Extracted, R = Rejected, S = Submitted
costChangeOrigin	Yes	Yes	No	string	Contains a code that identifies the origin of the cost change. This code is used to determine whether the event was created by supplier or by SKU. Cost events with a supplier origin will have multiple detail records all having the same supplier. Cost events with a SKU origin will have multiple detail records that may have multiple SKUs and multiple suppliers. Valid Values are SUP - By Supplier, SKU - By SKU

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
approvalId	No	No	No	string	Contains the user ID who last approved the cost change.
suppItemDetails					Child Node – cost change detail record. Either this node is populated or the detail location node.
suppItemLocDetails					Child Node – cost change detail location record. Either this node or the detail node is populated.
customFlexAttribute					Child Node – custom flex attributes

Input Parameters – Cost Change Detail Record

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
action	No	No	No	string	Holds action type. Valid values are NEW (for create), MOD (for update) or DEL (for delete)
supplier	Yes	Yes	Yes	number	Contains the number that identifies the supplier associated with the cost change.
originCountryId	Yes	Yes	Yes	string	The country where the item was manufactured or significantly altered.
item	Yes	Yes	Yes	string	Unique alphanumeric value that identifies the item.
bracketValue1	No	Yes	Yes	number	The values are the various levels in the vendor price list whereby costing of the items will be affected as the levels are achieved on the purchase orders.
costChangeType	Yes	Yes	No	number	Describes the type of cost change. Valid values are A - Amount, F - Fixed or P - Percentage.
costChangeValue	Yes	Yes	No	number	Holds the value of cost change

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
recalcOrdInd	No	Yes	No	string	This field indicates whether or not approved order costs should be re-calculated when a supplier cost has changed. Default value for this input is No. Valid values are Y - Yes, N - No
defaultBracketInd	No	No	No	string	This field contains the value that will control which cost is the base cost for processing throughout the system. Each bracket group (either by supplier, supplier/department, supplier/department/location, supplier/location) must have one bracket defined as the default bracket.
calculationBasis	No	No	No	string	This indicates whether the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	No	No	number	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Input Parameters – Cost Change Detail Location Record

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
action	No	No	No	string	Holds action type. Valid values are NEW (for create), MOD (for update) or DEL (for delete)

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
supplier	Yes	Yes	Yes	number	Contains the number that identifies the supplier associated with the cost change.
originCountryId	Yes	Yes	Yes	string	The country where the item was manufactured or significantly altered.
item	Yes	Yes	Yes	string	Unique alphanumeric value that identifies the item.
locType	Yes	Yes	Yes	string	Contains the type of the location. Valid values are Store and Warehouse.
loc	Yes	Yes	Yes	number	Number of the location the cost change will occur at. This can represent a store or a warehouse. Cost changes will be managed and stored at the physical warehouse level since the unit cost must remain consistent across all virtual warehouses within the same physical warehouse. The sccext batch program will handle exploding the cost change down to all virtual warehouses for the inputted physical warehouse.
bracketValue1	No	No	No	number	The values are the various levels in the vendors price list whereby costing of the items will be affected as the levels are achieved on purchase orders.
costChangeType	Yes	Yes	No	string	Describes the type of cost change
costChangeValue	Yes	Yes	No	number	Holds the value of the cost change
recalcOrdInd	No	Yes	No	string	This field indicates whether or not approved order costs should be re-calculated when a supplier cost has changed. Default value for this input is No. Valid values are Y - Yes, N - No

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
defaultBracketId	No	No	No	string	This field contains the value that will control which cost is the base cost for processing throughout the system. Each bracket group (either by supplier, supplier/department, supplier/department/location, supplier/location) must have one bracket defined as the default bracket.
calculationBasis	No	No	No	string	The ID of the user who last updated this record.
purchaseRate	No	No	No	number	The date and time when this record was last updated.

Input Parameters – Custom Flex Attribute Record

Parameter Name	Required (For Create)	Required (For Update)	Required (For Delete)	Data Type	Description
action	No	No	NA	string	Holds action type. Valid values are NEW (for create), MOD (for update) or DEL (for delete)
cfaName	Yes	Yes	NA	string	
cfaValue	No	No	NA	string	
cfaValueDate	No	No	NA	date	

Example JSON input:

```
[
  {
    "action": "NEW",
    "costChange": 123412,
    "costChangeDesc": "Sample Cost Change",
    "reason": 6,
    "activeDate": "2022-04-18",
    "status": "A",
    "costChangeOrigin": "SKU",
    "approvalId": "SomeUser",
    "suppItemDetails": [
      {
        "action": "NEW",
        "supplier": 60000,
        "originCountryId": "US",
        "item": "123412312",
        "bracketValue1": 0,
        "costChangeType": "F",
```



```

        "costChangeValue": 19,
        "recalcOrdInd": "N",
        "defaultBracketInd": "",
        "calculationBasis": "C",
        "purchaseRate": 0
    }
],
"suppItemLocDetails": [
    {
        "action": "NEW",
        "supplier": 30201,
        "originCountryId": "US",
        "item": "103500024",
        "locType": "S",
        "loc": 1521,
        "bracketValue1": 0,
        "costChangeType": "F",
        "costChangeValue": 19,
        "recalcOrdInd": "N",
        "defaultBracketInd": "",
        "calculationBasis": "",
        "purchaseRate": null
    }
],
"customFlexAttribute": [
    {
        "action": "NEW",
        "cfaName": "string",
        "cfaValue": "string",
        "cfaValueDate": "2022-04-18"
    }
]
]

```

Output – Cost Change Response Object

Attribute	Datatype	Description
message	string	Indicates processing outcome and the message is defined by cost type IIPS. Default configured values are “Processed Successfully”, “Processed with errors” or “Processed with warnings”
errors		Child Node – This will be empty when all the records got successfully processed. In case of multiple cost change request in the input and there are one/multiple/all failures, the error/warning reasons for each of the Non-successful cost change request will be captured here.

Output – Cost Change Response Record

Attribute	Datatype	Description
costChange	number	Contains the number that uniquely identifies the cost change.
field	string	This will contain the field in error.
item	string	Unique alphanumeric value that identifies the item.
supplier	number	Contains the number that identifies the supplier associated with the cost change.
originCountryId	string	The country where the item was manufactured or significantly altered.
loc	number	Number of the location the cost change will occur at.
locType	string	Contains the type of the location.
errorMessage	string	Contains the error message

Example JSON Output:

```
{
  "message": "Processed Successfully",
  "errors": [
    {
      "costChange": 123123,
      "field": "ACTIVE_DATE",
      "item": "123412312",
      "supplier": 60000,
      "originCountryId": "string",
      "loc": 0,
      "locType": "string",
      "errorMessage": "Invalid Active Date."
    }
  ]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_COST_SUSP_SUP_HEAD	Yes	Yes	No	Yes
SVC_COST_SUSP_SUP_DETAIL	Yes	Yes	No	Yes
SVC_COST_SUSP_SUP_DETAIL_LOC	Yes	Yes	No	Yes
CORESVC_COSTCHG_ERR	Yes	Yes	No	Yes
SVC_PROCESS_TRACKER	Yes	Yes	No	Yes
SVC_CFA_EXT	Yes	Yes	No	Yes
COST_SUSP_SUP_HEAD	Yes	Yes	Yes	Yes
COST_SUSP_SUP_DETAIL	Yes	Yes	Yes	Yes
COST_SUSP_SUP_DETAIL_LOC	Yes	Yes	Yes	Yes
COST_CHANGE_TEMP	Yes	Yes	Yes	Yes
COST_CHANGE_LOC_TEMP	Yes	Yes	Yes	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
COST_SUSP_SUP_HEAD_CFA_EXT	Yes	Yes	No	Yes

Pricing Cost

The following services are included in this functional area:

- [Pricing Cost Download Service](#)
 - [Get Pricing Cost](#)

Pricing Cost Download Service

The following services are included in this functional area:

- [Get Pricing Cost](#)

Get Pricing Cost

Functional Area

Cost - Pricing Cost

Business Overview

This web service is used to provide pricing cost information to external systems. This information is assumed to be primarily used by Order Management System (OMS), which manages franchise customer orders and needs visibility to cost information as part of the negotiation process for margin visibility.

Pricing cost for an item at an owned location is the unit cost for the primary supplier/country, less off invoice deals, plus estimated landed costs. Pricing cost for an item at a customer (franchise) location is the unit cost for the costing location, less any deals passed through, plus estimated landed costs (based on system option), plus the franchise cost template details. This service supports providing cost information for an item/location or item/supplier/location.

The following validations are carried out for this service -

- Only approved, transaction level items are valid.
- The location must be a company store or a physical warehouse that is customer orderable. For a physical warehouse, it must also include the channel ID that should be used.
- For physical warehouses, the cost returned will be for the virtual warehouse that matches the channel ID that is provided in the input. If there is no virtual warehouse that matches that channel in the physical warehouse, then the next best match will be determined based on channel type and the primary warehouse and protected flags on the virtual warehouses in the physical warehouse.

Service Type

POST

ReST URL

MerchIntegrations/services/cost/pricingCost/get

Input Payload Details

Table 5-95 Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-96 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the Merchandising item number that uniquely identifies the item. This is required input for the Pricing Cost Service
supplier	No	Number (10)	Contains the Merchandising Supplier number for retrieving the Pricing Cost
supplierCountry	No	String (3)	Contains the country, supplier belongs to for which the Pricing Cost needs to be retrieved.
fulfillLocation	Yes	Number (10)	Contains the location number. This is required input for the Pricing Cost Service.
locationType	Yes	String (1)	Type of item inventory location. Valid values are 'S' (store) and 'W' (warehouse).
channelId	No	Number (4)	Only used when a physical WH is passed. If a channel is passed the results will be limited by the passed in channel.
activeDate	No	date	Contains the active date for which the pricing cost needs to be retrieved.

Sample Input Message

```
{
  "items": [
    {
      "item": "100100068",
      "supplier": 2400,
      "supplierCountry": "US",
      "fulfillLocation": 1141,
      "locationType": "S",
      "channelId": 1,
      "activeDate": "2001-12-31"
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-97 GetResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-98 GetResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the Merchandising item number that uniquely identifies the item. This is required input for the Pricing Cost Service
supplier	No	Number (10)	Contains the Merchandising Supplier number for retrieving the Pricing Cost.
supplierCountry	No	String (3)	Contains the country, supplier belongs to for which the Pricing Cost needs to be retrieved.
fulfillLocation	Yes	Number (10)	Contains the location number. This is required input for the Pricing Cost Service.
locationType	Yes	String (1)	Type of item inventory location. Valid values are 'S' (store) and 'W' (warehouse).
channelId	No	Number (4)	Only used when a physical WH is passed. If a channel is passed the results will be limited by the passed in channel.
activeDate	No	date	Contains the active date for which the pricing cost needs to be retrieved.
pricingCost	Yes	Number (20,4)	Contains the retrieved pricing cost from the future cost table for the passed parameters.

Sample Response Message

```
{
  "items": [
    {
      "item": "100100076",
      "supplier": 6000,
      "supplierCountry": "US",
      "fulfillLocation": 6000,
      "locationType": "S",
      "channelId": 1,
      "activeDate": "2001-12-31",
      "pricingCost": 9.65
    }
  ]
}
```

```
]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Customer Orders

The following services are included in this functional area:

- [Customer Order Available Inventory Download Service](#)
 - [Get Available Inventory for Customer Orderable Items](#)
- [Customer Order Inventory Backorder Upload Service](#)
 - [Create Inventory Backorder](#)
- [Customer Order Item Substitution Upload Service](#)
 - [Create Customer Order Item Substitution](#)
- [Customer Order Upload Service](#)
 - [Cancel Fulfillment Order](#)
 - [Create Fulfillment Order](#)

Customer Order Available Inventory Download Service

The following services are included in this functional area:

- [Get Available Inventory for Customer Orderable Items](#)

Get Available Inventory for Customer Orderable Items

Functional Area

Customer Orders

Business Overview

This real-time inventory availability lookup facility can be used by external systems, such as an on-line order capture system (OOC) or order management system (OMS), to retrieve item/location available inventory based on Merchandising's current view of inventory. Merchandising will provide this information for any warehouse or store which is valid for customer order sourcing/fulfillment.

The Available inventory is calculated as Stock on Hand - (Transfer Reserved Quantity + Customer Reserved Quantity + RTV Quantity + Non-sellable Quantity)

Assumptions:

- This inventory detail lookup is only for customer orderable inventory - sellable items at customer order locations. If a physical warehouse and channel are passed into the service, then only the inventory for the customer orderable virtual warehouses for that physical warehouse/channel are returned.
- If the inventory lookup is for a pack item at store, the pack inventory is estimated based on the maximum number of complete packs which can be created by using the available inventory of its components.

Service Type

POST

ReST URL

MerchIntegrations/services/customerOrders/availableInventory/get

Input Payload Details

Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Array of String	Contains the transaction-level item numbers for which the inventory must be checked.
inventoryLocations	Yes	Collection of Object	Contains the locations at which to check inventory.

InventoryLocations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	The location at which to check item inventory.
locationType	Yes	String (1)	Type of item inventory location. Valid values are S (store) and W (warehouse).
channelId	No	Number (4)	Only used when a physical warehouse is passed. If a channel is passed the results will be limited by the passed-in channel.

Sample Input Message

```
{
  "items": [
    "String"
  ],
  "inventoryLocations": [
    {
      "location": 6000,
      "locationType": "S",
      "channelId": 1
    }
  ]
}
```

Response Code: 200 (Success)

GetResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	A collection of items and locations for which inventory must be retrieved.

Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item identifier.
location	Yes	Number (10)	Location where the item is to be found. This field may contain a store or a warehouse.
locationType	Yes	String (1)	Type of location. Valid values are S (store) and W (warehouse).
channelId	No	Number (4)	Channel identifier of the warehouse. If the location is a store, channelId will be blank. If the location is a warehouse, this is optional, based on the input criteria.
availableQuantity	Yes	Number (12,4)	Contains the available quantity for the item/location in the standard unit of measure.
uom	Yes	String (4)	Contains the unit of measure for available quantity.
packCalculateInd	No	String (1)	Valid for pack item only. Indicates whether the pack item's available quantity is actual a held value (N) or calculated using pack component (Y). Calculated pack quantity should be treated as an estimated value.

Sample Response Message

```
{
  "items": [
    {
      "item": "100100076",
      "location": 6000,
      "locationType": "S",
      "channelId": 1,
      "availableQuantity": 3,
      "uom": "EA",
      "packCalculateInd": "Y"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` is present when the input payload or input parameters do not match the schema definition for this service. The element `businessError` is present when the payload passes the schema validation but an exception is caught while processing the business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Customer Order Inventory Backorder Upload Service

The following services are included in this functional area:

- [Create Inventory Backorder](#)

Create Inventory Backorder

Functional Area

Customer Orders

Business Overview

Retailers selling through e-commerce channels often take customer orders even if inventory is not available immediately, with the expectation of future inventory being available to fill the order. If an order is captured against future inventory by the Order Management System (OMS), then a backorder message is sent to Merchandising through this service. The location which will be back ordered is decided by OMS.

This web service updates the backorder quantity in Merchandising - increasing when the backorder is taken and decreasing when the backorder is released for fulfillment or cancellation. Backorders can be taken against both stores and warehouses.

Service Type

POST

ReST URL

MerchIntegrations/services/customerOrders/inventoryBackorder/create

Input Payload Details

Table 5-99 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	No	Collection of Object	A collection of items for which backorder quantity has to be updated.

Table 5-100 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item identifier. An item does not need to have an open purchase order in order to increase backorder quantity. Catchweight, Transformable, Consignment, Concession and Deposit container items are not supported for backorder requests. The item should not be in discontinued status at the specified location.
locationType	Yes	String (1)	Type of location. Valid values are 'S' (store) and 'W' (warehouse).
location	Yes	Number (10)	Location where the item back order inventory has to be updated. This field may contain a store or a physical warehouse.
channelId	No	Number (4)	Channel id for the Physical Warehouse.

Table 5-100 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
backorderQuantity	Yes	Number (12,4)	Contains the Back Order quantity to be adjusted for the item/location.
uom	No	String (4)	Contains the unit of measure for Back Order quantity. If not defined, back orderable quantity is assumed to be in standard unit of measure.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100050005",
      "locationType": "S",
      "location": 1521,
      "channelId": 1,
      "backorderQuantity": 3,
      "uom": "EA"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

```

    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Customer Order Item Substitution Upload Service

The following services are included in this functional area:

- [Create Customer Order Item Substitution](#)

Create Customer Order Item Substitution

Functional Area

Customer Orders

Business Overview

When a store is picking inventory in order to fulfill a customer order, if the inventory of the item ordered does not meet the required quality standards or is unavailable and the order indicates that substitutions are allowed for that item, the store may choose to fulfill the order with a substitute item. If that occurs, the Store Inventory Operations Cloud Services (SIOCS) has the ability to substitute items on the customer order with another predefined substitute item. In such cases, it will notify the Order Management System (OMS) via the Stock Order Status message that an alternative item has been pushed into the order.

Based on this notification, OMS updates the customer order and notifies Merchandising with the same details received from SIOCS using this service. Merchandising will then update the inventory and customer order details by removing the reservation for the original item and adding a reservation for the new item. It will also update the cancelled quantity for the original item on the order and add the details for the substituted item, with a cross reference to the original item.

Service Type

POST

ReST URL

MerchIntegrations/services/customerOrders/itemSubstitution/create

Input Payload Details

Table 5-101 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	No	Collection of Object	A collection of CustOrdSubDesc objects.

Table 5-102 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerOrderNo	Yes	String (48)	Holds the master customer order number from OMS.
fulfillOrderNo	Yes	String (48)	Unique number from OMS related to the fulfillment details.
location	Yes	Number (10)	Indicates the store number associated with the customer order. Should be a valid store location number as item substitution is only supported for Store location.
item	Yes	String (25)	Contains the item number for which substitute item is used.
quantity	Yes	Number (12,4)	Contains the revised order quantity of the original item.
uom	Yes	String (4)	Contains the unit of measure for original item order quantity. This will be the standard uom of the item.
details	Yes	Collection of Object	Contains the substitute item details.
itemLineNo	No	Number (10)	Indicates the detail item line number on the order being substituted for.

Table 5-103 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
substituteItem	Yes	String (25)	Contains the substitute item number. Catchweight, Transformable, Consignment, Concession and Deposit container items are not supported for customer order item substitution.
substituteItemQuantity	Yes	Number (12,4)	Contains the order quantity for the substitute item.
substituteItemQuantityUom	Yes	String (4)	Contains the unit of measure for the substitute item order quantity. This will be the standard uom for the substitute item.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "customerOrderNo": "180008",
      "fulfillOrderNo": "180008",
      "location": 1311,
      "item": "101550001",
      "quantity": 3,
      "uom": "EA",
```

```

    "details": [
      {
        "substituteItem": "100000198",
        "substituteItemQuantity": 3,
        "substituteItemQuantityUom": "EA"
      }
    ],
    "itemLineNo": 1
  }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Customer Order Upload Service

The following services are included in this functional area:

- [Cancel Fulfillment Order](#)
- [Create Fulfillment Order](#)

Cancel Fulfillment Order

Functional Area

Customer Orders

Business Overview

This service can be used to cancel an existing Customer Order or Transfer within Merchandising based on cancellation requests from an external Order Management System.

Service Type

PUT

ReST URL

MerchIntegrations/services/customerOrders/fulfillmentOrder/cancel

Input Payload Details

Table 5-104 Cancel - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-105 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerOrderNo	Yes	String (48)	Holds the master customer order number from OMS.
fulfillOrderNo	Yes	String (48)	Unique number from OMS related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS.
sourceLocationType	No	String (2)	This would be either 'SU' for supplier, 'ST' for store, or 'WH' for warehouse. This would only be populated for vendor, warehouse or multi-site fulfillment orders.
sourceLocation	No	Number (10)	Indicates the supplier, store or warehouse number associated with sourcing the customer order. Only populated for vendor, warehouse or multi-site fulfillment orders.
fulfillLocationType	No	String (1)	This would be either 'S' (for physical store) or 'V' (for virtual store).

Table 5-105 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fulfillLocation	Yes	Number (10)	Indicates the store or warehouse number associated with fulfilling the customer order. For this release, this should always be populated with a virtual or physical store number.
details	Yes	Collection of Object	Indicates the customer order fulfillment cancellation detail.

Table 5-106 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Indicates the item ordered by the customer.
referenceItem	No	String (25)	Indicates the reference item ordered by the customer. Used only if a specific UPC is ordered. This is supported for vendor drop-ships orders only as we don't support transfer requests for a specific reference item.
cancelQuantityInSuom	Yes	Number (12,4)	Indicates the quantity that should be cancelled from the order in item's standard unit of measure.
standardUom	No	String (4)	Indicates item's standard unit of measure.
transactionUom	No	String (4)	Indicates the original transaction unit of measure the order is placed in.
itemLineNo	No	Number (10)	Indicates the detail item line number on the order.

Sample Input Message

```
{
  "items": [
    {
      "customerOrderNo": "123124",
      "fulfillOrderNo": "123124",
      "sourceLocationType": "SU",
      "sourceLocation": 2400,
      "fulfillLocationType": "S",
      "fulfillLocation": 1521,
      "details": [
        {
          "item": "100100068",
          "referenceItem": "String",
          "cancelQuantityInSuom": 3,
          "standardUom": "EA",
          "transactionUom": "EA",
          "itemLineNo": 1
        }
      ]
    }
  ]
}
```



```

    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Create Fulfillment Order

Functional Area

Customer Orders

Business Overview

This service is used to process Customer Order Fulfillment requests from an external order management system (OMS). If the system option OMS_IND = 'Y', then Merchandising expects to receive customer orders through this API. If the system option PERSIST_CUSTOMER_DATA_IND = 'N', personal information will not be stored in the customer order table in Merchandising.

The confirmation messages will be sent in a collection as part of the response object.

- Accepts a collection of fulfillment orders as input. If one order fails, the entire service call fails, and no orders will be created.
- Returns Failure status as part of the response object in the web service call if customer orders are not created due to validation errors.
- Returns Success status and a confirmation message as part of the response object of type:
 - X if customer orders are not created due to lack of inventory
 - P if customer orders are partially created due to insufficient inventory
 - C if customer orders are completely created, when sufficient inventory is available.
 - R if inventory reservations are created.
- A confirmation type of C and P will be accompanied by a confirmation number, which is the transfer or purchase order number created.

The Customer Order Fulfillment message staged will go through a process of validation. Records that pass validation will create new customer order records. If any validation error occurs, transaction will be rolled back, and no customer orders will be created.

There are two scenarios where a customer order fulfillment request cannot be created in Merchandising:

1. Due to data validation errors (e.g., invalid item).
2. Due to 'No Inventory' - There is not enough inventory available at the source location or item is not ranged or inactive at the source location, or item is not supplied by the supplier (in a PO scenario).

Notes:

1. Non-stockholding franchise stores cannot part of a fulfillment order, either as a sourcing location or as a fulfillment location.
2. Catch weight and transformable sellable items are not supported in this integration. To sell items that can vary by weight, like bananas through online channels, setup should be done as a regular (non-catch weight) item with a unit cost and standard UOM defined in items of eaches.
3. It is assumed that customer orders will be captured in the selling UOM in OMS, but that all transactions will be communicated to Merchandising in standard UOM.
4. If the same customer order fulfillment request is sent for a different item or for an existing item but with a different item line number, the existing PO or transfer will be updated.

The Customer Order Fulfillment messages contain information such as delivery type, source type and destination type. Based on these, the system will proceed to create a Purchase Order, Transfer or Inventory Reservation.

Service Type

POST

ReST URL

MerchIntegrations/services/customerOrders/fulfillmentOrder/create

Input Payload Details

Table 5-107 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-108 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerOrderNo	Yes	String (48)	Holds the master customer order number from OMS.
fulfillOrderNo	Yes	String (48)	Unique number from OMS related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS.
sourceLocationType	No	String (2)	This would be either 'SU' for supplier, 'ST' for store, or 'WH' for warehouse. This would only be populated for vendor, warehouse or multi-site fulfillment orders.
sourceLocation	No	Number (10)	Indicates the supplier, store or warehouse number associated with sourcing the customer order. Only populated for vendor, warehouse or multi-site fulfillment orders.
fulfillLocationType	No	String (1)	This would be either 'S' (for physical store) or 'V' (for virtual store).
fulfillLocation	Yes	Number (10)	Indicates the store or warehouse number associated with fulfilling the customer order. For this release, this should always be populated with a virtual or physical store number.
partialDeliveryInd	Yes	String (1)	Indicates if the order can be picked and shipped partially ('N') or if it should be shipped only when complete ('Y').
deliveryType	No	String (1)	Indicates the fulfillment method - ship to customer or store pickup. Expected values are 'S' (ship direct) and 'C' (customer pickup).
carrierCode	No	String (4)	Indicates the carrier the order is to be shipped with, if specified on the order.
carrierServiceCode	No	String (6)	Indicates the method that was selected for shipping by the customer placing the order (e. g. Standard Shipping, Overnight, etc.).

Table 5-108 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
consumerDeliveryDate	Yes	date	Indicates the desired date the delivery is required by the customer. This will be the GMT time.
consumerDeliveryTime	No	date-time	Indicates the desired time the delivery is required by the customer. This will be the GMT time.
deliveryCharges	No	Number (20,4)	Contains the delivery charges on drop ship. Mostly relevant for Brazil.
deliveryChargesCurrency	No	String (3)	Contains the currency of the delivery charges.
comments	No	String (2000)	Any comments sent by OMS about the order.
customerDetails	No	Record	Referenced object for customer information related to the order.
details	Yes	Collection of Object	References a collection of customer order fulfillment details.
orderPlacedStore	No	Number (10)	Indicates the store number associated with the location that the customer order was placed. For on-line orders this will contain the virtual store number associated with the on-line store. For orders captured in a physical store this will contain the store number for the physical store.

Table 5-109 CustomerDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerNo	No	String (14)	Indicates the number that uniquely identifies the customer in OMS.
deliverFirstName	No	String (120)	Contains the first name for the delivery address on the order.
deliverPhoneticFirst	No	String (120)	Contains the phonetic first name for the delivery address on the order.
deliverLastName	No	String (120)	Contains the last name for the delivery address on the order.
deliverPhoneticLast	No	String (120)	Contains the phonetic last name for the delivery address on the order.
deliverPreferredName	No	String (120)	Contains the preferred name for the delivery address on the order.
deliverCompanyName	No	String (120)	Contains the company name for the delivery address on the order.
deliverAdd1	No	String (240)	First line of the delivery address for the customer.
deliverAdd2	No	String (240)	Second line of the delivery address for the customer.

Table 5-109 (Cont.) CustomerDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deliverAdd3	No	String (240)	Third line of the delivery address for the customer.
deliverCounty	No	String (250)	County portion of the delivery address.
deliverCity	No	String (120)	City portion of the delivery address.
deliverState	No	String (3)	State portion of the delivery address.
deliverCountryId	No	String (3)	Country portion of the delivery address.
deliverPost	No	String (30)	Postal code portion of the delivery address.
deliverJurisdiction	No	String (10)	Identifies the jurisdiction code for the delivery country-state relationship.
deliverPhone	No	String (20)	Contains the delivery phone number.
deliverEmail	No	String (120)	Contains the delivery email.
billFirstName	No	String (120)	Contains the first name for the billing address on the order.
billPhoneticFirst	No	String (120)	Contains the phonetic first name for the billing address on the order.
billLastName	No	String (120)	Contains the last name for the billing address on the order.
billPhoneticLast	No	String (120)	Contains the phonetic last name for the billing address on the order.
billPreferredName	No	String (120)	Contains the preferred name for the billing address on the order.
billCompanyName	No	String (120)	Contains the company name for the billing address on the order.
billAdd1	No	String (240)	First line of the billing address for the customer.
billAdd2	No	String (240)	Second line of the billing address for the customer.
billAdd3	No	String (240)	Third line of the billing address for the customer.
billCounty	No	String (250)	County portion of the billing address.
billCity	No	String (120)	City portion of the billing address.
billState	No	String (3)	State portion of the billing address.
billCountryId	No	String (3)	Country portion of the billing address.
billPost	No	String (30)	Postal code portion of the billing address.
billJurisdiction	No	String (10)	Identifies the jurisdiction code for the billing country-state relationship.
billPhone	No	String (20)	Contains the billing phone number.
billEmail	No	String (120)	Contains the billing email.

Table 5-109 (Cont.) CustomerDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-110 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-111 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-112 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Indicates the item ordered by the customer.
referenceItem	No	String (25)	Indicates the reference item ordered by the customer. Used only if a specific UPC is ordered. This is supported for vendor drop-ships orders only as we don't support transfer requests for a specific reference item.
orderQuantitySuom	Yes	Number (12,4)	Indicates the quantity of the item ordered by the customer in item's standard unit of measure.
standardUom	No	String (4)	Indicates item's standard unit of measure.
transactionUom	No	String (4)	Indicates the original transaction unit of measure the order is placed in.
substituteInd	Yes	String (1)	Indicates if substitutes are allowed on a customer order. This will only be used by orders passed to SIM.

Table 5-112 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitRetail	No	Number (20,4)	Indicates the unit sales retail of item on the customer order. This will only be used by Brazil orders in case of Warehouse Fulfillment or vendor sourced POs shipped directly to the customer. It is needed for Sales NF generation.
retailCurrency	No	String (3)	Indicates currency of unit retail and total retail. This will only be used by Brazil orders in case of Warehouse Fulfillment or vendor sourced POs shipped directly to the customer. It is needed for Sales NF generation.
comments	No	String (2000)	Used to indicate any special instructions for the item, such as services (monograms, engrave, etc).
itemLineNo	No	Number (10)	Indicates the detail item line number on the order.

Sample Input Message

```
{
  "items": [
    {
      "customerOrderNo": "123124",
      "fulfillOrderNo": "123124",
      "sourceLocationType": "SU",
      "sourceLocation": 2400,
      "fulfillLocationType": "S",
      "fulfillLocation": 1521,
      "partialDeliveryInd": "N",
      "deliveryType": "S",
      "carrierCode": "30",
      "carrierServiceCode": "D",
      "consumerDeliveryDate": "2001-12-31",
      "consumerDeliveryTime": "2001-12-31T23:59:59.123Z",
      "deliveryCharges": 3.55,
      "deliveryChargesCurrency": "USD",
      "comments": "CO for John Scott",
      "customerDetails": {
        "customerNo": "123456",
        "deliverFirstName": "John",
        "deliverPhoneticFirst": "Mr",
        "deliverLastName": "Scott",
        "deliverPhoneticLast": null,
        "deliverPreferredName": null,
        "deliverCompanyName": null,
        "deliverAdd1": "7th Street",
        "deliverAdd2": "2504 Centergate Dr Apt 103",
        "deliverAdd3": null,
        "deliverCounty": "Miramar County",

```

```

        "deliverCity": "Minneapolis",
        "deliverState": "MN",
        "deliverCountryId": "US",
        "deliverPost": "55555-1234",
        "deliverJurisdiction": null,
        "deliverPhone": "1800800800",
        "deliverEmail": "someone@email.com",
        "billFirstName": "John",
        "billPhoneticFirst": "Mr",
        "billLastName": "Scott",
        "billPhoneticLast": null,
        "billPreferredName": null,
        "billCompanyName": null,
        "billAdd1": "7th Street",
        "billAdd2": "2504 Centergate Dr Apt 103",
        "billAdd3": null,
        "billCounty": "Miramar County",
        "billCity": "Minneapolis",
        "billState": "MN",
        "billCountryId": "US",
        "billPost": "55555-1234",
        "billJurisdiction": null,
        "billPhone": "1800800800",
        "billEmail": "johnscott@gmail.com",
        "localizationExtensions": [
            {
                "country": "BR",
                "attributes": [
                    {
                        "name": "SPECIAL_CATEGORY_CODE",
                        "value": "WS_PRC_08",
                        "valueDate": "2001-12-31"
                    }
                ]
            }
        ]
    },
    "details": [
        {
            "item": "100100068",
            "referenceItem": null,
            "orderQuantitySuom": 3,
            "standardUom": "EA",
            "transactionUom": "EA",
            "substituteInd": "N",
            "unitRetail": 12.99,
            "retailCurrency": "USD",
            "comments": "Customer Order Item Details",
            "itemLineNo": 1
        }
    ],
    "orderPlacedStore": 1311
}
]
}

```


Response Code: 200 (Success)

Table 5-113 CreateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-114 CreateResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerOrderNo	Yes	String (48)	Holds the master customer order number from OMS.
fulfillOrderNo	Yes	String (48)	Unique number from OMS related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS.
confirmationType	Yes	String (1)	Contains the confirmation type. 'P' - Order Partially Created; 'X' - Order Could not be Created; 'C' - Order Completely Created, 'R' - Inventory Reservation.
confirmationNo	No	Number (12)	Contains the PO or Transfer number in RMS related to the fulfillment order line. Required if fulfillment status is 'P'.
details	No	Collection of Object	References a collection of customer order confirmation item details. This will be null for confirmation type 'R' - inventory reservation.
fulfillLocation	No	Number (10)	Contains the fulfillment location for the order. This will be null for confirmation type 'R' - inventory reservation.
systemCode	No	String (10)	The code identifying the system associated with the location. This will be null for confirmation type 'R' - inventory reservation.

Table 5-115 CreateResponse.Items.Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Indicates the item ordered by the customer. This will be null for confirmation type 'R' - inventory reservation.

Table 5-115 (Cont.) CreateResponse.Items.Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
referenceItem	No	String (25)	Indicates the reference item ordered by the customer. Used only if a specific UPC is ordered. This is supported for vendor drop-ships orders only as we don't support transfer requests for a specific reference item. This will be null for confirmation type 'R' - inventory reservation.
confirmedQuantity	Yes	Number (12,4)	Indicates the quantity of the item that can be sourced or fulfilled on the order. This will be null for confirmation type 'R' - inventory reservation.
confirmedQuantityUom	Yes	String (4)	Indicates the unit of measure of the confirmation quantity. This will be null for confirmation type 'R' - inventory reservation.
itemLineNo	No	Number (10)	Indicates the detail item line number on the order. This will be null for confirmation type 'R' - inventory reservation.

Sample Response Message

```
{
  "items": [
    {
      "customerOrderNo": "123123",
      "fulfillOrderNo": "123124",
      "confirmationType": "P",
      "confirmationNo": 9999999,
      "details": [
        {
          "item": "100100076",
          "referenceItem": null,
          "confirmedQuantity": 3,
          "confirmedQuantityUom": "EA",
          "itemLineNo": 1
        }
      ],
      "fulfillLocation": 6000,
      "systemCode": "INV"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Deals

The following services are included in this functional area:

- [Manage Complex Deals](#)
- [Manage Fixed Deals](#)
- [Deals Download Service](#)
 - [Get Item Location Deals](#)

Manage Complex Deals

Functional Area

Deals

Business Overview

This service can be used to create, modify and delete all forms of complex deals that are currently supported via the application UI such as Off Invoice, Allowance, Bill Back, Bill Back Rebate, Vendor Funded Markdown, Vendor Funded Promotion, Promotional Consignment Rate and Clearance Consignment Rate.

It would work as an enhanced version of the existing dealupld batch but containing the latest set of enhancements that have taken place in this workflow.

Service Type

PUT

ReST URL

MerchIntegrations/services/deals/complexDeals/manage

Input Payload Details

Manage - Object. See list of elements for detail

Items - Object. See list of elements for detail

Components - Object. See list of elements for detail

ItemLocation - Object. See list of elements for detail

ProofOfPerformance - Object. See list of elements for detail

Fulfillment - Object. See list of elements for detail

Threshold - Object. See list of elements for detail

Promotion - Object. See list of elements for detail

ReferencedPromotion - Object. See list of elements for detail

ManageError - Object. See list of elements for detail

Response Code: 200 (Success)

ManageResponse - Object. See list of elements for detail

ComplexDeals - Object. See list of elements for detail

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

ManageError - Object. See list of elements for detail

Manage Fixed Deals

Functional Area

Deals

Business Overview

This service can be used to create, modify and delete Merchandise as well as non-Merchandise based Fixed Deals.

Service Type

PUT

ReST URL

MerchIntegrations/services/deals/fixedDeals/manage

Input Payload Details

Manage - Object. See list of elements for detail

Items - Object. See list of elements for detail

Taxes - Object. See list of elements for detail

Merchandise - Object. See list of elements for detail

Locations - Object. See list of elements for detail

Promotions - Object. See list of elements for detail

ProofOfPerformance - Object. See list of elements for detail

Fulfillment - Object. See list of elements for detail

ManageError - Object. See list of elements for detail

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "externalReferenceNo": "FIXEDDEALREST_01",
      "dealId": 133588,
      "status": "A",
      "dealDescription": "Fixed Deal",
      "vendorType": "SS",
      "vendorId": null,
      "supplierId": 2400,
      "dealType": "OTHER",
      "dealAmount": 1000,
      "currencyCode": "USD",
      "organizationUnitId": null,
      "merchandiseInd": "Y",
      "nonMerchandiseCode": null,
      "taxInd": "N",
      "taxCode": null,
      "merchandiseIncomeApportionMethod": "R",
      "locationIncomeApportionMethod": "R",
      "recalculateInventoryValueInd": "N",
      "prepaidInd": "N",
      "collectFrequency": "M",
      "collectStartDate": "2001-12-31",
      "collectCount": 1,
      "postingFrequency": null,
      "postingOccurrence": null,
      "invoiceProcessingLogic": "AA",
      "invoiceDocumentType": "C",
      "comments": "Fixed Deal Header via ReST FDEALREST_01",
    }
  ]
}
```

```

"clearComments": "N",
"taxes": [
{
"action": "CREATE",
"taxCode": "S",
"taxBasis": 1000,
"clearTaxBasis": "N",
"taxAmount": 50,
"clearTaxAmount": "N"
}
],
"merchandise": [
{
"action": "CREATE",
"sequenceNo": 1,
"merchandiseLevel": "D",
"dept": 1000,
"class": null,
"subclass": null,
"itemParent": null,
"item": null,
"contributionRatio": null,
"clearContributionRatio": "N",
"locations": [
{
"action": "CREATE",
"locationType": "S",
"location": 1521,
"contributionRatio": null,
"clearContributionRatio": "N"
}
]
}
],
"promotions": [
{
"action": "CREATE",
"promotionId": 1121
}
],
"proofOfPerformance": [
{
"action": "CREATE",
"proofOfPerformanceSequenceNo": 1,
"dealReferenceItem": "100100092",
"type": "ECD",
"duration": 7,
"durationType": "MONTH",
"vendorRecommendedStartDate": "2001-12-31",
"vendorRecommendedEndDate": "2001-12-31",
"plannedStartDate": "2001-12-31",
"plannedEndDate": "2001-12-31",
"comments": "Proof of performance for fixed deal",
"fulfillment": [
{
"action": "CREATE",

```

```

    "fulfillmentSequenceNo": 1,
    "promotionId": 14,
    "startDate": "2001-12-31",
    "endDate": "2001-12-31",
    "comments": "Fulfillment of POP"
  }
]
}
]
}
]
}
}

```

Response Code: 200 (Success)

ManageResponse - Object. See list of elements for detail

FixedDeals - Object. See list of elements for detail

Sample Response Message

```

{
  "fixedDeals": [
    {
      "dealId": 133578,
      "externalReferenceNo": "String"
    }
  ]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

ManageError - Object. See list of elements for detail

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "externalReferenceNo": "String",
      "dealId": 133578,
      "merchSeqNo": 271772,

```

```
"location": 190104,
"taxCode": "String",
"popDefSeqNo": 390833,
"popType": "String",
"popFulfillSeqNo": 195679,
"promotionId": 799212,
"errorMessage": [
  "String"
]
}
]
}
```

Deals Download Service

The following services are included in this functional area:

- [Get Item Location Deals](#)

Get Item Location Deals

Functional Area

Deals

Business Overview

This service allows the Store Inventory system to query Merchandising for deals that are present for a given item/location combination based on a specific date and source (supplier, partner) criteria. Merchandising will respond with details of the off-invoice deals that the item/location are part of based on the date that has been provided. These details will include the active and close dates and the discounts that are present on these deals.

Service Type

POST

ReST URL

MerchIntegrations/services/deals/itemLocation/get

Input Payload Details

Table 5-116 Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sourceId	Yes	Number (10)	This is the source supplier or warehouse.
location	Yes	Number (10)	This contains the destination location of the store order.
locationType	Yes	String (1)	This contains the destination location type of the store order.

Table 5-116 (Cont.) Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the item number in the distro.
notBeforeDate	Yes	date	Contains the first date that the delivery of the order will be accepted.

Sample Input Message

```
{
  "sourceId": 2400,
  "location": 6000,
  "locationType": "S",
  "item": "100100076",
  "notBeforeDate": "2001-12-31"
}
```

Response Code: 200 (Success)

Table 5-117 GetResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	References a collection of store order initiated deals.

Table 5-118 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dealId	Yes	Number (10)	Deal that applies to the order item.
dealType	Yes	String (10)	Contains the Deal Type
activeDate	Yes	date	Date the deal will become active.
closeDate	No	date	Date the deal will end or ended
dealClass	No	String (6)	Identifies the calculation class of the deal component.
limitType	No	String (6)	Identifies whether thresholds will be set up as quantity values, currency amount values or percentages. Deal limit types will be held on the codes table under a code type of DLLT (Deal Component Threshold Limits). It can only be NULL for an M-type deal (vendor funded markdown). Examples of threshold limits are Amount - total value of units (A) and Quantity - number of Units (Q).

Table 5-118 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
valueType	No	String (6)	Identifies whether the discount values associated with the thresholds will be set up as qty values, currency amount values, percentages or fixed amounts. Valid values are Q for qty, A for currency amount, P for percentage or F for fixed amount. Qty threshold deals are only allowed on off-invoice discounts. Deal threshold value types will be held on the codes table under a code type of DLL2. It can only be NULL for an M-type deal (vendor funded markdown).
lowerLimit	Yes	Number (20,4)	Lower limit of the deal component. This is the minimum value that must be met in order to get the specified discount. This value will be either a currency amount or quantity value, depending on the value of the threshold limit type.
upperLimit	Yes	Number (20,4)	Upper limit of the deal component. This is the maximum value for which the specified discount will apply. This value will be either a currency amount or quantity value, depending on the threshold_limit_type of this deal component.
value	Yes	Number (20,4)	Value of the discount that will be given for meeting the specified thresholds for this deal component.

Sample Response Message

```

{
  "items": [
    {
      "dealId": 133578,
      "dealType": "String",
      "activeDate": "2001-12-31",
      "closeDate": "2001-12-31",
      "dealClass": "String",
      "limitType": "String",
      "valueType": "String",
      "lowerLimit": 500,
      "upperLimit": 10000,
      "value": 20
    }
  ]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Financials

The following services are included in this functional area:

- [Budgets](#)
 - [Half Data Budget Upload Service](#)
- [Currency](#)
 - [Currency Download Service](#)
 - [Currency Exchange Rate Download Service](#)
 - [Currency Exchange Rate Upload Service](#)
- [General Ledger](#)
 - [General Ledger Chart of Accounts Upload Service](#)
- [Terms](#)
 - [Freight Term Upload Service](#)
 - [Payment Term Upload Service](#)
- [Transaction Data](#)
 - [Transaction Data Upload Service](#)

Budgets

The following services are included in this functional area:

- [Half Data Budget Upload Service](#)

Half Data Budget Upload Service

Update Half Data Budget

Business Overview

The primary role of this service is to modify half data budgets and send them to Merchandising.

Functional Area

Financials

Modify Half Data Budget

Business Overview

This service modifies half data budget by calling the SVCPROV_HDB package to load input data to the staging tables and then calling the core half data budget package to validate and update data to the HALF_DATA_BUDGET table.

Service Type

Post

Rest URL

financials/HalfDataBudgetREST/modifyHdb

Input Parameters

SvcprovHdbdescRecRDO

Parameter Name	Data Type
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
halfNo	BigDecimal
locType	String
location	BigDecimal
setOfBooksId	BigDecimal
cumMarkonPct	BigDecimal
shrinkagePct	BigDecimal
markdownPct	BigDecimal

JSON Structure

```
[
  {
    "dept": null,
    "class1":null,
    "subclass":null,
    "halfNo": null,
    "locType": null,
    "location": null,
    "setOfBooksId": null,
    "cumMarkonPct": null,
    "shrinkagePct": null,
```

```
        "markdownPct": null
    }
}
```

Output

SvcprovHdbStatusRecRDO

Parameter Name	Data Type
statusMsg	String
hdbErrTbl	List< SvcprovFailHdbRecRDO>

SvcprovFailHdbRecRDO

Parameter Name	Data Type
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
halfNo	BigDecimal
locType	String
location	BigDecimal
setOfBooksId	BigDecimal
errorMsg	String

The output will contain the status of the request including validation errors, if any.

JSON Structure

```
{
  "statusMsg": null,
  " hdbErrTbl ": [
    {
      "dept": null,
      "class1":null,
      "subclass":null,
      "halfNo": null,
      "locType": null,
      "location": null,
      "setOfBooksId": null,
      "errorMsg": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
HALF_DATA_BUGET	Yes	No	Yes	No
SVC_PROCESS_TRACKER	Yes	Yes	Yes	No
SVC_ADMIN_UPLD_ER	Yes	Yes	No	No
SVC_HALF_DATA_BUDGET	Yes	Yes	No	Yes
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Currency

The following services are included in this functional area:

- [Currency Download Service](#)
- [Currency Exchange Rate Download Service](#)
- [Currency Exchange Rate Upload Service](#)

Currency Download Service

Get Currency Definitions

Business Overview

Retrieve Merchandising's Currencies table records.

Service Type

GET

ReST URL

/Common/Currencies

Input Parameters

N/A

Output

Table 5-119 MblCurrenciesRDO

Parameter Name	Data Type
currencyCode	String
currencyDescription	String
currencyCostFormat	String
currencyRetailFormat	String
currencyCostDecimal	BigDecimal
currencyRetailDecimal	BigDecimal

JSON Structure:

```
[
  {
    "links": [],
    "currencyCode": "AED",
    "currencyDescription": "U.A.E. Dirham",
    "currencyCostFormat": "FM9G999G999G999G990D9099PR",
    "currencyRetailFormat": "FM9G999G999G999G990D90PR",
    "currencyCostDecimal": 4,
    "currencyRetailDecimal": 2,
    "hyperMediaContent": {
      "linkRDO": []
    }
  },
  {
    "links": [],
    "currencyCode": "ALL",
    "currencyDescription": "UNKNOWN",
    "currencyCostFormat": "FMD0",
    "currencyRetailFormat": "FMD90",
    "currencyCostDecimal": 2,
    "currencyRetailDecimal": 2,
    "hyperMediaContent": {
      "linkRDO": []
    }
  },
  .....
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CURRENCIES	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Currency Exchange Rate Download Service

Get Currency Exchange Rates

This section describes the Currency Rate service.

Business Overview

This service is used to retrieve all currencies and currency conversion rates. The conversion rate is the value used to convert to the primary currency.

Service Type

Get

ReST URL

CurrencyRates/currencyRates

Input Parameters

N/A

Output

RestCurrencyRatesRecRDO

Parameter Name	Data Type
exchangeRate	BigDecimal
effectiveDate	Timestamp
currencyCode	String
exchangeType	String

JSON Structure

```
[
  {
    "exchangeRate": null,
    "effectiveDate": null,
    "currencyCode": null,
    "exchangeType": null,
    "links": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CURRENCY_RATES	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Currency Exchange Rate Upload Service

Manage Currency Exchange Rates

Functional Area

Financials - Currency

Business Overview

Currency exchange rates constitute financial information that is subscribed to by Merchandising. A currency exchange rate is the price of one country's currency expressed in another country's currency. This webservice assumes that currency codes are already present in Merchandising. This service supports creating new rates by date and updating existing rates for the same conversion date.

Service Type

PUT

ReST URL

MerchIntegrations/services/financials/currency/exchangeRate/manage

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-120 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of currency exchange rate details.

Table 5-121 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromCurrency	Yes	String (3)	Contains the currency code that the rate exchange is from.
toCurrency	Yes	String (3)	Contains the currency code that the rate exchange is to.
conversionDate	Yes	date	Contains the date on which the currency rate became or will become active.
conversionRate	Yes	Number (20,10)	The exchange rate between the two currency codes for the type and effective date.
conversionType	Yes	String (30)	This field identifies the type of exchange rate the history exists for. Valid values are valid exchange types in fif_currency_xref table.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "fromCurrency": "USD",
      "toCurrency": "AUD",
      "conversionDate": "2001-12-31",
      "conversionRate": 3,
      "conversionType": "C"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

General Ledger

The following services are included in this functional area:

- [General Ledger Chart of Accounts Upload Service](#)

General Ledger Chart of Accounts Upload Service

Manage General Ledger Chart of Accounts

Functional Area

Financials - General Ledger

Business Overview

This service is used when Retail Financial Integration (RFI) is not being used and integration with a financial system is a custom or to a non-Oracle financials system. When using RFI with an Oracle Financial system, the chart of accounts (COA) information in Merchandising and Sales Audit is created through a mapping and COA validation process. It is used to create new chart of account entries, as well as update existing entries.

Before Merchandising can publish stock ledger data to an external financial application, it must receive the General Ledger chart of accounts (GLCOA) structure. The chart of accounts is the financial application's debit and credit account segments (for example, company, cost center, account, and so on). These are mapped to the transactions, locations, and product hierarchy in Merchandising when stock ledger data and Sales Audit totals are sent to the General Ledger. In some financial applications, these are known as code combination chart fields. There is also a primary account, in some systems known as a CCID that uniquely identifies the combination

of segment or chart field values. Upon receipt of GLCOA message data, Merchandising populates the data to the FIF_GL_ACCT table.

Service Type

PUT

ReST URL

MerchIntegrations/services/financials/generalLedger/chartOfAccounts/manage

Input Payload Details

Table 5-122 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of GL chart of accounts.

Table 5-123 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAccount	Yes	String (25)	This denotes the primary account for a chart of accounts.
attribute1	No	String (25)	Secondary Account information. A value is required if description1 is supplied.
attribute2	No	String (25)	Secondary Account information. A value is required if description2 is supplied.
attribute3	No	String (25)	Secondary Account information. A value is required if description3 is supplied.
attribute4	No	String (25)	Secondary Account information. A value is required if description4 is supplied.
attribute5	No	String (25)	Secondary Account information. A value is required if description5 is supplied.
attribute6	No	String (25)	Secondary Account information. A value is required if description6 is supplied.
attribute7	No	String (25)	Secondary Account information. A value is required if description7 is supplied.
attribute8	No	String (25)	Secondary Account information. A value is required if description8 is supplied.
attribute9	No	String (25)	Secondary Account information. A value is required if description9 is supplied.

Table 5-123 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
attribute10	No	String (25)	Secondary Account information. A value is required if description10 is supplied.
attribute11	No	String (25)	Secondary Account information. A value is required if description11 is supplied.
attribute12	No	String (25)	Secondary Account information. A value is required if description12 is supplied.
attribute13	No	String (25)	Secondary Account information. A value is required if description13 is supplied.
attribute14	No	String (25)	Secondary Account information. A value is required if description14 is supplied.
attribute15	No	String (25)	Secondary Account information. A value is required if description15 is supplied.
description1	No	String (50)	Description of the attribute1 field. Required if attribute1 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description2	No	String (50)	Description of the attribute2 field. Required if attribute2 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description3	No	String (50)	Description of the attribute3 field. Required if attribute3 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description4	No	String (50)	Description of the attribute4 field. Required if attribute4 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description5	No	String (50)	Description of the attribute5 field. Required if attribute5 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description6	No	String (50)	Description of the attribute6 field. Required if attribute6 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description7	No	String (50)	Description of the attribute7 field. Required if attribute7 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description8	No	String (50)	Description of the attribute8 field. Required if attribute8 is supplied and you are not using Oracle Financials 11.5.10 or greater.

Table 5-123 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
description9	No	String (50)	Description of the attribute9 field. Required if attribute9 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description10	No	String (50)	Description of the attribute10 field. Required if attribute10 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description11	No	String (50)	Description of the attribute11 field. Required if attribute11 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description12	No	String (50)	Description of the attribute12 field. Required if attribute12 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description13	No	String (50)	Description of the attribute13 field. Required if attribute13 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description14	No	String (50)	Description of the attribute14 field. Required if attribute14 is supplied and you are not using Oracle Financials 11.5.10 or greater.
description15	No	String (50)	Description of the attribute15 field. Required if attribute15 is supplied and you are not using Oracle Financials 11.5.10 or greater.
setOfBooksId	No	Number (15)	The Set of books Identifier.
attribute16	No	String (25)	Secondary Account information. A value is required if description16 is supplied.
attribute17	No	String (25)	Secondary Account information. A value is required if description17 is supplied.
attribute18	No	String (25)	Secondary Account information. A value is required if description18 is supplied.
attribute19	No	String (25)	Secondary Account information. A value is required if description19 is supplied.
attribute20	No	String (25)	Secondary Account information. A value is required if description20 is supplied.
description16	No	String (50)	Description of the attribute16 field. Required if attribute16 supplied and you are not using Oracle Financials 11. 5. 10 or greater.

Table 5-123 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
description17	No	String (50)	Description of the attribute17 field. Required if attribute17 supplied and you are not using Oracle Financials 11. 5. 10 or greater.
description18	No	String (50)	Description of the attribute18 field. Required if attribute18 supplied and you are not using Oracle Financials 11. 5. 10 or greater.
description19	No	String (50)	Description of the attribute19 field. Required if attribute19 supplied and you are not using Oracle Financials 11. 5. 10 or greater.
description20	No	String (50)	Description of the attribute20 field. Required if attribute20 supplied and you are not using Oracle Financials 11. 5. 10 or greater.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "primaryAccount": "99338",
      "attribute1": "1001",
      "attribute2": "1002",
      "attribute3": "1003",
      "attribute4": "1004",
      "attribute5": "1005",
      "attribute6": null,
      "attribute7": null,
      "attribute8": null,
      "attribute9": null,
      "attribute10": null,
      "attribute11": null,
      "attribute12": null,
      "attribute13": null,
      "attribute14": null,
      "attribute15": null,
      "description1": "Attribute 1001",
      "description2": "Attribute 1002",
      "description3": "Attribute 1003",
      "description4": "Attribute 1004",
      "description5": "Attribute 1005",
      "description6": null,
      "description7": null,
      "description8": null,
      "description9": null,
      "description10": null,
      "description11": null,
      "description12": null,

```

```

        "description13": null,
        "description14": null,
        "description15": null,
        "setOfBooksId": 111111111111111,
        "attribute16": null,
        "attribute17": null,
        "attribute18": null,
        "attribute19": null,
        "attribute20": null,
        "description16": null,
        "description17": null,
        "description18": null,
        "description19": null,
        "description20": null
    }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Manage Merchandising General Ledger Cross Reference

Functional Area

Financials - General Ledger

Business Overview

This service will aid management of the General Ledger Cross Reference Data in Merchandising in addition to the existing spreadsheet upload. This will allow create, update and delete of records from FIF_GL_CROSS_REF table based on the action.

The account validation offered by this service will be based on the configuration of the 'External Account Validation' system option. If external account validation is opted for and if the account combination specified on the cross-reference record is not found locally (in the FIF_GL_ACCT table), a webservice call will be invoked to validate account combinations in Cloud Financials. (Note: The Account validation service credentials and end point will need to be configured in Merchandising to carry out this service call/validation). If the account is found to be valid via the service call, then the account will be added to the local account repository (FIF_GL_ACCT) for future reference and the cross reference saved successfully.

If External Account Validation is not opted for, then the account validation will be limited to validating against the accounts available locally (in the FIF_GL_ACCT table). Combinations not existing in FIF_GL_ACCT will not be included in the cross-reference table.

Service Type

PUT

ReST URL

MerchIntegrations/services/financials/generalLedger/crossReference/manage

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-124 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	Reference a collection of GL Cross Reference Data.

Table 5-125 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Specifies the intended action to create, update or delete an externally generated location list. Valid values are CREATE, UPDATE or DELETE.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
setOfBooksId	Yes	Number (15)	The Set of books Identifier.
dept	No	Number (4)	Oracle Retail department. A value of -1 will be used as a wildcard to mean all departments.
class	No	Number (4)	Oracle Retail Class. A value of -1 will be used as a wildcard to mean all classes.
subclass	No	Number (4)	Oracle Retail Subclass. A value of -1 will be used as a wildcard to mean all subclasses.
location	No	Number (10)	Oracle Retail Location. A value of -1 will be used as a wildcard to mean all locations.
transactionCode	Yes	Number (4)	Oracle Retail Transaction code. A value of -1 will be used as a wildcard to mean all tran codes. A tran code of zero will be used to refer to purchase order accounts.
costRetailInd	Yes	String (1)	Cost C or Retail R Flag.
lineType	Yes	String (6)	This indicates whether the account code of this record is for an Item, Freight, Tax, or Miscellaneous. This information will be necessary when choosing which Oracle Financials account to associate with each line in the invoice. Without this field, we are only able to choose an account code for an item, not for freight, tax, or miscellaneous charges.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transactionReferenceNo	No	String (25)	Contains the intercompany transfer from/to location (tran code = 37, 38), the work order activity ID (tran code = 63, 64), the inventory adjustment reason (tran code = 22, 23), the vat code (tran code = 87, 88) or -1 as a catch-all value, the merchandise cost mapping code system option, the unmapped cost component mapping code system option or a non-merchandise code found on the non-merchandise codes table (tran code = 20). If the tran code is Intercompany Out, the field will contain the transfer To Location. If the tran code is Intercompany, the field will contain the transfer From Location. If the tran code is sales or return (Tran codes 1,2,3,4,5), the field will contain the posting classifier. If the tran code is sales - based deal (tran code 6), the field will contain the deal billing type .
debitAccount	No	Number (18)	Oracle debit account combination code ID, determined from the 10 sequence numbers.
debitSegment1	No	String (25)	Oracle debit account mapping field1, maps to Oracle side GL_Code_combinations table.
debitSegment2	No	String (25)	Oracle debit account mapping field2, maps to Oracle side GL_Code_combinations table.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
debitSegment3	No	String (25)	Oracle debit account mapping field3, maps to Oracle side GL_Code_combinations table.
debitSegment4	No	String (25)	Oracle debit account mapping field4, maps to Oracle side GL_Code_combinations table.
debitSegment5	No	String (25)	Oracle debit account mapping field5, maps to Oracle side GL_Code_combinations table.
debitSegment6	No	String (25)	Oracle debit account mapping field6, maps to Oracle side GL_Code_combinations table.
debitSegment7	No	String (25)	Oracle debit account mapping field7, maps to Oracle side GL_Code_combinations table.
debitSegment8	No	String (25)	Oracle debit account mapping field8, maps to Oracle side GL_Code_combinations table.
debitSegment9	No	String (25)	Oracle debit account mapping field9, maps to Oracle side GL_Code_combinations table.
debitSegment10	No	String (25)	Oracle debit account mapping field10, maps to Oracle side GL_Code_combinations table.
debitSegment11	No	String (25)	Oracle debit account mapping field11, maps to Oracle side GL_Code_combinations table.
debitSegment12	No	String (25)	Oracle debit account mapping field12, maps to Oracle side GL_Code_combinations table.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
debitSegment13	No	String (25)	Oracle debit account mapping field13, maps to Oracle side GL_Code_combinations table.
debitSegment14	No	String (25)	Oracle debit account mapping field14, maps to Oracle side GL_Code_combinations table.
debitSegment15	No	String (25)	Oracle debit account mapping field15, maps to Oracle side GL_Code_combinations table.
debitSegment16	No	String (25)	Oracle debit account mapping field16, maps to Oracle side GL_Code_combinations table.
debitSegment17	No	String (25)	Oracle debit account mapping field17, maps to Oracle side GL_Code_combinations table.
debitSegment18	No	String (25)	Oracle debit account mapping field18, maps to Oracle side GL_Code_combinations table.
debitSegment19	No	String (25)	Oracle debit account mapping field19, maps to Oracle side GL_Code_combinations table.
debitSegment20	No	String (25)	Oracle debit account mapping field20, maps to Oracle side GL_Code_combinations table.
creditAccount	No	Number (18)	Oracle credit account combination code ID, determined from the 10 sequence numbers.
creditSegment1	No	String (25)	Oracle credit account mapping field1, maps to Oracle side GL_Code_combinations table.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
creditSegment2	No	String (25)	Oracle credit account mapping field2, maps to Oracle side GL_Code_combinations table.
creditSegment3	No	String (25)	Oracle credit account mapping field3, maps to Oracle side GL_Code_combinations table.
creditSegment4	No	String (25)	Oracle credit account mapping field4, maps to Oracle side GL_Code_combinations table.
creditSegment5	No	String (25)	Oracle credit account mapping field5, maps to Oracle side GL_Code_combinations table.
creditSegment6	No	String (25)	Oracle credit account mapping field6, maps to Oracle side GL_Code_combinations table.
creditSegment7	No	String (25)	Oracle credit account mapping field7, maps to Oracle side GL_Code_combinations table.
creditSegment8	No	String (25)	Oracle credit account mapping field8, maps to Oracle side GL_Code_combinations table.
creditSegment9	No	String (25)	Oracle credit account mapping field9, maps to Oracle side GL_Code_combinations table.
creditSegment10	No	String (25)	Oracle credit account mapping field10, maps to Oracle side GL_Code_combinations table.
creditSegment11	No	String (25)	Oracle credit account mapping field11, maps to Oracle side GL_Code_combinations table.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
creditSegment12	No	String (25)	Oracle credit account mapping field12, maps to Oracle side GL_Code_combinations table.
creditSegment13	No	String (25)	Oracle credit account mapping field13, maps to Oracle side GL_Code_combinations table.
creditSegment14	No	String (25)	Oracle credit account mapping field14, maps to Oracle side GL_Code_combinations table.
creditSegment15	No	String (25)	Oracle credit account mapping field15, maps to Oracle side GL_Code_combinations table.
creditSegment16	No	String (25)	Oracle credit account mapping field16, maps to Oracle side GL_Code_combinations table.
creditSegment17	No	String (25)	Oracle credit account mapping field17, maps to Oracle side GL_Code_combinations table.
creditSegment18	No	String (25)	Oracle credit account mapping field18, maps to Oracle side GL_Code_combinations table.
creditSegment19	No	String (25)	Oracle credit account mapping field19, maps to Oracle side GL_Code_combinations table.
creditSegment20	No	String (25)	Oracle credit account mapping field20, maps to Oracle side GL_Code_combinations table.

Table 5-125 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
baseInd	No	String (1)	This column indicates if the record will be used as priority for cross reference mapping with dynamic segments. Valid values are 'Y' or Null.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "setOfBooksId": 1,
      "dept": 209,
      "class": -1,
      "subclass": -1,
      "location": 265080,
      "transactionCode": 30,
      "costRetailInd": "C",
      "lineType": "ITEM",
      "transactionReferenceNo": "265109",
      "debitAccount": 1922,
      "debitSegment1": "KS LLC",
      "debitSegment2": "123100",
      "debitSegment3": "32520",
      "debitSegment4": "0",
      "debitSegment5": null,
      "debitSegment6": null,
      "debitSegment7": null,
      "debitSegment8": null,
      "debitSegment9": null,
      "debitSegment10": null,
      "debitSegment11": null,
      "debitSegment12": null,
      "debitSegment13": null,
      "debitSegment14": null,
      "debitSegment15": null,
      "debitSegment16": null,
      "debitSegment17": null,
      "debitSegment18": null,
      "debitSegment19": null,
      "debitSegment20": null,
      "creditAccount": 498,
      "creditSegment1": "KS LLC",
      "creditSegment2": "123200",
      "creditSegment3": "32520",
      "creditSegment4": "0",
    }
  ]
}
```

```

"creditSegment5": null,
"creditSegment6": null,
"creditSegment7": null,
"creditSegment8": null,
"creditSegment9": null,
"creditSegment10": null,
"creditSegment11": null,
"creditSegment12": null,
"creditSegment13": null,
"creditSegment14": null,
"creditSegment15": null,
"creditSegment16": null,
"creditSegment17": null,
"creditSegment18": null,
"creditSegment19": null,
"creditSegment20": null,
"baseInd": null
}
]
}

```

Response Code: 200 (Success)

Table 5-126 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
message	Yes	String (250)	Indicates processing outcome and the message is defined by code_detail code_type IIPS. Default configured values are Processed Successfully or Processed with errors or Processed with warnings or Pending validation from external Financial System.
errors	Yes	Collection of Object	References a collection non-successful Cross Reference Data with its corresponding errors. This will be empty when all the records get successfully processed. In case of multiple GL Cross Reference data in the input and there are one/multiple/all failures, the error/warning reasons for each of the non-successful Cross Reference data will be captured here.

Table 5-127 Errors - Object. See list of elements for detail

Element Name	Required	Data Type	Description
setOfBooksId	Yes	Number (15)	The Set of books Identifier.
dept	Yes	Number (4)	Oracle Retail department. A value of -1 will be used as a wildcard to mean all departments.
class	Yes	Number (4)	Oracle Retail Class. A value of -1 will be used as a wildcard to mean all classes.
subclass	Yes	Number (4)	Oracle Retail Subclass. A value of -1 will be used as a wildcard to mean all subclasses.
location	Yes	Number (10)	Oracle Retail Location. A value of -1 will be used as a wildcard to mean all locations.
transactionCode	Yes	Number (4)	Oracle Retail Transaction code. A value of -1 will be used as a wildcard to mean all tran codes. A tran code of zero will be used to refer to purchase order accounts.
costRetailInd	Yes	String (1)	Cost C or Retail R Flag.
lineType	Yes	String (6)	This indicates whether the account code of this record is for an Item, Freight, Tax, or Miscellaneous. This information will be necessary when choosing which Oracle Financials account to associate with each line in the invoice. Without this field, we are only able to choose an account code for an item, not for freight, tax, or miscellaneous charges.

Table 5-127 (Cont.) Errors - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transactionReferenceNo	Yes	String (25)	Contains the intercompany transfer from/to location (tran code = 37, 38), the work order activity ID (tran code = 63, 64), the inventory adjustment reason (tran code = 22, 23), the vat code (tran code = 87, 88) or -1 as a catch-all value, the merchandise cost mapping code system option, the unmapped cost component mapping code system option or a non-merchandise code found on the non-merchandise codes table (tran code = 20). If the tran code is Intercompany Out, the field will contain the transfer To Location. If the tran code is Intercompany, the field will contain the transfer From Location. If the tran code is sales or return (Tran codes 1,2,3,4,5), the field will contain the posting classifier. If the tran code is sales-based deal (tran code 6), the field will contain the deal billing type.
errorMessage	Yes	Array of String	References a collection of errors for a given setOfBooksId , dept, class, subclass, location, transactionCode , costRetailInd , lineType , transactionReferenceNo record.

Sample Response Message

```
{
  "message": "Processed Successfully",
}
```

```

"errors": [
{
"setOfBooksId": null,
"dept": null,
"class": null,
"subclass": null,
"location": null,
"transactionCode": null,
"costRetailInd": null,
"lineType": null,
"transactionReferenceNo": null,
"errorMessage": [
>null"
]
}
]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Manage Sales Audit General Ledger Cross Reference

Functional Area

Sales Audit - Financials

Business Overview

This service will aid management of the General Ledger Cross Reference Data in Sales Audit in addition to the existing spreadsheet upload. This will allow create, update and delete of records from SA_FIF_GL_CROSS_REF table based on the action.

The account validation offered by this service will be based on the configuration of the 'External Account Validation' system option. If external account validation is opted for and if the account combination specified on the cross-reference record is not found locally (in the FIF_GL_ACCT table), a webservice call will be invoked to validate account combinations in Cloud Financials. (Note: The Account validation service credentials and end point will need to

be configured in Merchandising to carry out this service call/validation). If the account is found to be valid via the service call, then the account will be added to the local account repository (FIF_GL_ACCT) for future reference and the cross reference saved successfully.

If External Account Validation is not opted for, then the account validation will be limited to validating against the accounts available locally (in the FIF_GL_ACCT table). Combinations not existing in FIF_GL_ACCT will not be included in the cross-reference table.

Service Type

PUT

ReST URL

MerchIntegrations/services/resa/glCrossReference/manage

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-128 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	Reference a collection of GL Cross Reference Data.

Table 5-129 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Specifies the intended action to create, update or delete an externally generated location list. Valid values are CREATE, UPDATE or DELETE.
store	Yes	Number (10)	This column indicates the location the account code is valid for.
total	Yes	String (10)	This column indicates the type of total the account is valid for.
setOfBooksId	Yes	Number (15)	Oracle set of books for Oracle Retail transactions.
rollupLevel1	Yes	String (120)	This column indicates the first rollup level of total.
rollupLevel2	Yes	String (120)	This column indicates the second rollup level of total.
rollupLevel3	Yes	String (120)	This column indicates the third rollup level of total.
debitAccount	No	Number (18)	Oracle debit account combination code ID, determined from the 10 sequence numbers.

Table 5-129 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
debitSegment1	No	String (25)	Oracle debit account mapping field1, maps to Oracle side GL_Code_combinations table.
debitSegment2	No	String (25)	Oracle debit account mapping field2, maps to Oracle side GL_Code_combinations table.
debitSegment3	No	String (25)	Oracle debit account mapping field3, maps to Oracle side GL_Code_combinations table.
debitSegment4	No	String (25)	Oracle debit account mapping field4, maps to Oracle side GL_Code_combinations table.
debitSegment5	No	String (25)	Oracle debit account mapping field5, maps to Oracle side GL_Code_combinations table.
debitSegment6	No	String (25)	Oracle debit account mapping field6, maps to Oracle side GL_Code_combinations table.
debitSegment7	No	String (25)	Oracle debit account mapping field7, maps to Oracle side GL_Code_combinations table.
debitSegment8	No	String (25)	Oracle debit account mapping field8, maps to Oracle side GL_Code_combinations table.
debitSegment9	No	String (25)	Oracle debit account mapping field9, maps to Oracle side GL_Code_combinations table.
debitSegment10	No	String (25)	Oracle debit account mapping field10, maps to Oracle side GL_Code_combinations table.
debitSegment11	No	String (25)	Oracle debit account mapping field11, maps to Oracle side GL_Code_combinations table.
debitSegment12	No	String (25)	Oracle debit account mapping field12, maps to Oracle side GL_Code_combinations table.
debitSegment13	No	String (25)	Oracle debit account mapping field13, maps to Oracle side GL_Code_combinations table.
debitSegment14	No	String (25)	Oracle debit account mapping field14, maps to Oracle side GL_Code_combinations table.
debitSegment15	No	String (25)	Oracle debit account mapping field15, maps to Oracle side GL_Code_combinations table.
debitSegment16	No	String (25)	Oracle debit account mapping field16, maps to Oracle side GL_Code_combinations table.
debitSegment17	No	String (25)	Oracle debit account mapping field17, maps to Oracle side GL_Code_combinations table.

Table 5-129 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
debitSegment18	No	String (25)	Oracle debit account mapping field18, maps to Oracle side GL_Code_combinations table.
debitSegment19	No	String (25)	Oracle debit account mapping field19, maps to Oracle side GL_Code_combinations table.
debitSegment20	No	String (25)	Oracle debit account mapping field20, maps to Oracle side GL_Code_combinations table.
creditAccount	No	Number (18)	Oracle credit account combination code ID, determined from the 10 sequence numbers.
creditSegment1	No	String (25)	Oracle credit account mapping field1, maps to Oracle side GL_Code_combinations table.
creditSegment2	No	String (25)	Oracle credit account mapping field2, maps to Oracle side GL_Code_combinations table.
creditSegment3	No	String (25)	Oracle credit account mapping field3, maps to Oracle side GL_Code_combinations table.
creditSegment4	No	String (25)	Oracle credit account mapping field4, maps to Oracle side GL_Code_combinations table.
creditSegment5	No	String (25)	Oracle credit account mapping field5, maps to Oracle side GL_Code_combinations table.
creditSegment6	No	String (25)	Oracle credit account mapping field6, maps to Oracle side GL_Code_combinations table.
creditSegment7	No	String (25)	Oracle credit account mapping field7, maps to Oracle side GL_Code_combinations table.
creditSegment8	No	String (25)	Oracle credit account mapping field8, maps to Oracle side GL_Code_combinations table.
creditSegment9	No	String (25)	Oracle credit account mapping field9, maps to Oracle side GL_Code_combinations table.
creditSegment10	No	String (25)	Oracle credit account mapping field10, maps to Oracle side GL_Code_combinations table.
creditSegment11	No	String (25)	Oracle credit account mapping field11, maps to Oracle side GL_Code_combinations table.
creditSegment12	No	String (25)	Oracle credit account mapping field12, maps to Oracle side GL_Code_combinations table.
creditSegment13	No	String (25)	Oracle credit account mapping field13, maps to Oracle side GL_Code_combinations table.

Table 5-129 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
creditSegment14	No	String (25)	Oracle credit account mapping field14, maps to Oracle side GL_Code_combinations table.
creditSegment15	No	String (25)	Oracle credit account mapping field15, maps to Oracle side GL_Code_combinations table.
creditSegment16	No	String (25)	Oracle credit account mapping field16, maps to Oracle side GL_Code_combinations table.
creditSegment17	No	String (25)	Oracle credit account mapping field17, maps to Oracle side GL_Code_combinations table.
creditSegment18	No	String (25)	Oracle credit account mapping field18, maps to Oracle side GL_Code_combinations table.
creditSegment19	No	String (25)	Oracle credit account mapping field19, maps to Oracle side GL_Code_combinations table.
creditSegment20	No	String (25)	Oracle credit account mapping field20, maps to Oracle side GL_Code_combinations table.
attribute3Map	No	String (30)	This column will store the metadata information on which field can be mapped to the STG_FIF_GL_DATA.ATTRIBUTE3 column for a specific GL cross reference mapping.
attribute4Map	No	String (30)	This column will store the metadata information on which field can be mapped to the STG_FIF_GL_DATA.ATTRIBUTE4 column for a specific GL cross reference mapping.
attribute5Map	No	String (30)	This column will store the metadata information on which field can be mapped to the STG_FIF_GL_DATA.ATTRIBUTE5 column for a specific GL cross reference mapping.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "store": 320,
      "total": "TENDER",
      "setOfBooksId": 2021,
      "rollupLevel1": "44",
      "rollupLevel2": "-1",

```

```

    "rollupLevel3": "-1",
    "debitAccount": null,
    "debitSegment1": "1",
    "debitSegment2": "1",
    "debitSegment3": "0",
    "debitSegment4": "0",
    "debitSegment5": "10003",
    "debitSegment6": "52028",
    "debitSegment7": "0",
    "debitSegment8": "0",
    "debitSegment9": null,
    "debitSegment10": null,
    "debitSegment11": null,
    "debitSegment12": null,
    "debitSegment13": null,
    "debitSegment14": null,
    "debitSegment15": null,
    "debitSegment16": null,
    "debitSegment17": null,
    "debitSegment18": null,
    "debitSegment19": null,
    "debitSegment20": null,
    "creditAccount": null,
    "creditSegment1": "1",
    "creditSegment2": "1",
    "creditSegment3": "0",
    "creditSegment4": "0",
    "creditSegment5": "10003",
    "creditSegment6": "52045",
    "creditSegment7": "0",
    "creditSegment8": "0",
    "creditSegment9": null,
    "creditSegment10": null,
    "creditSegment11": null,
    "creditSegment12": null,
    "creditSegment13": null,
    "creditSegment14": null,
    "creditSegment15": null,
    "creditSegment16": null,
    "creditSegment17": null,
    "creditSegment18": null,
    "creditSegment19": null,
    "creditSegment20": null,
    "attribute3Map": "ROLLUP_LEVEL_1",
    "attribute4Map": null,
    "attribute5Map": null
  }
]
}

```


Response Code: 200 (Success)

Table 5-130 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
message	Yes	String (30)	Indicates processing outcome and the message is defined by code_detail code_type IIPS. Default configured values are Processed Successfully or Processed with errors or Processed with warnings or Pending validation from external Financial System.
errors	Yes	Collection of Object	References a collection non-successful Cross Reference Data with its corresponding errors. This will be empty when all the records get successfully processed. In case of multiple GL Cross Reference data in the input and there are one/multiple/all failures, the error/warning reasons for each of the non-successful Cross Reference data will be captured here.

Table 5-131 Errors - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	This column indicates the location the account code is valid for
total	Yes	String (10)	This column indicates the type of total the account is valid for.
rollupLevel1	Yes	String (120)	This column indicates the first rollup level of total.
rollupLevel2	Yes	String (120)	This column indicates the second rollup level of total.
rollupLevel3	Yes	String (120)	This column indicates the third rollup level of total.
setOfBooksId	Yes	Number (15)	Oracle set of books for Oracle Retail transactions.
errorMessage	Yes	Array of String	References a collection of errors for a given setOfBooksId, dept, class, subclass, location, transactionCode, costRetailInd, lineType, transactionReferenceNo record.

Sample Response Message

```
{
  "message": "Processed Successfully",
  "errors": [
    {
      "store": null,
      "total": null,
      "rollupLevel1": null,
      "rollupLevel2": null,
```

```

        "rollupLevel3": null,
        "setOfBooksId": null,
        "errorMessage": [
            "null"
        ]
    }
]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Terms

The following services are included in this functional area:

- [Freight Term Upload Service](#)
- [Payment Term Upload Service](#)

Freight Term Upload Service

Manage Freight Term

Functional Area

Financials - Terms

Business Overview

This service is exposed to allow an external financial system to manage freight terms in Merchandising. Freight Terms are financial arrangement information related to shipping that can be subscribed to by Merchandising from a financial system. Freight terms are the terms for shipping - for example, the freight terms could be a certain percentage of the total cost; a flat fee per order, etc. After confirming the validity of the records enclosed within the message, Merchandising is updated with the information.

Both new records and updates to already existing ones are sent through this single service. Freight Terms and external reference ID are optional fields. If both are empty or the external reference ID does not map to an existing record, a new ID will be generated to create the freight term. If both are passed, Freight Terms will be used to create or update the record. If

only the external reference ID is provided and more than one matching record is found, service will raise an error.

When a new freight term is created, this service will validate that all required fields such as description, enabled flag, start and end dates are present in the message. After this field validation is completed successfully, the freight term record in the message will be inserted if the term does not exist. If the freight term exists, then the external reference ID, dates and enabled flag will be updated.

Service Type

PUT

ReST URL

MerchIntegrations/services/financials/terms/freightTerm/manage

Input Payload Details

Table 5-132 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	Contains a number that uniquely identifies the freight terms.
externalReferenceId	No	String (50)	Contains the ID for the freight terms used in the external financial system.
description	Yes	String (240)	Contains a description of the freight terms used in the system.
enabledFlag	Yes	String (1)	Indicates whether the freight terms are valid or invalid within the respective application. Valid values are 'Y'es or 'N'o. If interface is from Oracle Financials, the Check_Enabled logic will be bypassed which means field would not be required.
startActiveDate	No	date	Indicates the date for assigning an active date to the Freight Terms.
endActiveDate	No	date	Indicates the date for assigning an inactive date to the Freight Terms.

Sample Input Message

```
{
  "freightTerms": "FTTERM01",
  "externalReferenceId": "FTTERM55",
```

```

    "description": "Freight Term 01",
    "enabledFlag": "Y",
    "startActiveDate": "2001-12-31",
    "endActiveDate": "2001-12-31"
  }

```

Response Code: 200 (Success)

Table 5-133 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	Yes	String (10)	Status code for a successful service call.
freightTerms	Yes	String (30)	Contains a number that uniquely identifies the freight terms.
externalReferenceId	No	String (50)	Contains the ID for the freight terms used in the external financial system.

Sample Response Message

```

{
  "status": "SUCCESS",
  "freightTerms": "FTTERM01",
  "externalReferenceId": "FTTERM55"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Payment Term Upload Service

Create Payment Term

Functional Area

Financials - Terms

Business Overview

This service is used by Oracle Retail Financial Integration (RFI) for integration of payment terms with PeopleSoft Financials, and can also be used by an external financial system to send new payment terms information to Merchandising. Terms and external reference Id are optional fields. If Terms id is empty, a new Id will be generated to create the payment term.

Service Type

POST

ReST URL

MerchIntegrations/services/financials/terms/paymentTerm/create

Input Payload Details

Table 5-134 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
terms	No	String (15)	Unique id for this payment term.
externalReferenceId	No	String (50)	ID for the supplier terms used in the external financial system.
termsCode	Yes	String (50)	The alpha value which acts as the Term code in Oracle Financials.
description	Yes	String (240)	Description of the payment terms.
rank	No	Number (10)	Unique rank to rate invoice payment terms against PO terms
details	No	Collection of Object	Child node.

Table 5-135 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
termsSequence	Yes	Number (10)	Sequence for the discount percent and part of unique identifier for the detail line.
dueDays	Yes	Number (3)	The number of days until payment is due.
dueMaximumAmount	Yes	Number (12,4)	Maximum payment amount due by a certain date.
dueDayOfMonth	Yes	Number (2)	Day of month used to calculate due date.
discountDays	Yes	Number (3)	Number of days in which payment must be made to get the discount.
percent	Yes	Number (12,4)	Percentage used to calculate discount available.
discountDayOfMonth	Yes	Number (2)	Day of month used to calculate discount date.
discountMonthForward	Yes	Number (3)	Number of months ahead to calculate discount date.

Table 5-135 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fixedDate	No	date	Fixed due date.
enabledFlag	Yes	String (1)	Indicates whether the Payment terms are valid or invalid within the respective application. The values would be either (Y)es or (N)o.
startActiveDate	No	date	Start date terms are in effect.
endActiveDate	No	date	End date terms are in effect.
dueMonthForward	Yes	Number (3)	Number of months ahead to calculate due date.
cutoffDay	Yes	Number (2)	Day of month after which the following month will be used for calculations.

Sample Input Message

```
{
  "terms": "55",
  "externalReferenceId": "55",
  "termsCode": "55",
  "description": "55 termsDesc",
  "rank": 12,
  "details": [
    {
      "termsSequence": 1,
      "dueDays": 100,
      "dueMaximumAmount": 12.95,
      "dueDayOfMonth": 10,
      "discountDays": 100,
      "percent": 3,
      "discountDayOfMonth": 10,
      "discountMonthForward": 100,
      "fixedDate": "2001-12-31",
      "enabledFlag": "N",
      "startActiveDate": "2001-12-31",
      "endActiveDate": "2001-12-31",
      "dueMonthForward": 100,
      "cutoffDay": 10
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-136 CreateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	Yes	String (10)	Status code for a successful service call.

Table 5-136 (Cont.) CreateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
terms	Yes	String (15)	Unique id for this payment term.
externalReferenceId	No	String (50)	ID for the supplier terms used in the external financial system.

Sample Response Message

```
{
  "status": "SUCCESS",
  "terms": "55",
  "externalReferenceId": "55"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Payment Term Detail**Functional Area**

Financials - Terms

Business Overview

This service can be used by an external financial system to send additional details for existing payment terms to Merchandising. Terms and external reference ID are optional fields, but at least one must be provided when adding details to an existing payment term. If both are passed, Terms will be used to identify the record. If only the external reference ID is provided, it will be used to find a matching payment term. If more than one record is found, service will raise an error.

Service Type

POST

ReST URL

MerchIntegrations/services/financials/terms/paymentTerm/detail/create

Input Payload Details

Table 5-137 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
terms	No	String (15)	Unique id for this payment term.
externalReferenceId	No	String (50)	ID for the supplier terms used in the external financial system.
termsCode	Yes	String (50)	The alpha value which acts as the Term code in Oracle Financials.
description	Yes	String (240)	Description of the payment terms.
rank	No	Number (10)	Unique rank to rate invoice payment terms against PO terms
details	No	Collection of Object	Child node.

Table 5-138 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
termsSequence	Yes	Number (10)	Sequence for the discount percent and part of unique identifier for the detail line.
dueDays	Yes	Number (3)	The number of days until payment is due.
dueMaximumAmount	Yes	Number (12,4)	Maximum payment amount due by a certain date.
dueDayOfMonth	Yes	Number (2)	Day of month used to calculate due date.
discountDays	Yes	Number (3)	Number of days in which payment must be made to get the discount.
percent	Yes	Number (12,4)	Percentage used to calculate discount available.
discountDayOfMonth	Yes	Number (2)	Day of month used to calculate discount date.
discountMonthForward	Yes	Number (3)	Number of months ahead to calculate discount date.
fixedDate	No	date	Fixed due date.
enabledFlag	Yes	String (1)	Indicates whether the Payment terms are valid or invalid within the respective application. The values would be either (Y)es or (N)o.
startActiveDate	No	date	Start date terms are in effect.
endActiveDate	No	date	End date terms are in effect.
dueMonthForward	Yes	Number (3)	Number of months ahead to calculate due date.
cutoffDay	Yes	Number (2)	Day of month after which the following month will be used for calculations.

Sample Input Message

```
{
  "terms": "55",
  "externalReferenceId": "55",
  "termsCode": "55DTLADD",
  "description": "55DESCDTLADD",
  "rank": 14,
  "details": [
    {
      "termsSequence": 2,
      "dueDays": 100,
      "dueMaximumAmount": 12.95,
      "dueDayOfMonth": 10,
      "discountDays": 100,
      "percent": 4,
      "discountDayOfMonth": 10,
      "discountMonthForward": 100,
      "fixedDate": "2001-12-31",
      "enabledFlag": "N",
      "startActiveDate": "2001-12-31",
      "endActiveDate": "2001-12-31",
      "dueMonthForward": 100,
      "cutoffDay": 12
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-139 CreateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	Yes	String (10)	Status code for a successful service call

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "elementName": "terms",
      "errorMessage": "Invalid terms value"
    }
  ]
}
```

```

    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}

```

Update Payment Term

Functional Area

Financials - Terms

Business Overview

This service can be used by an external financial system to send updated payment terms information to Merchandising. Terms and external reference ID are optional fields, but at least one must be provided when updating a payment term. If both are passed, Terms will be used to identify the record. If only the external reference ID is provided, it will be used to find a matching payment term. If more than one record is found, the service will raise an error.

For more details on Payment Terms, see the Create Payment Term service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/financials/terms/paymentTerm/update

Input Payload Details

Table 5-140 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
terms	No	String (15)	Unique id for this payment term.
externalReferenceId	No	String (50)	ID for the supplier terms used in the external financial system.
termsCode	Yes	String (50)	The alpha value which acts as the Term code in Oracle Financials.
description	Yes	String (240)	Description of the payment terms.
rank	No	Number (10)	Unique rank to rate invoice payment terms against PO terms

Sample Input Message

```

{
  "terms": "55",
  "externalReferenceId": "55",
  "termsCode": "55HDRUPD",
  "description": "55 TERMS Description HDR UPD",
  "rank": 13
}

```

Response Code: 200 (Success)

Table 5-141 UpdateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	Yes	String (10)	Status code for a successful service call

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Payment Term Detail

Functional Area

Financials - Terms

Business Overview

This service can be used by an external financial systems to send detail level updates to the payment terms that already exist within Merchandising. Terms and external reference ID are optional fields, but at least one must be provided when updating details of an existing payment term. If both are passed, Terms will be used to identify the record. If only the external reference ID is provided, it will be used to find a matching payment term. If more than one record is found, service will raise an error.

Service Type

PUT

ReST URL

MerchIntegrations/services/financials/terms/paymentTerm/detail/update

Input Payload Details

Table 5-142 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
terms	No	String (15)	Unique id for this payment term.
externalReferenceId	No	String (50)	ID for the supplier terms used in the external financial system.
termsCode	Yes	String (50)	The alpha value which acts as the Term code in Oracle Financials.
description	Yes	String (240)	Description of the payment terms.
rank	No	Number (10)	Unique rank to rate invoice payment terms against PO terms
details	No	Collection of Object	Child node.

Table 5-143 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
termsSequence	Yes	Number (10)	Sequence for the discount percent and part of unique identifier for the detail line.
dueDays	Yes	Number (3)	The number of days until payment is due.
dueMaximumAmount	Yes	Number (12,4)	Maximum payment amount due by a certain date.
dueDayOfMonth	Yes	Number (2)	Day of month used to calculate due date.
discountDays	Yes	Number (3)	Number of days in which payment must be made to get the discount.
percent	Yes	Number (12,4)	Percentage used to calculate discount available.
discountDayOfMonth	Yes	Number (2)	Day of month used to calculate discount date.
discountMonthForward	Yes	Number (3)	Number of months ahead to calculate discount date.
fixedDate	No	date	Fixed due date.
enabledFlag	Yes	String (1)	Indicates whether the Payment terms are valid or invalid within the respective application. The values would be either (Y)es or (N)o.
startActiveDate	No	date	Start date terms are in effect.
endActiveDate	No	date	End date terms are in effect.
dueMonthForward	Yes	Number (3)	Number of months ahead to calculate due date.
cutoffDay	Yes	Number (2)	Day of month after which the following month will be used for calculations.

Sample Input Message

```
{
  "terms": "55",
  "externalReferenceId": "55",
  "termsCode": "55DTLUPD",
  "description": "55DESCDTLUPD",
  "rank": 14,
  "details": [
    {
      "termsSequence": 2,
      "dueDays": 55,
      "dueMaximumAmount": 12.95,
      "dueDayOfMonth": 10,
      "discountDays": 55,
      "percent": 4,
      "discountDayOfMonth": 10,
      "discountMonthForward": 100,
      "fixedDate": "2001-12-31",
      "enabledFlag": "N",
      "startActiveDate": "2001-12-31",
      "endActiveDate": "2001-12-31",
      "dueMonthForward": 55,
      "cutoffDay": 12
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-144 UpdateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	Yes	String (10)	Status code for a successful service call

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "element": "terms",
      "message": "terms must be 55"
    }
  ]
}
```

```

    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}

```

Transaction Data

The following services are included in this functional area:

- [Transaction Data Upload Service](#)

Transaction Data Upload Service

Manage Transaction Data

Functional Area

Financials

Business Overview

The primary role of this service is to create transaction data records in the stock ledger, such as to adjust previously posted transactions. If the transaction is marked for WAC recalculation, it will recalculate the WAC for the item/location at the same time. The transactions written by this service will not impact unit inventory in Merchandising.

Service Type

Post

ReST URL

/Financials/uploadTranData

Input Parameters

Parameter Name	Required	Data Type	Description
Item	No	String	Item Identification
Dept	Yes	Number	Department Id
Class	Yes	Number	Class Id
Subclass	Yes	Number	Subclass Id
LocType	Yes	String	Location Type, Valid values are S-Store, W-Warehouse and E-External Finisher.
Location	Yes	Number	Location ID
TranDate	No	Date	Date of Transaction. (Format: DD-MON-YYYY). If null, vdate will be populated.
TranCode	Yes	Number	Transaction Type Identification.

Parameter Name	Required	Data Type	Description
AdjCode	No	String	Type of Adjustment. Valid values are A-Actual Landed Cost Adjustment, C-Cost Adjustment, U-Unit Adjustment.
Units	Yes	BigDecimal	Number of units involved in transaction.
TotalCost	No	BigDecimal	Total Cost associated with transaction.
TotalRetail	No	BigDecimal	Total Retail value associated with transaction.
RefNo1	No	String	Reference number associated with transaction.
RefNo2	No	String	Reference number associated with transaction.
GLRefNo	No	String	Reference number associated with transactions, and is used for defining the General Ledger account relationship, along with dept, class, subclass, location and tran_code.
OldUnitRetail	No	BigDecimal	Old Unit Retail
NewUnitRetail	No	BigDecimal	New Unit Retail
PgmName	No	String	Identifies the Oracle Retail module which inserted the record into the transaction data table.
SalesType	No	String	Type of Sale.Valid Values are 'C'learance,'P'romotion and 'R'egular.
VatRate	No	BigDecimal	VAT Rate
Avg Cost	No	BigDecimal	Average cost for the SKU
RefPackNo	No	String	Pack Number for the transaction item.
TotalCostExclusiveELC	No	BigDecimal	Cost Exclusive for ELC
WACRecalcIndicator	No	String	WAC should be recalculated.Valid values are Y-Yes or N-No.

Example JSON input

```
[
  {
    "item": "string",
    "dept": 0,
    "classId": 0,
    "subClass": 0,
    "locType": "string",
    "location": 0,
    "tranDate": "string",
    "tranCode": 0,
    "adjCode": "string",
    "units": 0,
    "totalCost": 0,
    "totalRetail": 0,
    "refNo1": "string",
    "refNo2": "string",
```

```
        "glRefNo": "string",
        "oldUnitRetail": 0,
        "newUnitRetail": 0,
        "pgmName": "string",
        "salesType": "string",
        "vatRate": 0,
        "avCost": 0,
        "refPackNo": "string",
        "totalCostExclElc": 0,
        "wacRecalcInd": "string"
        "hyperMediaContent": {}
    }
]
```

Output

Table 5-145 RestTranDataStatusRec

Parameter Name	Data Type
Status Message	String
TranDataErrorTBL	List< RestTranDataFailTbl>

Table 5-146 TranDataErrorTBL

Parameter Name	Data Type
Item	String
Dept	Number
Class	Number
Subclass	Number
Location Type	String
Location	Number
Tran Code	Number
Units	BigDecimal
Total Cost	BigDecimal
Total Retail	BigDecimal
RefNo1	String
RefNo2	String
GLRefNo	String
WACRecalcIndicator	String
ErrorMessage	String

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
TRAN_DATA	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
DEPS	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
SUBCLASS	Yes	No	No	No
STORE	Yes	No	No	No
WH	Yes	No	No	No
PARTNER	Yes	No	No	No
TRAN_DATA_CODES	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	Yes	No
VAT_RATE	Yes	No	No	No
VAT_REGION	Yes	No	No	No
PACKITEM	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
PERIOD	Yes	No	No	No

Foundation

The following services are included in this functional area:

- [Banners and Channels Download Service](#)
 - [Get Banners](#)
 - [Get Channels](#)
- [Location List Upload Service](#)
 - [Manage Location Lists](#)
- [Location Trait Upload Service](#)
 - [Create Location Trait](#)
 - [Delete Location Trait](#)
 - [Update Location Trait](#)
- [Suppliers and Partners](#)
 - [Partner Download Service](#)
 - [Supplier and Partner Upload Service](#)
 - [Supplier Download Service](#)
 - [Supplier Upload Service](#)
- [Tax Rules](#)
 - [Tax Rule Download Service](#)
 - * [Get Item Location Tax Rules](#)
 - * [Get Store Tax Location](#)
 - * [Get Tax Rule Groups](#)
 - [Tax Rule Upload Service](#)

Banners and Channels Download Service

The following services are included in this functional area:

- [Get Banners](#)
- [Get Channels](#)

Get Banners

Endpoints

/MerchIntegrations/services/foundation/banners

Functional Area

Foundation

Business Overview

This service retrieves all Banners. This service takes an optional query parameter of bannerId to retrieve detail for specified banner id

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/banners

Input Parameters

Parameter Name	Required	Data Type	Description
bannerId	No	String (4)	Banner ID

Output “Banner

Table 5-147 Banner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
bannerId	Yes	Number	This field contains the banner id.
bannerName	Yes	String (120)	This field contains the banner name.

Sample Response Message

```
[  
{  
  "bannerId": 1,  
  "bannerName": "Brick and mortar"
```

```
}  
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
BANNER	Yes	No	No	No

Get Channels

Endpoint

/MerchIntegrations/services/foundation/channels

Functional Area

Foundation

Business Overview

This service retrieves all Channel records. This service takes an optional query parameter of channel id to retrieve detail for specified channel

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/channels

Input Parameters

Parameter Name	Required	Data Type	Description
channelId	No	String (4)	Channel Id

Output “ Channel

Table 5-148 Channel - Object. See list of elements for detail

Element Name	Required	Data Type	Description
channelId	Yes	Number	This field contains the channel id.
channelName	Yes	String (120)	This field contains the channel name.

Table 5-148 (Cont.) Channel - Object. See list of elements for detail

Element Name	Required	Data Type	Description
channelType	Yes	String (6)	This field contains the channel type. Channel type is a user configured value defined in Code Detail with code_type as CHTY. Default values (which can be changed) are BANDM - Brick and Mortar, WEBSTR - Webstore, CAT - Catalog.
bannerId	Yes	Number	This field contains the banner id associated with this Channel.

Sample Response Message

```
[
{
"channelId": 1,
"channelName": "Brick and mortar",
"channelType": "BANDM",
"bannerId": 1
}
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CHANNELS	Yes	No	No	No

Location List Upload Service

The following services are included in this functional area:

- [Manage Location Lists](#)

Manage Location Lists

Functional Area

Foundation

Business Overview

This service allows for the creation and management of externally generated static location lists, as well as managing translated location list descriptions. Externally generated location lists are a list of explicitly provided stores and/or virtual warehouses. Static location lists cannot be rebuilt; no criteria is stored against externally generated static location lists. These lists can

only be maintained through the web service; no updates are allowed through the Merchandising user interface. Externally generated location lists are always public. Restricting the modification of an externally generated list should be handled in the source system.

This service supports three actions: Create, Update and Delete. The Create action allows the creation of a new location list, along with a list of one or more stores and/or stock-holding virtual warehouses that are not finishers to be grouped within. The Update action supports edits to location list definitions, adding or deleting of stores or virtual warehouses grouped in a list and adding or deleting translations of location list descriptions. The Delete action supports the ability to delete an entire location list. Physical warehouses can be passed to the service to be added to or deleted from a given list, the result will be the addition or removal of all virtual warehouses within the provided physical warehouse to/from the location list.

If Data Filtering is enabled in Merchandising through system options, you can control which users may use a given location list within Merchandising solutions by providing a filter organizational identifier with the list. This identifier can be for a chain, area, region or district depending on the Data Filtering Location List Organization Hierarchy level selected in system options. For example, if the system option level is area, then an area can be associated with a location list. Subsequently, only users with data filtering access to that area are able to use that location list on transactions throughout Merchandising. If no filter organizational identifier is associated with a location list, then any user can use the list within Merchandising.

Upon successful completion the service returns success along with the location list identifier and, if provided, the external reference. If there are upload validation errors the entire message is rejected and the service returns a failure status, along with a list of errors for each location list identifier, external reference, and location.

Service Type

PUT

ReST URL

MerchIntegrations/services/foundation/locationLists/manage

Input Payload Details

Table 5-149 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of Location List Data.

Table 5-150 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Specifies the intended action to create, update or delete an externally generated location list. Valid values are CREATE, UPDATE, or DELETE.

Table 5-150 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationList	No	Number (10)	Contains a location list identifier. When creating a location list, if a value is not provided, the system will create a unique sequence-generated identifier. For action as UPDATE, DELETE and if this value is provided, then it will be used to identify the location list.
locationListDescription	No	String (120)	Contains the location list description. This is mandatory field with action as CREATE.
externalReferenceNo	No	Number (20)	This field will optionally contain an external reference number used by an external system to identify the location list. For action as UPDATE and DELETE, if the location list id is not provided, then this field is mandatory.
filterOrganizationHierarchy	No	Number (10)	When Data Filtering is enabled in the system, the Filter Organizational Hierarchy may contain the Organizational Hierarchy value to which the Location List is assigned. This field will be used to control the Location Lists a user can use, based on whether the user has data filtering access to the Filter Organizational Hierarchy value
comments	No	String (2000)	Contains comments about the location list. For externally generated lists user may wish to provide information about the list origins or intended use.
locations	No	Collection of Object	References a collection of location list locations.
translations	No	Collection of Object	References a collection of translations of the location list description field.

Table 5-151 Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationType	Yes	String (1)	This field will carry the location type of the location being added to or deleted from the location list. Valid values for this field are Store (S) or Warehouse (W).

Table 5-151 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field will contain a location to be added to or deleted from the location list. The provided location may be a store, a stockholding virtual warehouses that is not finisher, or a physical warehouse. If a physical warehouse is provided, all stockholding/non-finisher virtual warehouses within that physical warehouse will be added or deleted.
deleteInd	No	String (1)	This optional field indicates if an existing location (or all virtual warehouses matching a physical warehouse location) should be deleted from a location list. This field will not have a value as Y with action as CREATE.

Table 5-152 Translations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field contains the language ID.
locationListDescription	No	String (120)	This field contains the translated description of the location list in the provided language. This field is mandatory if the translation tag is present. This field is not needed for delete of an existing translation.
deleteInd	No	String (1)	This optional field indicates if an existing translation should be deleted from a location list.

Table 5-153 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationList	Yes	Number (10)	Contains a location list identifier.
externalReferenceNo	Yes	Number (20)	This field will optionally contain an external reference number used by an external system to identify the location list.
errors	Yes	Array of String	

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
```

```

    "action": "CREATE",
    "locationList": 585313,
    "locationListDescription": "location list description",
    "externalReferenceNo": 157810,
    "filterOrganizationHierarchy": 126882,
    "comments": "comments",
    "locations": [
      {
        "locationType": "S",
        "location": 190104,
        "deleteInd": "N"
      }
    ],
    "translations": [
      {
        "language": 2,
        "locationListDescription": "Standortliste Beschreibung",
        "deleteInd": "N"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Table 5-154 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	

Table 5-155 ManageResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationList	Yes	Number (10)	Contains a location list identifier.
externalReferenceNo	Yes	Number (20)	This field will optionally contain an external reference number used by an external system to identify the location list.

Sample Response Message

```

{
  "items": [
    {
      "locationList": 585313,
      "externalReferenceNo": 157810
    }
  ]
}

```


Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service. The element `businessError` will be present if the payload passes schema validation but an exception is caught while processing the business logic.

Table 5-156 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationList	Yes	Number (10)	Contains a location list identifier.
externalReferenceNo	Yes	Number (20)	This field will optionally contain an external reference number used by an external system to identify the location list.
errors	Yes	Array of String	

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "locationList": 585313,
      "externalReferenceNo": 433378,
      "errors": [
        "String"
      ]
    }
  ]
}
```

Location Trait Upload Service

The following services are included in this functional area:

- [Create Location Trait](#)
- [Delete Location Trait](#)
- [Update Location Trait](#)

Create Location Trait

Functional Area

Foundation

Business Overview

The service is used to create location traits within Merchandising in order to keep it in sync with an external system that is used to maintain location traits. While a location trait is being created, this service will first validate that all required fields are present in the payload. Next, business level validations will be performed on the input information. Once all the validations are met, the location trait gets created in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/foundation/locationTrait/create

Input Payload Details

Table 5-157 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The unique id number of the location trait.
description	Yes	String (120)	The description of the location trait.

Sample Input Message

```
{
  "traitId": 1000,
  "description": "traitDesc"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Location Trait

Functional Area

Foundation

Business Overview

The service is used to delete an existing location trait from Merchandising in order to keep it in sync with an external system that is used to maintain location traits. While deleting a location trait, this service will first validate that all required fields are present in the payload. If all the validations are met, the location trait data is deleted from Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/foundation/locationTrait/delete

Input Payload Details

Table 5-158 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The id number of the location trait being deleted.

Sample Input Message

```
{
  "traitId": 1000
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Location Trait

Functional Area

Foundation

Business Overview

This service is used to update existing location traits within Merchandising in order to keep it in sync with an external system that is used to maintain location traits. While updating a location trait, this service will first validate that all required fields are present in the payload. Next, business level validations will be performed on the input information. Once all the validations are met, the location trait data gets updated in Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/foundation/locationTrait/update

Input Payload Details

Table 5-159 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The unique id number of the location trait.
description	Yes	String (120)	The description of the location trait.

Sample Input Message

```
{
  "traitId": 1000,
  "description": "traitDesc"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Suppliers and Partners

The following services are included in this functional area:

- [Partner Download Service](#)
- [Supplier and Partner Upload Service](#)

- [Supplier Download Service](#)
- [Supplier Upload Service](#)

Partner Download Service

Partner Publish Services

Endpoints

MerchIntegrations/services/foundation/partner

MerchIntegrations/services/foundation/partner/{partnerId}

Functional Area

Foundation - Partners

Business Overview

This service publishes partner master data to downstream consuming systems. There is no explicit filter in V_MERCHAPI_EXT_PARTNER_JSON, so functionally all rows from PARTNER are eligible for publication.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: PARTNER, PARTNER_ADDR

Additional business tables: ADDR, ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, COUNTRY, PARTNER_CFA_EXT, ADDR_CFA_EXT

JSON cache table: MERCHAPI_EXT_PARTNER

JSON generation view: V_MERCHAPI_EXT_PARTNER_JSON

Builds the partner header from PARTNER, derives the primary address from ADDR joined to ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, and COUNTRY, aggregates all partner addresses from ADDR, and adds partner- and address-level custom flex attributes from MERCHAPI_EXT_PARTNER_CFA and MERCHAPI_EXT_PARTNER_ADDR_CFA.

There is no explicit filter in V_MERCHAPI_EXT_PARTNER_JSON, so functionally all rows from PARTNER are eligible for publication.

The primary-address portion of the payload uses ADDR.MODULE = 'PTNR', ADD_TYPE_MODULE.PRIMARY_IND = 'Y', and ADDR.PRIMARY_ADDR_IND = 'Y'; the address array includes all partner addresses for ADDR.MODULE = 'PTNR'.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_PARTNER_JSON and apply no additional filter criteria; all PARTNER rows are eligible for publication.

The REBUILD branch updates changed and new rows and marks missing cache rows with DELETE_DATETIME.

ICL (Integration Change Log) table: ICL_RMS_PARTNER and ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on PARTNER, and address-related republishes are captured through ICL_RMS_ADDR for PARTNER_ADDR and PARTNER_ADDR_CFA_EXT changes.

ICL_RMS_ADDR rows are filtered to MODULE = 'PTNR' and ICL_TABLE_NAME IN ('PARTNER_ADDR','PARTNER_ADDR_CFA_EXT'), then republish the parent partner payload.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_PARTNER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_PARTNER_ADHOC_PROCESS ->
MERCHAPI_EXT_PARTNER_ADHOC_JOB

Webhook configuration api name: foundation/partner

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/partner

MerchIntegrations/services/foundation/partner/{partnerId}

Input Parameters for MerchIntegrations/services/foundation/partner

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (17)	Offset key ([partner type],[partner id] Example: AG,17)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-160 Input Parameter for "MerchIntegrations/services/foundation/partner/{partnerId}"

Parameter Name	Required	Data Type	Description
partnerId	Yes	String	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/partner/{partnerId}

Parameter Name	Required	Data Type	Description
partnerId	Yes	String	Partner ID
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Partner

Table 5-161 PageResultsItemsMerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-162 MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for partner details.
partnerType	No	String (6)	Specifies the type of partner. Valid partner types are held on the code tables under code type PTNR.
partnerId	No	String (10)	This field contains the unique number that identifies the partner within the system.
partnerName	No	String (240)	This field contains the partner's description or name.
partnerNameSecondary	No	String (240)	This field contains the secondary name of the partner.
currencyCode	No	String (3)	This field contains a code identifying the currency the partner uses for business transactions.
language	No	Number(6,0)	This field contains the partner's preferred language.
status	No	String (1)	This field indicates if the partner is currently active.
contactName	No	String (120)	This field contains the name of the partner's representative contact.
contactPhone	No	String (20)	This field contains the phone number of the partner's representative contact.
contactFax	No	String (20)	This field contains the fax number of the partner's representative contact.
contactTelex	No	String (20)	This field contains the telex number of the partner's representative contact.
contactEmail	No	String (100)	This field contains the email address of the partner's representative contact.

Table 5-162 (Cont.) MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manufacturerTaxId	No	String (18)	This field contains the manufacturer's tax identification number. This field is used for partner type Manufacturer.
principleCountryId	No	String (3)	This field contains the country ID to which the partner is assigned.
lineOfCredit	No	Number(20,4)	This field contains the line of credit the company has at the bank in the partner's currency. This field is not null when the partner type is Bank (BK).
outstandingCredit	No	Number(20,4)	This field contains the total amount of credit that the company has used or has charged against in the partner's currency. This field is not null when the partner type is Bank (BK).
openCredit	No	Number(20,4)	This field contains the total amount that the company can still charge against in the partner's currency. This field is not null when the partner type is Bank (BK).
yearToDateCredit	No	Number(20,4)	This field contains the total amount of credit the company has used year to date in the partner's currency. This field is not null when the partner type is Bank (BK).
yearToDateDrawdowns	No	Number(20,4)	This field contains the year-to-date payments the bank has made on behalf of the company in the partner's currency. This field is not null when the partner type is Bank (BK).

Table 5-162 (Cont.) MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
taxId	No	String (20)	This field contains the unique tax identification number of the partner. This will be used for reporting during the Customs Entry process.
terms	No	String (15)	This field contains the payment terms for the partner. These terms specify when payment is due and if any discounts exist for early payment. If populated, they will default on any invoice entered for this partner.
servicePerfReqInd	No	String (1)	This field indicates if the expense vendor's services (for example, snowplowing and window washing) must be confirmed as performed before paying an invoice from that expense vendor. Valid values are Y and N.
invoicePayLocation	No	String (6)	This field indicates where invoices from this expense vendor are paid, either at the store or centrally through corporate accounting. Valid values are S and C. This field will only be populated for expense vendors, and should only be S if using ReSA to accept payment at the store.

Table 5-162 (Cont.) MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
invoicecReceiveLocation	No	String (6)	This field indicates where invoices from this expense vendor are received, either at the store or centrally through corporate accounting. Valid values are S and C. This field should only be populated when using invoice matching.
importCountryId	No	String (3)	This field contains the import country of the Import Authority. This field is not populated for other partner types.
primaryImportAuthInd	No	String (1)	This field indicates if an Import Authority is the primary Import Authority for an import country. This field will always be N for other partner types. There must be one and only one primary Import Authority for each country associated with an Import Authority on the partner table.
comments	No	String (2000)	This field contains any comments associated with the partner.
transferEntityId	No	Number(10,0)	This field contains the transfer entity with which an external finisher is associated.

Table 5-162 (Cont.) MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatRegion	No	Number(4,0)	This field contains the tax region with which a partner is associated. Valid values will be found on the VAT_REGION table. If SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is SVAT or GTS, then each partner will be required to have an associated VAT_REGION. For other DEFAULT_TAX_TYPE values, VAT functionality is not used and PARTNER.VAT_REGION will not be required.
orgUnitId	No	Number(15,0)	This field contains the org unit ID with which the partner is associated.
autoRcvStockInd	No	String (1)	This field indicates whether the system will update the stock for the external finisher when the first leg of the transfer is shipped. Valid values are Y and N.
primaryAddressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressAdd1	No	String (240)	This field contains the first line of the address.
primaryAddressAdd2	No	String (240)	This field contains the second line of the address.
primaryAddressAdd3	No	String (240)	This field contains the third line of the address.

Table 5-162 (Cont.) MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressCity	No	String (120)	This field contains the name of the city that is associated with the address.
primaryAddressState	No	String (3)	This field contains the abbreviation of the state for the address.
primaryAddressStateName	No	String (120)	This field contains the full name of the state for the address.
primaryAddressCountryId	No	String (3)	This field contains the country where the address exists.
primaryAddressCountryName	No	String (120)	This field contains the country name for the address.
primaryAddressPost	No	String (30)	This field contains the zip code for the address.
primaryAddressContactName	No	String (120)	This field contains the name of the contact for the partner at this address.
primaryAddressContactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
primaryAddressContactTelex	No	String (20)	This field contains the telex number of the partner's representative contact.
primaryAddressContactFax	No	String (20)	This field contains the fax number of the contact person at this address.
primaryAddressContactEmail	No	String (100)	This field contains the email address of the partner's representative contact.
primaryAddressCounty	No	String (250)	This field contains the county name for the address.
primaryAddressJurisdictionCode	No	String (10)	This field identifies the jurisdiction code for the country-state relationship.
createDateTime	No	dateTime	This field holds the date and time when the record was inserted (UTC).

Table 5-162 (Cont.) MerchApiPartner - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This field holds the date and time when the record was updated (UTC).
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the partner.
address	No	Collection of Object	This list holds address details for the partner.
cacheTimestamp	No	dateTime	This field specifies the date and time when the partner record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-163 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-164 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressKey	No	Number(11,0)	This field contains the unique address key.
addressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
addressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressInd	No	String (1)	This field indicates whether the address is the primary address for the address type.
add1	No	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	No	String (120)	This field contains the name of the city that is associated with the address.
state	No	String (3)	This field contains the state abbreviation that is associated with the address.
stateName	No	String (120)	This field contains the state description that is associated with the address.
countryId	No	String (3)	This field contains the country where the address exists.
countryName	No	String (120)	This field contains the country description that is associated with the address.
post	No	String (30)	This field contains the zip code for the address.
contactName	No	String (120)	This field contains the name of the contact person at this address.

Table 5-164 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
contactTelex	No	String (20)	This field contains the telex number of the contact person at this address.
contactFax	No	String (20)	This field contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This field contains the email address of the contact person at this address.
phone2Type	No	String (6)	This field contains the type of the phone number stored in the phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This field contains an additional phone number associated with the address.
phone3Type	No	String (6)	This field contains the type of the phone number stored in the phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This field contains a second additional phone number associated with the address.

Table 5-164 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
latitude	No	String (30)	This field contains the north-south geographic coordinate of the address, represented in decimal degrees.
longitude	No	String (30)	This field contains the east-west geographic coordinate of the address, represented in decimal degrees.
county	No	String (250)	This field contains the county where the address exists.
jurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse's address.

Table 5-165 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
```

```

"partnerType": "AG",
"partnerId": null,
"partnerName": null,
"partnerNameSecondary": null,
"currencyCode": "USD",
"language": 1,
"status": null,
"contactName": "contactName",
"contactPhone": "1800800800",
"contactFax": "617-897-0902",
"contactTelex": null,
"contactEmail": "someone@email.com",
"manufacturerTaxId": null,
"principleCountryId": null,
"lineOfCredit": 1.0,
"outstandingCredit": 1.0,
"openCredit": 1.0,
"yearToDateCredit": 1.0,
"yearToDateDrawdowns": 1.0,
"taxId": null,
"terms": "108",
"servicePerfReqInd": null,
"invoicePayLocation": "S",
"invoicecReceiveLocation": null,
"importCountryId": null,
"primaryImportAuthInd": null,
"comments": null,
"transferEntityId": 1,
"vatRegion": 1000,
"orgUnitId": 1111111111,
"autoRcvStockInd": null,
"primaryAddressType": "01",
"primaryAddressTypeDescription": null,
"primaryAddressAdd1": "101 First Street",
"primaryAddressAdd2": "201 Second Street",
"primaryAddressAdd3": null,
"primaryAddressCity": "Minneapolis",
"primaryAddressState": "MN",
"primaryAddressStateName": "stateName",
"primaryAddressCountryId": "US",
"primaryAddressCountryName": "countryName",
"primaryAddressPost": "55555-1234",
"primaryAddressContactName": "contactName",
"primaryAddressContactPhone": "1800800800",
"primaryAddressContactTelex": null,
"primaryAddressContactFax": "617-897-0902",
"primaryAddressContactEmail": "someone@email.com",
"primaryAddressCounty": "Some County",
"primaryAddressJurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",

```

```

    "createDateTime": "2001-12-31T23:59:59.000Z",
    "updateDateTime": "2001-12-31T23:59:59.000Z"
  },
  ],
  "address": [
    {
      "addressKey": 114772,
      "addressType": "01",
      "addressTypeDescription": null,
      "primaryAddressInd": "Y",
      "add1": "101 First Street",
      "add2": "201 Second Street",
      "add3": null,
      "city": "Minneapolis",
      "state": "MN",
      "stateName": "MN",
      "countryId": "US",
      "countryName": "US",
      "post": "55555-1234",
      "contactName": "Laura Johnson",
      "contactPhone": "1800800800",
      "contactTelex": null,
      "contactFax": "617-897-0902",
      "contactEmail": "someone@email.com",
      "phone2Type": "W0",
      "phone2": "5551234567",
      "phone3Type": "W0",
      "phone3": "5551234567",
      "latitude": "44.38129",
      "longitude": "-92.03521",
      "county": null,
      "jurisdictionCode": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ]
    }
  ],
  ],
  "cacheTimestamp": "2001-12-31T23:59:59.000Z",
  "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
  {
    "href": "https://<server>:<port>/Application/services/resources?
    limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs

```

```

etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Address

Name	Data Type	Description
addressKey	Number	This field contains the unique address key.
addressType	String	This field specifies type for the address. Valid values are: 01 -Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
addressTypeDescription	String	This field specifies address type description.
primaryAddressInd	String	This field indicates whether the address is the primary address for the address type.
add1	String	This field contains the first line of the address.
add2	String	This field contains the second line of the address.
add3	String	This field contains the third line of the address.
city	String	This field contains name of the city that is associated with the address.
state	String	This field contains name of the state that is associated with the address.
stateName	String	This field contains name of the state description that is associated with the address.
countryID	String	This field contains name of the country that is associated with the address.
countryName	String	This field contains name of the country description that is associated with the address.
post	String	This field contains the zip code for the address.
contactName	String	This field contains the name of the contact person at this address.
contactPhone	String	This field contains the phone number of the contact person at this address.
contactTelex	String	This field contains the telex number of the warehouse's representative contact.
contactFax	String	This field contains the fax number of the contact person at this address.
contactEmail	String	This field contains email address of the warehouse's representative contact.
county	String	This field contains the county where the address exists.
jurisdictionCode	String	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttribute		This list holds custom flex attributes for the address.

Output - Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```
{
  "items": [
    {
      "action": "INSERT",
      "partnerType": "E",
      "partnerId": "B",
      "partnerName": "Packaging External Finisher updated",
      "partnerNameSecondary": "Packaging External Finisher",
      "currencyCode": "USD",
      "language": null,
      "status": "A",
      "contactName": "Elinor",
      "contactPhone": "1545223556",
      "contactFax": null,
      "contactTelex": null,
      "contactEmail": null,
      "manufacturerTaxId": null,
    }
  ]
}
```

```

"principleCountryId": "US",
"lineOfCredit": null,
"outstandingCredit": null,
"openCredit": null,
"yearToDateCredit": null,
"yearToDateDrawdowns": null,
"taxId": null,
"terms": "1",
"servicePerfReqInd": "N",
"invoicePayLocation": null,
"invoicecReceiveLocation": null,
"importCountryId": null,
"primaryImportAuthInd": "N",
"comments": null,
"transferEntityInd": 2000,
"vatRegion": null,
"orgUnitId": 111111111,
"autoRcvStockInd": "N",
"primaryAddrAdd1": "Add1",
"primaryAddrAdd2": null,
"primaryAddrAdd3": null,
"primaryAddrCity": "City",
"primaryAddrState": null,
"primaryAddrStateName": null,
"primaryAddrCountryId": "US",
"primaryAddrCountryName": "United States of America",
"primaryAddrPost": null,
"primaryAddrContactName": null,
"primaryAddrContactPhone": null,
"primaryAddrContactTelex": null,
"primaryAddrContactFax": null,
"primaryAddrContactEmail": null,
"primaryAddrOracleVendorSiteId": null,
"primaryAddrCounty": null,
"primaryAddrJurisdictionCode": null,
"createDateTime": "2021-05-07T09:23:59",
"updateDateTime": "2021-05-07T09:23:59",
"customFlexAttribute": [{
  "name": "LANGUAGE",
  "value": "German",
  "valueDate": null,
  "createDateTime": "2021-02-17T00:00:00",
  "updateDateTime": "2021-02-17T00:00:00"
}, {
  "name": "COMPANY_SIZE",
  "value": "525",
  "valueDate": null,
  "createDateTime": "2021-02-17T00:00:00",
  "updateDateTime": "2021-02-17T00:00:00"
}],
"address": [{
  "addressKey": 20069,
  "addressType": "05",
  "primaryAddressInd": "Y",
  "add1": "Add1",
  "add2": null,

```

```

        "add3": null,
        "city": "City",
        "state": null,
        "stateName": null,
        "countryId": "US",
        "countryName": "United States of America",
        "post": null,
        "contactName": null,
        "contactPhone": null,
        "contactTelex": null,
        "contactFax": null,
        "contactEmail": null,
        "oracleVendorSiteId": null,
        "county": null,
        "jurisdictionCode": null,
        "createDateTime": "2021-02-16T00:00:00",
        "updateDateTime": "2018-09-20T00:00:00",
        "customFlexAttribute": null
    }, {
        "addressKey": 20068,
        "addressType": "01",
        "primaryAddressInd": "Y",
        "add1": "Add1",
        "add2": null,
        "add3": null,
        "city": "City",
        "state": null,
        "stateName": null,
        "countryId": "US",
        "countryName": "United States of America",
        "post": null,
        "contactName": null,
        "contactPhone": null,
        "contactTelex": null,
        "contactFax": null,
        "contactEmail": null,
        "oracleVendorSiteId": null,
        "county": null,
        "jurisdictionCode": null,
        "createDateTime": "2021-02-16T00:00:00",
        "updateDateTime": "2018-09-20T00:00:00",
        "customFlexAttribute": null
    }],
    "hasMore": false,
    "limit": 10000,
    "count": 3,
    "links": [
        {
            "href": "https://rex.retail.us-phoenix-1.ocs.oc-test.com/rgbu-rex-rgbu-dev1-mfcs19-1/MerchIntegrations/services/foundation/partner?limit=10000&since=1970-01-01T00:00:00.001Z&before=2022-03-25T10:34:36.748111+00:00",
            "rel": "self"
        }
    ]
}

```


Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_PARTNER	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes
MERCHAPI_PARTNER	Yes	Yes	Yes	Yes
PARTNER_CFA_EXT	Yes	No	No	No
ADDR_CFA_EXT	Yes	No	No	No
V_MERCHAPI_EXT_PARTNER_JSON	Yes	No	No	No

Supplier and Partner Upload Service

Manage Suppliers and Partners

Functional Area

Foundation - Suppliers and Partners

Business Overview

This service is used for the creation and update of suppliers, supplier sites, partners, org unit association, addresses and flex attributes from an external application.

When new suppliers or supplier sites are sent from an external system, they must contain the required header details, as well as address and org unit information. One or more org units can be associated with a supplier site using this service, if applicable. When creating a supplier parent, providing the supplier site information is optional. However, when creating or updating supplier sites, both supplier parent and supplier site information must be provided.

The following defaults are performed by the service:

Yes/No indicators are defaulted to No if not provided.

Settlement code - defaulted to N if not provided.

EDI sales report frequency - defaulted to NULL if the provided value is not D or W.

Supplier Quantity Level - defaulted to EA if not provided.

Inventory Management Level - Defaulted to D/S if not provided. Updating this value later will trigger deletion of all existing Supplier Inventory Management Records.

Delivery Policy - Defaulted to NEXT if not provided.

Invoice Gross Net - Defaulted to Net if not provided.

Mandatory addresses as defined in the system are defaulted when creating address records even if the request does not contain the mandatory address types.

Finance Integration Specific Enrichment Workflow

The dependence of this service on the FINANCIAL_AP system option has been functionally migrated to the External Supplier Management system option and a set of sub options

discussed below. When External Supplier Management is enabled, the optional attribute `vendorTypeCode` in the payload will be used to identify if the inbound information should be used to create or update a supplier or a partner record in Merchandising. External financial systems can utilize this attribute to create partners in Merchandising. The `vendorTypeCode` will be validated against data available in the Vendor type code mapping table in Merchandising that holds detail of whether a specific code maps to a supplier or partner and the type of partner intended. The vendor type code mapping must be setup via the Merchandising spreadsheet upload functionality prior to the integration of supplier/partner records. Additionally, if External Supplier Management is opted for the payload would be subject to enrichment based on the below options.

Filter inactive suppliers - Controls whether inactive suppliers should be excluded in the Create subscription.

Use Purchase Pay Site Flag - Controls if the Purchase Pay site Flags should be used to create and manage the 'Purchasing Address' and 'Remittance Address' respectively if a specific address type isn't specified in the payload.

Use Site Defaults - Controls if a subset of supplier details should be populated using information at the site level if not specified for the supplier.

Use Supplier Alternate Name - Controls if the secondary name on the payload should be used for Supplier Description Creation as opposed to the Supplier Name

As part of the Financial Integration specific workflow, supplier/partner integration into Merchandising will be subject to the below.

While creating a new supplier, there must be at least one supplier site that is either a payment site or purchasing site as part of the payload. If there is only one supplier site associated, it must be both a purchasing and payment site.

Freight Terms, Payment Terms, Language, Currency, Contact Name, and Contact Phone information will be defaulted from the supplier site (primary pay site/procurement site/payment site) to the supplier record.

Address Type is typically not expected from the external financial system and will be determined using the procurement site and payment site attributes if the Purchase Pay Site Flag is set. An address type of 04 will be created for a procurement site while an address type of 06 will be created for a payment site. If the supplier site has both procurement and payment site flags set, both address types 04 and 06 will be created.

Contact specific defaulting will be based on the below

If there are multiple contacts for a supplier site, the contact name with the lowest external reference key (earliest record) will be used as the default contact name for the supplier site.

If there are multiple contacts for a supplier site and the contact phone/contact fax for the supplier site is not provided, it will be defaulted to 999999999.

If no contact name is available for the supplier and supplier site, it will be set to 'UNKNOWN'.

If no contact phone is available for the supplier and supplier site, it will be set to 0.

If no contact fax/contact email is available for the supplier site, it will be set to its addresses' contact fax/contact email.

If default tax type is SVAT or GTS, vat region on the supplier/partner record will be defaulted from a store having the same state in the address as the payment site or procurement site address. If state is not provided, country will be used as the basis for defaulting. If a default value is not found, the supplier/supplier site will be created with a null vat region and set to Inactive status until the vat region is associated in Merchandising.

The Use Alternate Supplier Site Name system option will control if the supplier site name will be substituted with the secondary name in the payload and vice versa. This is intended to allow for a more descriptive supplier name integration into Merchandising if the site name carried an abridged description/key. Substitution will only be performed if supplier site secondary name is provided. This system option can be managed via the System Options screen.

For the nullable fields, if no value is passed or value passed is NULL in the payload, then the service will update the corresponding fields in the database to NULL. For non-nullable fields the existing value will be retained.

Note:

Creation of partners of type Escheat-State and Escheat-Country will not be supported through this integration.

If the vendorTypeCode is not provided in the payload, a supplier record will be created.

The payload provides reference id fields for certain attributes that will enable the external financial system to send its own IDs for these attributes as opposed to sending IDs generated by Merchandising. These are listed below.

languageReferenceId - defaulted to primary language if not provided.

termsReferenceId

freightTermsReferenceId

orgUnitReferenceId

Usage of the external reference IDs is subject to the assumption that these external references have been setup and maintained against the attribute in Merchandising. For example, the external financial system can send a terms reference id provided that the same information has been sent to Merchandising as part of the terms import process. This will allow for Merchandising to use the terms reference id provided to cross reference and identify the correct term in Merchandising to be associated with the supplier/partner record being created. If an external reference id is provided for any attribute say terms it will be cross-referenced against the attribute definition tables (terms tables in this case) for a match. If the cross-reference check returns multiple records one of these will be associated with the supplier/partner record being created. If no matching data is found an error will be raised.

Partner Integration

If the vendor type code specified maps to a partner, the sites in the inbound payload will correspond to the partner record created. The partner record will however hold a reference of the supplier record associated with the site via the parent reference ID and parent name fields stored in the partner table. The defaulting logic and validations for partners is similar to that supplier sites with the below exceptions.

For new external finisher type partners, the service defaults the transfer entity from the resolved org unit. If the org unit or matching transfer entity cannot be resolved, the partner is created in Inactive status until the transfer entity is populated in Merchandising. Existing partner updates do not override the stored transfer entity.

For Bank type partners, the fields Line of Credit, Outstanding Credit, Open Credit, Year to Date Credit and Year to Date Drawdowns will be defaulted to 0. Principle Country ID will be defaulted from the address as also done for the Import Authority partner type).

Any information present in the webservice request will update the existing attributes with the following exceptions- (contact fax, contact pager, ship method, payment method, contact telex, contact email, default lead time, duns number, duns loc, tax id, vat region). When modifying a

supplier parent, providing the supplier site information is optional. Updates will be accepted for both supplier and partner irrespective of status.

Service Type

POST

ReST URL

MerchIntegrations/services/foundation/suppliers/create

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Attributes - Object. See list of elements for detail

CustomFlexAttribute - Object. See list of elements for detail

LocalizationExtensions - Object. See list of elements for detail

LocalizationExtensions.Attributes - Object. See list of elements for detail

SupplierSites - Object. See list of elements for detail

OrgUnit - Object. See list of elements for detail

Address - Object. See list of elements for detail

Address.Address - Object. See list of elements for detail

Response Code: 200 (Success)

CreateResponse - Object. See list of elements for detail

CreateResponse.Items - Object. See list of elements for detail

CreateResponse.Items.SupplierSites - Object. See list of elements for detail

CreateResponse.Items.SupplierSites.Address - Object. See list of elements for detail

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Update Suppliers and Partners (deprecated)

Functional Area

Foundation - Suppliers and Partners

Business Overview

Deprecated in 25.0.201.0: Scheduled for Removal in 2 years. Use MerchIntegrations/services/foundation/suppliers/create.

This service modifies existing parent suppliers, supplier sites, partners, org unit association, addresses and flex attributes. Any information present in the webservice request will update the existing attributes with a few exceptions, mentioned below. When modifying a supplier parent, providing the supplier site information is optional.

The following attribute values are defaulted to the old value if not provided in the request:

- contact name
- contact phone

- contact fax
- contact pager
- ship method
- payment method
- contact telex
- contact email
- default lead time
- duns number
- duns loc
- tax id
- vat region

When modifying supplier sites, it is expected that both supplier parent and supplier site information will be provided. Certain supplier attribute values are defaulted to the old values if not provided in the update request. Refer to the list of attributes under Modifying a Supplier Parent. Attributes not in the above list will be replaced by user-provided values.

Existing address details are overwritten by the information contained in the address section of the service request. Addresses with address types 04 - Order and 06 - Remittance, however, are deleted from the system if these addresses are not included in the request.

When Financials integration is enabled (`financial_ap = 'A'`), the optional attribute `vendorTypeCode` is used to identify between suppliers and partners. External financial systems utilizing this attribute can also update partners in Merchandising.

Note

The vendor type code mapping must be set up through the Merchandising spreadsheet upload functionality.

Partner integrations of type Escheat-State and Escheat-Country are not be supported through this interface. If `vendorTypeCode` is blank, it is assumed that a supplier is being updated.

The external financial system can send its own ID for certain supplier attributes on optional reference fields in the payload. These external reference IDs do not need to be set up against the relevant data to enable their usage. Reference IDs that are available in the payload are listed below. If provided these will be used to cross-reference and identify the corresponding Merchandising ID and data. If the external ID provided maps to multiple records in the foundation data, then one of the records will be associated with the supplier/partner record being created. An error will be raised if no match is found.

- `languageReferenceId` - defaulted to primary language if not provided.
- `termsReferenceId`
- `freightTermsReferenceId`
- `orgUnitReferenceId`

Updates will be accepted for both supplier and partner irrespective of status.

Service Type

PUT

ReST URL

MerchIntegrations/services/foundation/suppliers/update

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-166 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of suppliers.

Table 5-167 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierReferenceNo	Yes	String (32)	This holds the ID for the supplier or parent partner used in the external financial system. It is populated by the integration that brings suppliers or partners from external financial systems into Merchandising. This ID and the supplier site ID or partner ID can be used to join to information about the supplier in the external system.
supplier	No	Number (10)	Unique identifying number for a supplier parent within the system. When a new supplier is added to the system, this value can either be provided by the user or it can be system generated. Leave this field as null for a system generated supplier ID.
vendorTypeCode	No	String (50)	This column contains the identifier for the vendor type from the external financial system.
attributes	Yes	Record	This contains supplier parent information.
supplierSites	No	Collection of Object	References a collection of supplier sites.

Table 5-168 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierName	Yes	String (240)	This field contains the supplier or partner name.

Table 5-168 (Cont.) Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierNameSecondary	No	String (240)	This type can hold secondary name for the supplier or partner with a max length of 240 characters.
contactName	No	String (120)	This field contains the name of the supplier or partner representative contact for this site.
contactPhone	No	String (20)	This field contains a telephone number for the supplier's or partner's representative contact.
contactFax	No	String (20)	This field contains a fax number for the supplier's or partner's representative contact.
contactPager	No	String (20)	This field contains a pager number for the supplier's representative contact.
supplierStatus	Yes	String (1)	This field contains the status of the supplier or partner. Valid values include: 'A' for an active supplier or 'I' for an inactive supplier. The status of a supplier will be checked when an order is being created to make certain the supplier is active.
qualityControlPercentage	No	Number (12,4)	This field contains the percentage of items per receipt that will be marked for quality checking. If the field is null and qc_ind is Y, existing value will be retained.
qualityControlFrequency	No	String (2)	This field contains the frequency for which items per receipt will be marked for quality checking. If the field is null and qc_ind is Y, existing value will be retained.
vendorCheckingPercentage	No	Number (12,4)	This field contains percentage of items per receipt that will be marked for vendor checking. If the field is null and vc_ind is Y, existing value will be retained.
vendorCheckingFrequency	No	Number (2)	This field contains the frequency for which items per receipt that will be marked for vendor checking. If the field is null and vc_ind is Y, existing value will be retained.
currencyCode	No	String (3)	This field contains code identifying the currency the supplier site or partner uses for business transactions.
language	No	Number (6)	This field contains the suppliers or partners preferred language. This field is provided for custom purchase orders in a specified language.

Table 5-168 (Cont.) Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageReferenceId	No	String (50)	This field is primarily used to hold the ID for the language used in the external financial system.
terms	No	String (15)	This field contains an indicator identifying the purchase terms that will default when an order is created for the supplier site or partner. These terms specify when payment is due and if any discounts exist for early payment.
termsReferenceId	No	String (50)	This column holds the ID for the supplier or partner terms used in the external financial system.
freightTerms	No	String (30)	This field contains code indicating what freight terms will default when an order is created for the supplier site.
freightTermsReferenceId	No	String (50)	This column holds the ID for the freight terms used in the external financial system.
minReturnValue	No	Number (20,4)	This field contains a value if the supplier site requires a minimum merchandise value to be returned to accept the return. Returns of less than this amount will not be processed by the system. This field is stored in the supplier's currency.
returnCourier	No	String (250)	This field contains the name of the courier that should be used for returns to the supplier site.
handlingPercentage	No	Number (12,4)	This field contains the default percent to be multiplied by the return's total cost to determine the handling cost for the return.
ediChannel	No	Number (4)	If the supplier is an EDI supplier and supports vendor initiated ordering, this field will contain the channel ID for the channel to which all inventory for these types of orders will flow. This field is used when a vendor initiated order is created for a physical warehouse to determine the virtual warehouse within the physical warehouse to which the inventory will flow. The virtual warehouse belonging to the indicated channel will be used. This will only be used in a multichannel environment. Valid values are Yes (Y) and No (N).

Table 5-168 (Cont.) Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
costChangeVariancePercentage	No	Number (12,4)	This field contains a percent that determines whether a cost change can be auto approve via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
costChangeVarianceAmt	No	Number (20,4)	This field contains an amount (in supplier currency) that determines whether a cost change can be auto approve via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
shipMethod	No	String (6)	This field contains the default method used to ship the items on the purchase order from the supplier site. Valid values are held in the Shipment Type code type, SHPM. Examples of shipment types are Vessel, Non-Container (10), Vessel, Container (11), Rail, Non-container (20) and Rail, Container (21).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
contactTelex	No	String (20)	This field contains the telex number of the partner or suppliers representative contact.
contactEmail	No	String (100)	This field contains the email address of the partner or suppliers representative contact.
vatRegion	No	Number (4)	This field contains the unique identifying number for the VAT region applicable for this site.
invoicePayLocation	No	String (6)	This field indicates where invoices from this supplier site or partner are paid - at the store (S) or centrally through corporate accounting (C).
invoiceReceiveLocation	No	String (6)	This field indicates where invoices from this supplier site or partner are received - at the store (S) or centrally through corporate accounting (C).
comments	No	String (2000)	This field contains any miscellaneous comments associated with the supplier or partner.

Table 5-168 (Cont.) Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
defaultItemLeadTime	No	Number (4)	This field holds the default lead time for the supplier site. The lead time is the time the supplier needs between receiving an order and having the order ready to ship. This value will be defaulted to item/supplier relationships.
dunsNumber	No	String (9)	The Dun and Bradstreet number of the supplier.
dunsLocation	No	String (4)	The Dun and Bradstreet number of the location of the supplier.
defaultPricingControl	No	String (1)	Indicates the default value of the pricing control attribute at the supplier site level when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: R - Retailer, S - Supplier.
generateConsignmentOrderInd	No	String (1)	This field will allow specification of whether Merchandising is required to generate POs against consignment sales. Valid values are: 'Y' - MFCS will create PO, 'N' - POs will get created externally. If supplier parent, this is defaulted to Y. If supplier site and field is null, this is defaulted to Y.
generateSelfInvoicingInd	No	String (1)	This field will allow specification of whether self-invoicing needs to be done for consignment purchases or if vendor will send invoices to be matched. Valid values are: 'Y' - MFCS will create invoice, 'N' - Invoice will get created by vendor. If supplier parent, this is defaulted to Y. If supplier site and field is null, this is defaulted to Y.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-168 (Cont.) Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dealUploadStatus	No	String (6)	Indicates the status in which the deal needs to be uploaded into the system by upload batch process. Valid values are in the code type DLUS(Deal Upload Status). Examples are Worksheet(W), Submitted(S) and Approved(A). If the supplier does not exist and there is no deal_upload_status input then this is defaulted to W (worksheet).
taxId	No	String (20)	This field contains the unique tax identification number of the supplier site or partner.
procurementSiteInd	No	String (1)	Indicates if the supplier site or partner is a procurement site used for purchasing.
paymentSiteInd	No	String (1)	Indicates if the supplier site or partner is a payment site used for invoice payment.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-169 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-170 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-171 LocalizationExtensions.Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-172 SupplierSites - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierSiteReferenceNo	Yes	String (32)	This holds the ID for the supplier or partner used in the external financial system. It is populated by the integration that brings suppliers or partners from external financial systems into Merchandising. This ID and the supplier site ID or partner ID can be used to join to information about the supplier in the external system.
supplierSite	No	Number (10)	Unique identifying number for a supplier site or partner within the system. When a new supplier site is added to the system, this can either be provided by the user or it can be system generated. Leave this field as null for a system generated supplier site ID.
attributes	Yes	Record	This contains supplier site attributes.
orgUnit	Yes	Collection of Object	References a collection of supplier org units.
address	Yes	Collection of Object	References a collection of supplier addresses.

Table 5-173 OrgUnit - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orgUnitId	No	Number (15)	This field contains org_unit_id.
orgUnitReferenceId	No	String (50)	This column holds the organizational unit ID used in the external financial system.
primaryPaySite	No	String (1)	This field contains the primary pay site indicator.

Table 5-174 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressReferenceKey	No	String (32)	This column contains the external reference ID used to distinguish between different addresses.
addressKey	No	Number (11)	This column contains a unique number used to distinguish between different addresses.
address	Yes	Record	This contains the supplier's address information.

Table 5-175 Address.Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressType	No	String (2)	This contains the address type. Valid values (e. g. 01 - Business, 02 - Postal, etc.) are on the add_type table. If any address types have been flagged as mandatory and are not included when the supplier or partner is being created, then if an ordering address has been included, the missing mandatory addresses will be defaulted to that address. If not, then the remittance address will be used. If neither an order nor remittance address is included, then the first address sent is used.
primaryAddressInd	No	String (1)	This column indicates whether the address is the primary address for the address type.
add1	Yes	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	Yes	String (120)	This column contains the name of the city that is associated with the address.
state	No	String (3)	This column contains the state abbreviation for the address.
country	Yes	String (3)	This column contains the country where the address exists.
post	No	String (30)	This column contains the zip code for the address.
contactName	No	String (120)	This column contains the name of the contact for the supplier or partner at this address.
contactPhone	No	String (20)	This column contains the phone number of the contact person at this address.
contactFax	No	String (20)	This column contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This column contains the email address of the partner or suppliers representative contact.
jurisdictionCode	No	String (10)	ID associated to the tax jurisdiction of the country-state relationship.

Table 5-175 (Cont.) Address.Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
contactTelex	No	String (20)	This field contains the telex number of the partner or suppliers representative contact.

Sample Input Message

```

{
  "items": [
    {
      "supplierReferenceNo": "1",
      "supplier": 3026,
      "vendorTypeCode": null,
      "attributes": {
        "supplierName": "Prime Hardware",
        "supplierNameSecondary": "Prime Hardware 2",
        "contactName": "Alexandre Roma",
        "contactPhone": "234-112-7654",
        "contactFax": "234-112-7654",
        "contactPager": null,
        "supplierStatus": "A",
        "qualityControlPercentage": 3,
        "qualityControlFrequency": "5",
        "vendorCheckingPercentage": 3,
        "vendorCheckingFrequency": 2,
        "currencyCode": "USD",
        "language": 1,
        "languageReferenceId": null,
        "terms": "108",
        "termsReferenceId": null,
        "freightTerms": "03",
        "freightTermsReferenceId": null,
        "minReturnValue": 3.55,
        "returnCourier": "125",
        "handlingPercentage": 3,
        "ediChannel": 2,
        "costChangeVariancePercentage": 9.65,
        "costChangeVarianceAmt": 9.65,
        "shipMethod": "30",
        "paymentMethod": "0A",
        "contactTelex": "234-112-7654",
        "contactEmail": "a_roma@primehardware.com",
        "vatRegion": 1000,
        "invoicePayLocation": "S",

```

```

"invoiceReceiveLocation": "C",
"comments": null,
"defaultItemLeadTime": 7,
"dunsNumber": "108890",
"dunsLocation": "1055",
"defaultPricingControl": "R",
"generateConsignmentOrderInd": "Y",
"generateSelfInvoicingInd": "Y",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"dealUploadStatus": "W",
"taxId": null,
"procurementSiteInd": "N",
"paymentSiteInd": "N",
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]
},
"supplierSites": [
  {
    "supplierSiteReferenceNo": "2",
    "supplierSite": 3026001,
    "attributes": {
      "supplierName": "Prime Hardware",
      "supplierNameSecondary": "Prime Hardware 2",
      "contactName": "Alexandre Roma",
      "contactPhone": "234-112-7654",
      "contactFax": "234-112-7654",
      "contactPager": null,
      "supplierStatus": "A",
      "qualityControlPercentage": 3,
      "qualityControlFrequency": "5",
      "vendorCheckingPercentage": 3,
      "vendorCheckingFrequency": 2,
      "currencyCode": "USD",
      "language": 1,
      "languageReferenceId": null,
      "terms": "108",
      "termsReferenceId": null,
      "freightTerms": "03",
      "freightTermsReferenceId": null,
      "minReturnValue": 3.55,

```

```

    "returnCourier": "125",
    "handlingPercentage": 3,
    "ediChannel": 2,
    "costChangeVariancePercentage": 9.65,
    "costChangeVarianceAmt": 9.65,
    "shipMethod": "30",
    "paymentMethod": "0A",
    "contactTelex": "234-112-7654",
    "contactEmail": "a_roma@primehardware.com",
    "vatRegion": 1000,
    "invoicePayLocation": "S",
    "invoiceReceiveLocation": "C",
    "comments": null,
    "defaultItemLeadTime": 7,
    "dunsNumber": "108890",
    "dunsLocation": "1055",
    "defaultPricingControl": "R",
    "generateConsignmentOrderInd": "Y",
    "generateSelfInvoicingInd": "Y",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ],
    "dealUploadStatus": "W",
    "taxId": null,
    "procurementSiteInd": "N",
    "paymentSiteInd": "N",
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  },
  "orgUnit": [
    {
      "orgUnitId": 1111111111,
      "orgUnitReferenceId": null,
      "primaryPaySite": "Y"
    }
  ],
  "address": [
    {
      "addressReferenceKey": "2",
      "addressKey": 114772,
      "address": {
        "addressType": "01",

```



```

        "primaryAddressInd": "Y",
        "add1": "200 Ryan Way",
        "add2": "Suite 100",
        "add3": null,
        "city": "Somerville",
        "state": "MA",
        "country": "US",
        "post": "55555-1234",
        "contactName": "Laura Johnson",
        "contactPhone": "617-897-0900",
        "contactFax": "617-897-0902",
        "contactEmail": "l_johnson@primehardware.com",
        "jurisdictionCode": null,
        "customFlexAttribute": [
            {
                "name": "REVIEW_DATE",
                "value": null,
                "valueDate": "2001-12-31"
            }
        ],
        "contactTelex": null
    }
}
]
}
]
}
]
}
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "items": [
    {
      "supplierReferenceNo": "1",
      "supplier": 3026,
      "vendorTypeCode": null,
      "attributes": {
        "supplierName": "Prime Hardware",
        "supplierNameSecondary": "Prime Hardware 2",
        "contactName": "Alexandre Roma",
        "contactPhone": "234-112-7654",
        "contactFax": "234-112-7654",
        "contactPager": null,
        "supplierStatus": "A",
        "qualityControlPercentage": 3,
        "qualityControlFrequency": "5",
        "vendorCheckingPercentage": 3,
        "vendorCheckingFrequency": 2,
        "currencyCode": "USD",
        "language": 1,
        "languageReferenceId": null,

```

```

"terms": "108",
"termsReferenceId": null,
"freightTerms": "03",
"freightTermsReferenceId": null,
"minReturnValue": 3.55,
"returnCourier": "125",
"handlingPercentage": 3,
"ediChannel": 2,
"costChangeVariancePercentage": 9.65,
"costChangeVarianceAmt": 9.65,
"shipMethod": "30",
"paymentMethod": "0A",
"contactTelex": "234-112-7654",
"contactEmail": "a_roma@primehardware.com",
"vatRegion": 1000,
"invoicePayLocation": "S",
"invoiceReceiveLocation": "C",
"comments": null,
"defaultItemLeadTime": 7,
"dunsNumber": "108890",
"dunsLocation": "1055",
"defaultPricingControl": "R",
"generateConsignmentOrderInd": "Y",
"generateSelfInvoicingInd": "Y",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"dealUploadStatus": "W",
"taxId": null,
"procurementSiteInd": "N",
"paymentSiteInd": "N",
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]
},
"supplierSites": [
  {
    "supplierSiteReferenceNo": "2",
    "supplierSite": 3026001,
    "attributes": {
      "supplierName": "Prime Hardware",
      "supplierNameSecondary": "Prime Hardware 2",
      "contactName": "Alexandre Roma",

```

```

"contactPhone": "234-112-7654",
"contactFax": "234-112-7654",
"contactPager": null,
"supplierStatus": "A",
"qualityControlPercentage": 3,
"qualityControlFrequency": "5",
"vendorCheckingPercentage": 3,
"vendorCheckingFrequency": 2,
"currencyCode": "USD",
"language": 1,
"languageReferenceId": null,
"terms": "108",
"termsReferenceId": null,
"freightTerms": "03",
"freightTermsReferenceId": null,
"minReturnValue": 3.55,
"returnCourier": "125",
"handlingPercentage": 3,
"ediChannel": 2,
"costChangeVariancePercentage": 9.65,
"costChangeVarianceAmt": 9.65,
"shipMethod": "30",
"paymentMethod": "0A",
"contactTelex": "234-112-7654",
"contactEmail": "a_roma@primehardware.com",
"vatRegion": 1000,
"invoicePayLocation": "S",
"invoiceReceiveLocation": "C",
"comments": null,
"defaultItemLeadTime": 7,
"dunsNumber": "108890",
"dunsLocation": "1055",
"defaultPricingControl": "R",
"generateConsignmentOrderInd": "Y",
"generateSelfInvoicingInd": "Y",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"dealUploadStatus": "W",
"taxId": null,
"procurementSiteInd": "N",
"paymentSiteInd": "N",
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]

```

```

    }
  ],
  "orgUnit": [
    {
      "orgUnitId": 1111111111,
      "orgUnitReferenceId": null,
      "primaryPaySite": "Y"
    }
  ],
  "address": [
    {
      "addressReferenceKey": "2",
      "addressKey": 114772,
      "address": {
        "addressType": "01",
        "primaryAddressInd": "Y",
        "add1": "200 Ryan Way",
        "add2": "Suite 100",
        "add3": null,
        "city": "Somerville",
        "state": "MA",
        "country": "US",
        "post": "55555-1234",
        "contactName": "Laura Johnson",
        "contactPhone": "617-897-0900",
        "contactFax": "617-897-0902",
        "contactEmail": "l_johnson@primehardware.com",
        "jurisdictionCode": null,
        "customFlexAttribute": [
          {
            "name": "REVIEW_DATE",
            "value": null,
            "valueDate": "2001-12-31"
          }
        ],
        "contactTelex": null
      }
    }
  ]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response is returned. The element `validationErrors` is present when the input payload or input parameters do not follow the schema definition of this service. The element `businessError` is present if the payload passes schema validation but an exception is caught while processing the business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Supplier Download Service**Supplier Detail Service**

This section describes the Supplier Detail service.

Business Overview

This service publishes supplier-site master data to downstream consuming systems. It publishes the supplier-site definition together with primary and additional addresses, supplier org-unit relationships, and supplier and address custom flex attributes. Functionally, only supplier sites are published because both the main view query and the address CTEs require SUPS.SUPPLIER_PARENT IS NOT NULL; parent supplier rows are excluded.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: SUPS, ADDR

Additional business tables: ADD_TYPE_MODULE, V_ADD_TYPE_TL, PARTNER_ORG_UNIT, SYSTEM_OPTIONS, SUPS_CFA_EXT, ADDR_CFA_EXT

JSON cache table: MERCHAPI_EXT_SUPPLIER

JSON generation view: V_MERCHAPI_EXT_SUPPLIER_JSON

Builds the supplier-site header from SUPS, derives the primary address from ADDR joined to ADD_TYPE_MODULE and V_ADD_TYPE_TL, aggregates all supplier-site addresses from ADDR, adds supplier-site org-unit relationships from PARTNER_ORG_UNIT, and enriches supplier- and address-level custom flex attributes from MERCHAPI_EXT_SUPPLIER_CFA and MERCHAPI_EXT_SUPPLIER_ADDR_CFA.

The view also reads SYSTEM_OPTIONS to derive defaultPricingControl.

Functionally, only supplier sites are published because both the main view query and the address CTEs require SUPS.SUPPLIER_PARENT IS NOT NULL; parent supplier rows are excluded.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_SUPPLIER_JSON and apply supplier-site rows only where SUPS.SUPPLIER_PARENT IS NOT NULL.

The current REBUILD branch merges changed and new rows only; it does not include a delete-marking pass for supplier sites that disappear from V_MERCHAPI_EXT_SUPPLIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_SUPPLIER and ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on SUPS, and address changes are captured through ICL_RMS_ADDR for ADDR rows where MODULE = 'SUPP'.

ICL_RMS_ADDR rows are filtered to MODULE = 'SUPP' and republish the parent supplier-site payload.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_SUPPLIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_SUPPLIER_ADHOC_PROCESS ->
MERCHAPI_EXT_SUPPLIER_ADHOC_JOB

Webhook configuration api name: foundation/supplier

Service Type

Get

ReST URL

MerchIntegrations/services/foundation/supplier

MerchIntegrations/services/foundation/supplier/{supplierId}

Functional Area

Foundation - Suppliers

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Supplier)

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-176 Input Parameter for "MerchIntegrations/services/foundation/supplier/{supplierId}"

Parameter Name	Required	Data Type	Description
supplierId	Yes	Number	Supplier Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-177 PageResultsItemsMerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch

Table 5-177 (Cont.) PageResultsItemsMerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-178 MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String	Action for supplier details.
supplier	No	Number(10,0)	This field specifies the unique identifying number for a supplier site within the system.
supplierName	No	String (240)	This field specifies the supplier site's name.
supplierNameSecondary	No	String (240)	This field specifies the secondary name of the supplier.
supplierStatus	No	String (1)	This field specifies whether the supplier is currently active.
parentSupplier	No	Number(10,0)	This field contains the supplier number of the supplier parent.
contactName	No	String (120)	This field contains the name of the supplier's representative contact for this site.
contactPhone	No	String (20)	This field contains the telephone number for the supplier's representative contact.
contactFax	No	String (20)	This field contains the fax number for the supplier's representative contact.
contactPager	No	String (20)	This field contains the number for the pager of the supplier's representative contact.
contactTelex	No	String (20)	This field contains the telex number of the partner's or supplier's representative contact.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactEmail	No	String (100)	This field contains the email address of the partner's or supplier's representative contact.
qualityControlInd	No	String (1)	This field determines whether orders from this supplier will default as requiring quality control.
qualityControlPercentage	No	Number(12,4)	This field specifies the percentage of items per receipt that will be marked for quality checking.
qualityControlFrequency	No	Number(2,0)	This field specifies the frequency in which items per receipt will be marked for quality checking.
vendorCheckingInd	No	String (1)	This field specifies whether orders from this supplier will default as requiring vendor control.
vendorCheckingPercentage	No	Number(12,4)	This field specifies the percentage of items per receipt that will be marked for vendor checking.
vendorCheckingFrequency	No	Number(2,0)	This field specifies the frequency in which items per receipt will be marked for vendor checking.
currencyCode	No	String (3)	This field specifies the code identifying the currency the supplier uses for business transactions.
language	No	Number(6,0)	This field contains the supplier's preferred language.
terms	No	String (15)	This field contains an indicator identifying the purchase terms that will default when an order is created for the supplier site. These terms specify when payment is due and if any discounts exist for early payment.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	This field contains the code indicating what freight terms will default when an order is created for the supplier site.
returnAllowedInd	No	String (1)	This field specifies whether or not the supplier will accept returns.
returnAuthorizationRequiredInd	No	String (1)	This field specifies if returns must be accompanied by an authorization number when sent back to the vendor.
minimumReturnValue	No	Number(20,4)	This field contains a value if the supplier site requires a minimum merchandise value to be returned to accept the return. Returns of less than this amount will not be processed by the system. This field is stored in the supplier's currency.
returnCourier	No	String (250)	This field specifies the name of the courier that should be used for all returns to the supplier site.
handlingPercentage	No	Number(12,4)	This field contains the default percent to be multiplied by the return's total cost to determine the handling cost for the return.
ediOrderInd	No	String (1)	This field specifies whether purchase orders will be sent to the supplier via EDI.
ediOrderChangeInd	No	String (1)	This field specifies whether purchase order changes will be sent to the supplier via EDI.
ediOrderConfirmInd	No	String (1)	This field specifies whether this supplier will send acknowledgment of purchase orders sent via EDI.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ediAsnInd	No	String (1)	This field specifies whether this supplier will send Advance Shipment Notifications electronically.
ediSalesReportFrequency	No	String (1)	This field specifies the EDI sales report frequency for this supplier.
ediSupplierAvailabilityInd	No	String (1)	This field specifies whether the supplier will send availability via EDI.
ediContractInd	No	String (1)	This field indicates whether the supplier site supports contract ordering sent via EDI.
ediInvoiceInd	No	String (1)	This field specifies whether invoices, debit memos and credit note requests will be sent to/ from the supplier via Electronic Data Interchange.
ediChannelId	No	Number(4,0)	This field specifies if the supplier is an EDI supplier and supports vendor initiated ordering, this field will contain the channel ID for the channel to which all inventory for these types of orders will flow.
costChangeVariancePercentage	No	Number(12,4)	This field contains a percent that determines whether a cost change can be auto approved via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
costChangeVarianceAmt	No	Number(20,4)	This field contains an amount (in supplier currency) that determines whether a cost change can be auto approved via induction. If the cost change falls within these boundaries, it will be approved when uploaded.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
replenishmentApprovalInd	No	String (1)	This field indicates whether contract orders created via replenishment should be created in Approved status.
shipMethod	No	String (6)	This field contains the default method used to ship the items on the purchase order from the supplier site. Valid values are held in code type SHPM.
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are Letter of Credit (LC), Wire Transfer (WT) and Open Account (OA).
settlementCode	No	String (1)	This field indicates which payment process method is used for the supplier. Valid values are N/A (N) and Evaluated Receipts Settlement (E).
preMarkInd	No	String (1)	This field indicates whether the supplier site supports pre-marking containers for cross-dock orders.
autoApproveInvoiceInd	No	String (1)	This field indicates whether the supplier's invoices can be automatically approved for payment.
debitMemoCode	No	String (1)	This field indicates when a debit memo will be sent to the supplier site to resolve a discrepancy. Valid values are: Y - if debit memos are always to be sent, L - if debit memos are used only if a credit note is not sent by the invoice due date and N - if debit memos are never sent.
freightChargeInd	No	String (1)	This field indicates whether a supplier site can charge freight costs.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
autoApproveDebitMemoInd	No	String (1)	This field specifies whether debit memos sent to the supplier can be automatically approved on creation.
invManagementLevel	No	String (6)	This field indicates the level for managing supplier inventory information. Valid values are: supplier (S), supplier/location (L), supplier/department (D), and supplier/department/location (A).
backorderInd	No	String (1)	This field specifies if backorders or partial shipments will be accepted.
vatRegion	No	Number(4,0)	This field contains the unique identifying number for the VAT region applicable for this site.
prepayInvoiceInd	No	String (1)	This field specifies whether all invoices for the supplier can be pre-paid.
servicePerformedRequiredInd	No	String (1)	This field indicates if the supplier's services must be confirmed as performed before paying an invoice from that supplier site.
invoicePayLocation	No	String (6)	This field indicates where invoices from this supplier site are paid - at the store (S) or centrally through corporate accounting (C).
invoiceReceiveLocation	No	String (6)	This field indicates where invoices from this supplier site are received - at the store (S) or centrally through corporate accounting (C).
invoiceGrossNet	No	String (6)	This field indicates if the supplier site invoice lists items at gross cost (G) or net cost (N).

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deliveryPolicy	No	String (6)	This field contains the default delivery policy of the supplier site. Valid values are held in code type DLVY.
comments	No	String (2000)	This field contains any miscellaneous comments associated with the supplier.
defaultItemLeadTime	No	Number(4,0)	This field specifies the default lead time for the supplier site. The lead time is the time the supplier needs between receiving an order and having the order ready to ship. This value will be defaulted to item/supplier relationships.
dunsNo	No	String (9)	This field holds the Dun and Bradstreet number to identify the supplier.
dunsLocation	No	String (4)	This field holds the Dun and Bradstreet number to identify the location of the supplier.
bracketCostingInd	No	String (1)	This field will determine if the supplier site supports bracket costing pricing structures.
vmiOrderStatus	No	String (6)	This field determines the status in which any inbound POs from this supplier will be created. A NULL value indicates that the supplier is not a VMI supplier.
dsdSupplierInd	No	String (1)	This field specifies whether the vendor supports DSD ordering, where the supplier replenishes the store directly, creating the PO and receipt at the same time.
supplierQuantityLevel	No	String (6)	This field indicates the supplier site order quantity level. Valid values are cases (CA) or eaches (EA).

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
finalDestinationInd	No	String (1)	This field indicates whether the supplier site can ship to final destination or not.
externalReferenceId	No	String (32)	This field holds the ID for the supplier used in the external financial system.
statusUpdateByRms	No	String (1)	This field indicates that the supplier is inactivated from RMS. Suppliers inactivated in RMS will have this field updated as Y. Otherwise this field will be null.
defaultPricingControl	No	String (1)	This field contains the pricing control of the supplier and determines whether the supplier can control the price. Valid values are R - Managed by Retailer and S - Managed by Supplier.
primaryAddressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressAdd1	No	String (240)	This field contains the first line of the address.
primaryAddressAdd2	No	String (240)	This field contains the second line of the address.
primaryAddressAdd3	No	String (240)	This field contains the third line of the address.
primaryAddressCity	No	String (120)	This field contains the name of the city that is associated with the address.
primaryAddressState	No	String (3)	This field contains the state abbreviation for the address.
primaryAddressCountryId	No	String (3)	This field contains the country where the address exists.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressJurisdictionCode	No	String (10)	This field identifies the jurisdiction code for the country-state relationship.
primaryAddressPost	No	String (30)	This field contains the zip code for the address.
primaryAddressContact Name	No	String (120)	This field contains the name of the contact person at this address.
primaryAddressContact Phone	No	String (20)	This field contains the phone number of the contact person at this address.
primaryAddressContact Telex	No	String (20)	This field contains the telex number of the contact person at this address.
primaryAddressContact Fax	No	String (20)	This field contains the fax number of the contact person at this address.
primaryAddressContact Email	No	String (100)	This field contains the email address of the supplier site's contact person.
generateConsignmentOrderInd	No	String (1)	This field will allow specification of whether Merchandising is required to generate POs against consignment sales.
generateSelfInvoicingInd	No	String (1)	This field will allow specification of whether self-invoicing needs to be done for consignment purchases or if the vendor will send invoices to be matched.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the supplier.
address	No	Collection of Object	This list holds all supplier addresses.

Table 5-178 (Cont.) MerchApiSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierOrgUnit	No	Collection of Object	This list holds supplier or supplier site attributes by org unit ID. The list is populated when either Multiple Set of Books is on and/or Financial AP is on.
cacheTimestamp	No	dateTime	This field specifies the date and time when the supplier record was last fetched into the cache.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-179 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-180 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressKey	No	Number(11,0)	This field contains the unique address key.

Table 5-180 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
addressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressInd	No	String (1)	This field indicates whether the address is the primary address for the address type.
add1	No	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	No	String (120)	This field contains the name of the city that is associated with the address.
state	No	String (3)	This field contains the state abbreviation that is associated with the address.
stateName	No	String (120)	This field contains the state description that is associated with the address.
countryId	No	String (3)	This field contains the country where the address exists.
countryName	No	String (120)	This field contains the country description that is associated with the address.
post	No	String (30)	This field contains the zip code for the address.
contactName	No	String (120)	This field contains the name of the contact person at this address.
contactPhone	No	String (20)	This field contains the phone number of the contact person at this address.

Table 5-180 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactTelex	No	String (20)	This field contains the telex number of the contact person at this address.
contactFax	No	String (20)	This field contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This field contains the email address of the contact person at this address.
phone2Type	No	String (6)	This field contains the type of the phone number stored in the phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This field contains an additional phone number associated with the address.
phone3Type	No	String (6)	This field contains the type of the phone number stored in the phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This field contains a second additional phone number associated with the address.
latitude	No	String (30)	This field contains the north-south geographic coordinate of the address, represented in decimal degrees.

Table 5-180 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
longitude	No	String (30)	This field contains the east-west geographic coordinate of the address, represented in decimal degrees.
county	No	String (250)	This field contains the county where the address exists.
jurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse's address.

Table 5-181 SupplierOrgUnit - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orgUnitId	No	Number(15,0)	This field holds the org unit ID.
primaryPaySiteInd	No	String (1)	This field holds the primary payment site indicator.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).

Table 5-182 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-182 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "supplier": 3026001,
      "supplierName": "Prime Hardware",
      "supplierNameSecondary": "Prime Hardware 2",
      "supplierStatus": "A",
      "parentSupplier": 3026,
      "contactName": "Alexandre Roma",
      "contactPhone": "234-112-7654",
      "contactFax": "234-112-7654",
      "contactPager": null,
      "contactTelex": "234-112-7654",
      "contactEmail": "a_roma@primehardware.com",
      "qualityControlInd": "Y",
      "qualityControlPercentage": 3,
      "qualityControlFrequency": 5,
      "vendorCheckingInd": "Y",
      "vendorCheckingPercentage": 3,
      "vendorCheckingFrequency": 2,
      "currencyCode": "USD",
      "language": 1,
      "terms": "108",
      "freightTerms": "03",
      "returnAllowedInd": "Y",
      "returnAuthorizationRequiredInd": "N",
      "minimumReturnValue": 3.55,
      "returnCourier": "125",
      "handlingPercentage": 3,
      "ediOrderInd": "Y",
      "ediOrderChangeInd": "Y",
      "ediOrderConfirmInd": "Y",
      "ediAsnInd": "Y",
      "ediSalesReportFrequency": null,
      "ediSupplierAvailabilityInd": "Y",
      "ediContractInd": "Y",
      "ediInvoiceInd": "Y",
      "ediChannelId": 1,
      "costChangeVariancePercentage": 9.65,
    }
  ]
}
```

```

"costChangeVarianceAmt": 9.65,
"replenishmentApprovalInd": "Y",
"shipMethod": "30",
"paymentMethod": "OA",
"settlementCode": "N",
"preMarkInd": "Y",
"autoApproveInvoiceInd": "Y",
"debitMemoCode": "N",
"freightChargeInd": "Y",
"autoApproveDebitMemoInd": "Y",
"invManagementLevel": "S",
"backorderInd": "Y",
"vatRegion": 1000,
"prepayInvoiceInd": "Y",
"servicePerformedRequiredInd": "Y",
"invoicePayLocation": "S",
"invoiceReceiveLocation": "C",
"invoiceGrossNet": "G",
"deliveryPolicy": "NEXT",
"comments": null,
"defaultItemLeadTime": 7,
"dunsNo": "108890",
"dunsLocation": "1055",
"bracketCostingInd": "Y",
"vmiOrderStatus": null,
"dsdSupplierInd": "Y",
"supplierQuantityLevel": "EA",
"finalDestinationInd": null,
"externalReferenceId": null,
"statusUpdateByRms": null,
"defaultPricingControl": "R",
"primaryAddressType": "01",
"primaryAddressTypeDescription": null,
"primaryAddressAdd1": "200 Ryan Way",
"primaryAddressAdd2": "Suite 100",
"primaryAddressAdd3": null,
"primaryAddressCity": "Somerville",
"primaryAddressState": "MA",
"primaryAddressCountryId": "US",
"primaryAddressJurisdictionCode": null,
"primaryAddressPost": "55555-1234",
"primaryAddressContactName": "Laura Johnson",
"primaryAddressContactPhone": "617-897-0900",
"primaryAddressContactTelex": null,
"primaryAddressContactFax": "617-897-0902",
"primaryAddressContactEmail": "l_johnson@primehardware.com",
"generateConsignmentOrderInd": null,
"generateSelfInvoicingInd": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",

```

```

    "updateDateTime": "2001-12-31T23:59:59.000Z"
  },
  ],
  "address": [
    {
      "addressKey": 114772,
      "addressType": "01",
      "addressTypeDescription": null,
      "primaryAddressInd": "Y",
      "add1": "101 First Street",
      "add2": "201 Second Street",
      "add3": null,
      "city": "Minneapolis",
      "state": "MN",
      "stateName": "MN",
      "countryId": "US",
      "countryName": "US",
      "post": "55555-1234",
      "contactName": "Laura Johnson",
      "contactPhone": "1800800800",
      "contactTelex": null,
      "contactFax": "617-897-0902",
      "contactEmail": "someone@email.com",
      "phone2Type": "W0",
      "phone2": "5551234567",
      "phone3Type": "W0",
      "phone3": "5551234567",
      "latitude": "44.38129",
      "longitude": "-92.03521",
      "county": null,
      "jurisdictionCode": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ]
    }
  ],
  ],
  "supplierOrgUnit": [
    {
      "orgUnitId": 1111111111,
      "primaryPaySiteInd": "Y",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ],
  "cacheTimestamp": "2001-12-31T23:59:59.000Z",
  "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],

```

```
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SUPS	Yes	No	No	No
ADDR	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Supplier Publish Services

Endpoints

MerchIntegrations/services/foundation/supplier

MerchIntegrations/services/foundation/supplier/{supplierId}

Functional Area

Foundation – Suppliers and Partners

Business Overview

This service can be used by external applications to get all or selected supplier sites and the address information from Merchandising. This information is further subscribed by integrating systems like Oracle Retail Store Inventory and Operations Cloud Service (SIOCS). The address types included as part of the information are Returns, Order and Invoice address types.

Only supplier site level information is published. The supplier level information will not be published.

Service Type

GET

ReST URL

/MerchIntegrations/services/foundation/supplier?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}

/MerchIntegrations/services/foundation/supplier/{supplierId}

Input Parameters

Table 5-183 MerchIntegrations/services/foundation/supplier

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-184 /MerchIntegrations/services/ foundation/supplier/{supplierId}

Parameter Name	Required	Data Type	Description
supplierId	Yes	String	Supplier ID.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Name	Data Type	Description
action	String	Action for supplier details. Valid values are INSERT, UPDATE and DELETE.
supplier	Number	This field specifies unique identifying number for a supplier site within the system.
supplierName	String	This field specifies supplier's site's name.
supplierNameSecondary	String	This field specifies secondary name of the supplier.
supplierStatus	String	This field specifies whether the supplier is currently active.
parentSupplier	String	This field contains the supplier number for the supplier site.
contactName	String	This field contains the name of the supplier's representative contact for this site.

Name	Data Type	Description
contactPhone	String	This field contains telephone number for the supplier's representative contact.
contactFax	String	This field contains fax number for the supplier's representative contact.
contactPager	String	This field contains number for the pager of the supplier's representative contact.
contactTelex	String	This field contains telex number of the partner or supplier's representative contact.
contactEmail	String	This field contains email address of the partner or supplier's representative contact.
qualityControlInd	String	This field determines whether orders from this supplier will default as requiring quality control.
qualityControlPercentage	Number	This field specifies percentage of items per receipt that will be marked for quality checking.
qualityControlFrequency	Number	This field specifies frequency in which items per receipt will be marked for quality checking.
vendorCheckingInd	String	This field specifies whether orders from this supplier will default as requiring vendor control.
vendorCheckingPercentage	Number	This field specifies percentage of items per receipt that will be marked for vendor checking.
vendorCheckingFrequency	Number	This field specifies frequency in which items per receipt will be marked for vendor checking.
CurrencyCode	String	This field specifies code identifying the currency the supplier uses for business transactions.
language	String	This field contains the supplier's preferred language.
terms	String	This field contains an indicator identifying the purchase terms that will default when an order is created for the supplier site. These terms specify when payment is due and if any discounts exist for early payment.
freightTerms	String	This field contains code indicating what freight terms will default when an order is created for the supplier site.
returnAllowedInd	String	This field specifies whether the supplier will accept returns.
returnAuthorizationRequiredInd	String	This field specifies if returns must be accompanied by an authorization number when sent back to the vendor.

Name	Data Type	Description
minimumReturnValue	Number	This field contains a value if the supplier site requires a minimum merchandise value to be returned to accept the return. Returns of less than this amount will not be processed by the system. This field is stored in the supplier's currency.
returnCourier	String	This field specifies name of the courier that should be used for all returns to the supplier site.
handlingPercentage	Number	This field contains the default percent to be multiplied by the return's total cost to determine the handling cost for the return.
ediOrderInd	String	This field specifies whether purchase orders will be sent to the supplier via EDI.
ediOrderChangeInd	String	This field specifies whether purchase order changes will be sent to the supplier via EDI.
ediOrderConfirmInd	String	This field specifies whether this supplier will send acknowledgment of a purchase orders sent via EDI.
ediAsnInd	String	This field specifies whether this supplier will send Advance Shipment Notifications electronically.
ediSalesReportFrequency	String	This field specifies EDI sales report frequency for this supplier.
ediSupplierAvailabilityInd	String	This field specifies whether the supplier will send availability via EDI.
ediContractInd	String	This field indicates whether the supplier site supports contract ordering sent via EDI.
ediInvoiceInd	String	This field specifies whether invoices, debit memos and credit note requests will be sent to/from the supplier via Electronic Data Interchange.
EdiChannelId	String	This field specifies if the supplier is an EDI supplier and supports vendor-initiated ordering, this field will contain the channel ID for the channel to which all inventory for these types of orders will flow.
costChangeVariancePercentage	Number	This field contains a percent that determines whether a cost change can be auto approved via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
costChangeVarianceAmt	Number	This field contains an amount (in supplier currency) that determines whether a cost change can be auto approved via induction. If the cost change falls within these boundaries, it will be approved when uploaded.
replenishmentApprovalInd	String	This field indicates whether contract orders created via replenishment should be created in Approved status.

Name	Data Type	Description
shipMethod	String	This field contains the default method used to ship the items on the purchase order from the supplier site. Valid values are held in code type SHPM.
paymentMethod	String	This field indicates the default method for how purchase orders for this site will be paid. Valid values are Letter of Credit (LC), Wire Transfer (WT) and Open Account (OA).
settlementCode	String	This field indicates which payment process method is used for the supplier. Valid values are N/A (N) and Evaluated Receipts Settlement (E).
preMarkInd	String	This field indicates whether the supplier site supports pre-marking containers for cross dock order.
autoApproveInvoiceInd	String	This field indicates whether the supplier's invoices can be automatically approved for payment.
debitMemoCode	String	This field indicates when a debit memo will be sent to the supplier site to resolve a discrepancy. Valid values are: Y - if debit memos are always to be sent, L - if debit memos are used only if a credit note is not sent by the invoice due date and N - if debit memos are never sent.
freightChargeInd	String	This field indicates whether a supplier site can charge freight costs.
autoApproveDebitMemoInd	String	This field specifies whether debit memos sent to the supplier can be automatically approved on creation.
invManagementLevel	String	This field indicates the level for managing supplier inventory information. Valid values are: supplier (S), supplier/location (L), supplier/department (D), and supplier/department/location (A).
backorderInd	String	This field specifies if backorders or partial shipments will be accepted.
vatRegion	Number	This field contains the unique identifying number for the VAT region applicable for this site.
prepayInvoiceInd	String	This field specifies whether all invoices for the supplier can be pre-paid.
servicePerformedRequiredInd	String	This field indicates if the supplier's services must be confirmed as performed before paying an invoice from that supplier site.
invoicePayLocation	String	This field indicates where invoices from this supplier site are received - at the store (S) or centrally through corporate accounting (C).

Name	Data Type	Description
invoiceReceiveLocation	String	This field indicates where invoices from this supplier site are received - at the store (S) or centrally through corporate accounting (C).
invoiceGrossNet	String	This field indicates if the supplier site invoice lists items at gross cost (G) or net cost (N).
deliveryPolicy	String	This field contains the default delivery policy of the supplier site. Valid values are held in code type DLVY.
comments	String	This field contains any miscellaneous comments associated with the supplier.
defaultItemLeadTime	Number	This field specifies default lead time for the supplier site. The lead time is the time the supplier needs between receiving an order and having the order ready to ship. This value will be defaulted to item/supplier relationships.
dunsNo	String	This field holds the Dun and Bradstreet number to identify the supplier.
dunsLocation	String	This field holds the Dun and Bradstreet number to identify the location of the supplier.
bracketCostingInd	String	This field will determine if the supplier site supports bracket costing pricing structures.
vmiOrderStatus	String	This field determines the status in which any inbound POs from this supplier will be created. A NULL value indicates that the supplier is not a VMI supplier.
dsdSupplierInd	String	This field specifies whether the vendor supports DSD ordering, where the supplier replenishes the store directly, creating the PO and receipt at the same time.
supplierQuantityLevel	String	This field indicates the supplier site order quantity level. Valid values are cases (CA) or eaches (EA).
finalDestinationInd	String	This field indicates whether the supplier site can ship to final destination or not.
externalReferenceId	String	This field holds the ID for the supplier used in the external financial system.
statusUpdateByRms	String	This field indicates that Supplier is inactivated from RMS. Suppliers inactivated in RMS will have this field updated as Y. Otherwise this field will be null.
defaultPricingControl	String	Indicates the default value of the pricing control attribute at the supplier site level when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are R – Retailer and S – Supplier.

Name	Data Type	Description
primaryAddressType	String	This field specifies type for the address. Valid values are: 01 -Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	String	This field specifies address type description.
primaryAddressAdd1	String	This field contains the first line of the address.
primaryAddressAdd2	String	This field contains the second line of the address.
primaryAddressAdd3	String	This field contains the third line of the address.
primaryAddressCity	String	This field contains the name of the city that is associated with the address.
primaryAddressState	String	This field contains abbreviation of the state for the address
primaryAddressStateName	String	This field contains full name of the state for the address.
primaryAddressCountryId	String	This field contains the country where the address exists.
primaryAddressJurisdictionCode	String	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
primaryAddressPost	String	This field contains the zip code for the address.
primaryAddressContactName	String	This field contains the name of the contact for the supplier at this address.
primaryAddressContactPhone	String	This field contains the phone number of the contact person at this address.
primaryAddressContactTelex	String	This field contains the telex number of the partner or supplier's representative contact.
primaryAddressContactFax	String	This field contains the fax number of the contact person at this address.
primaryAddressContactEmail	String	This field contains the email address of the partner or supplier's representative contact.
generateConsignmentOrderInd	String	This field will allow specification of the whether Merchandising is required to generate POs against consignment sales.
generateSelfInvoicingInd	String	This field will allow specification of the whether the self-invoicing needs to be done for consignment Purchases or if Vendor will send invoices to be matched.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttribute		This List holds custom flex attributes for address details.
address		This list holds all the addresses of the supplier.

Name	Data Type	Description
supplierOrgUnit		This List holds list attributes for supplier/supplier site and Org unit id. List is populated when populated when either Multiple Set of Books is ON And/or Financial AP is O.
cacheTimestamp	String	This field specifies date and time when the supplier record was last maintained.

Table 5-185 address

Name	Data Type	Description
addressType	String	This field specifies type for the address. Valid values are: 01 -Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
addressTypeDescription	String	This field specifies address type description.
sequenceNo	Number	This field indicates the sequence that addresses within the same type were entered.
primaryAddressInd	String	This field indicates whether the address is the primary address for the address type.
add1	String	This field contains the first line of the address.
add2	String	This field contains the second line of the address.
add3	String	This field contains the third line of the address.
city	String	This field contains name of the city that is associated with the address.
state	String	This field contains name of the state that is associated with the address.
stateName	String	This field contains name of the state description that is associated with the address.
countryID	String	This field contains name of the country that is associated with the address.
jurisdictionCode	String	This field contains the ID associated to the tax jurisdiction of the country-state relationship.
post	String	This field contains the zip code for the address.
contactName	String	This field contains the name of the contact person at this address.
contactPhone	String	This field contains the phone number of the contact person at this address.
contactTelex	String	This field contains the telex number of the warehouse's representative contact.
contactFax	String	This field contains the fax number of the contact person at this address.
contactEmail	String	This field contains email address of the warehouse's representative contact.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttribute		This list holds custom flex attributes for the address.

Table 5-186 supplierOrgUnit

Name	Data Type	Description
orgUnitId	Number	This field holds org unit id.
primaryPaySiteInd	String	This field holds primary payment site indicator.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.

Table 5-187 customFlexAttribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Table 5-188 Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined, then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists, then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self": This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```
{
  "items": [
    {
      "action": null,
      "supplier": 2330,
```



```
"supplierName": "Dummy site 1",
"supplierNameSecondary": null,
"supplierStatus": "A",
"parentSupplier": "2300",
"contactName": "Shaurya S",
"contactPhone": "98787987",
"contactFax": null,
"contactPager": null,
"contactTelex": null,
"contactEmail": null,
"qualityControlInd": "N",
"qualityControlPercentage": null,
"qualityControlFrequency": null,
"vendorCheckingInd": "N",
"vendorCheckingPercentage": null,
"vendorCheckingFrequency": null,
"currencyCode": "USD",
"language": null,
"terms": "108",
"freightTerms": null,
"returnAllowedInd": "N",
"returnAuthorizationRequiredInd": "Y",
"minimumReturnValue": null,
"returnCourier": null,
"handlingPercentage": null,
"ediOrderInd": "N",
"ediOrderChangeInd": "N",
"ediOrderConfirmInd": "N",
"ediAsnInd": "N",
"ediSalesReportFrequency": null,
"ediSupplierAvailabilityInd": "N",
"ediContractInd": "N",
"ediInvoiceInd": "N",
"ediChannelId": null,
"costChangeVariancePercentage": 0,
"costChangeVarianceAmt": 0,
"replenishmentApprovalInd": "N",
"shipMethod": null,
"paymentMethod": null,
"settlementCode": "N",
"preMarkInd": "N",
"autoApproveInvoiceInd": "N",
"debitMemoCode": null,
"freightChargeInd": "N",
"autoApproveDebitMemoInd": "N",
"invManagementLevel": "D",
"backorderInd": "N",
"vatRegion": 1000,
"prepayInvoiceInd": "N",
"servicePerformedRequiredInd": "N",
"invoicePayLocation": null,
"invoiceReceiveLocation": null,
"invoiceGrossNet": "N",
"deliveryPolicy": "NEXT",
"comments": null,
"defaultItemLeadTime": null,
```

```

"dunsNo": null,
"dunsLocation": null,
"bracketCostingInd": "N",
"vmiOrderStatus": null,
"dsdSupplierInd": "N",
"supplierQuantityLevel": "EA",
"finalDestinationInd": "N",
"externalReferenceId": null,
"statusUpdateByRms": null,
"defaultPricingControl": null,
"primaryAddressType": "01",
"primaryAddressTypeDescription": "Business",
"primaryAddressAdd1": "test address",
"primaryAddressAdd2": null,
"primaryAddressAdd3": null,
"primaryAddressCity": "MN",
"primaryAddressState": "PA",
"primaryAddressCountryId": "US",
"primaryAddressJurisdictionCode": null,
"primaryAddressPost": "3182",
"primaryAddressContactName": null,
"primaryAddressContactPhone": null,
"primaryAddressContactTelex": null,
"primaryAddressContactFax": null,
"primaryAddressContactEmail": null,
"generateConsignmentOrderInd": "Y",
"generateSelfInvoicingInd": "Y",
"createDateTime": "2023-08-08T01:23:39.000Z",
"updateDateTime": "2023-10-26T17:07:34.000Z",
"customFlexAttribute": [
  {
    "name": "PARTS_ATTR",
    "value": "test",
    "valueDate": null,
    "createDateTime": "2023-08-09T01:25:13.000Z",
    "updateDateTime": "2023-08-09T01:33:18.000Z"
  },
],
"address": [
  {
    "addressType": "01",
    "addressTypeDescription": "Business",
    "sequenceNo": 1,
    "primaryAddressInd": "Y",
    "add1": "test address",
    "add2": null,
    "add3": null,
    "city": "MN",
    "state": "PA",
    "countryId": "US",
    "jurisdictionCode": null,
    "post": "3182",
    "contactName": null,
    "contactPhone": null,
    "contactTelex": null,
    "contactFax": null,
  }
]

```

```

        "contactEmail": null,
        "createDateTime": "2023-08-08T01:34:34.000Z",
        "updateDateTime": "2023-08-09T00:56:21.000Z",
        "customFlexAttribute": null
      },
      {
        "addressType": "02",
        "addressTypeDescription": "Postal",
        "sequenceNo": 1,
        "primaryAddressInd": "Y",
        "add1": "test address",
        "add2": null,
        "add3": null,
        "city": "MN",
        "state": "PA",
        "countryId": "US",
        "jurisdictionCode": null,
        "post": "3182",
        "contactName": null,
        "contactPhone": null,
        "contactTelex": null,
        "contactFax": null,
        "contactEmail": null,
        "createDateTime": "2023-08-08T01:34:35.000Z",
        "updateDateTime": "2023-08-09T00:57:07.000Z",
        "customFlexAttribute": null
      }
    ],
    "supplierOrgUnit": [
      {
        "orgUnitId": 111111111,
        "primaryPaySiteInd": "N",
        "createDateTime": "2023-08-08T01:36:30.000Z",
        "updateDateTime": "2023-08-08T01:36:30.000Z"
      }
    ],
    "cacheTimestamp": "2023-10-26T20:03:46.186Z"
  }
],
"hasMore": false,
"limit": 1,
"count": 1,
"links": null
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
SUPS	Yes	No	No	No
SUPS_CFA_EXT	Yes	No	No	No
ICL_RMS_SUPPLIER	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes
MERCHAPI_SUPPLIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_SUPPLIER_JSON	Yes	No	No	No

Supplier Upload Service

Manage Supplier Evaluation Codes

Functional Area

Foundation - Suppliers and Partners

Business Overview

The Supplier Evaluation Codes Service is used by the Supplier Evaluation (ORSE) module to integrate master glossary data to Merchandising. The glossary contains a list of codes and valid values for each code. It is expected that the complete glossary will be published in a single request and Merchandising will overlay the existing codes with the information received in the service call.

Service Type

PUT

ReST URL

MerchIntegrations/services/foundation/suppliers/evaluation/codes

Input Payload Details

Table 5-189 Codes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	

Table 5-190 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
type	Yes	String (255)	This contains the supplier evaluation code type. Valid values are ASSESSMENT_STATUS, AUDIT_STATUS, AUDIT_SCORE, SCORE, EVALUATION_TYPE, STATUS, and ITEM_STATUS.
code	Yes	String (255)	This is the supplier evaluation code which contains valid values for the code type.
locale	Yes	Collection of Object	

Table 5-191 Locale - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	String (255)	This field contains the Supplier Evaluation language code in which the translated text is maintained. The supported language codes are defined in SELG code type. Few examples are en_GB, en_US, sv, ru, nl, ja, zh_CN, it, pt_BR, de, fr, es. Any text in a non supported language will be ignored.
description	Yes	String (255)	This field contains the translated value of the code.

Table 5-192 CodesError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
type	Yes	String (255)	This contains the supplier evaluation code type. Valid values are ASSESSMENT_STATUS, AUDIT_STATUS, AUDIT_SCORE, SCORE, EVALUATION_TYPE, STATUS, and ITEM_STATUS.
code	Yes	String (255)	This is the supplier evaluation code which contains valid values for the code type.
errorMessage	Yes	String (2000)	This field contains the error message.

Sample Input Message

```
{
  "items": [
    {
      "type": "EVALUATION_TYPE",
      "code": "SITE_AUDIT",
      "locale": [
        {
          "language": "it",
          "description": "Verifica del sito"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-193 CodesError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
type	Yes	String (255)	This contains the supplier evaluation code type. Valid values are ASSESSMENT_STATUS, AUDIT_STATUS, AUDIT_SCORE, SCORE, EVALUATION_TYPE, STATUS, and ITEM_STATUS.
code	Yes	String (255)	This is the supplier evaluation code which contains valid values for the code type.
errorMessage	Yes	String (2000)	This field contains the error message.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "type": "EVALUATION_TYPE",
      "code": "SITE_AUDIT",
      "errorMessage": "More than one record for the same type, code and
language exist."
    }
  ]
}
```

Manage Supplier Evaluation Transactions**Functional Area**

Foundation - Suppliers and Partners

Business Overview

The Supplier Evaluation Transactions service is used by the Supplier Evaluation (ORSE) module for providing supplier evaluation related information to Merchandising. The information provided by this service includes detailed information about each evaluation event undertaken by ORSE at the supplier site level. This information can be used in Merchandising for decision making such as supplier activation or purchase order approval etc.

Service Type

PUT

ReST URL

MerchIntegrations/services/foundation/suppliers/evaluation/transactions

Input Payload Details

Table 5-194 Transactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	

Table 5-195 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	This is the supplier site.
transactionType	Yes	String (255)	This field determines the type of transaction. This identifies the message type as Audit, Assessment, Summary, Document or Site Score.
evaluationType	Yes	String (255)	This field contains the code specific to the supplier evaluation type. The default set of suggested supplier evaluation types will be Ethical, Environmental and Governance, however the retailer/portal owner can configure their own set of values in addition or instead of these
typeBusinessKey	No	String (255)	This field contains the code of the assessment type or the audit template.
typeBusinessKeyDescription	No	Collection of Object	This node contains the language and the corresponding translation of the typeBusinessKey.
recordBusinessKey	No	String (255)	This field contains the name of the assessment or audit code.
createdOn	No	date-time	This field contains the timestamp of the event.
dueDate	No	date	This field contains the due date of the latest assessment for the assessment type for this site.
status	No	String (255)	This field contains the status of the supplier evaluation.
due	No	String (5)	For assessments, this field indicates whether the latest assessment for the assessment type for this site is due. For audit events, this indicates whether the latest audit/visit based on the audit template for this site is due.
overDue	No	String (5)	For assessments, this field indicates whether the latest assessment for the assessment type for this site is overdue. For audit events, this indicates whether the latest audit/visit based on the audit template for this site is overdue.

Table 5-195 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
completionDate	No	date	For documents, this field contains the date on which the document was last published. For assessments, this field contains the timestamp that the most recent assessment at completed status for this assessment type for this site most recently moved to completed status. For audits, this field contains the timestamp that the most recent audit/visit at completed status for this audit template for this site most recently moved to completed status.
score	No	String (255)	This field contains the descriptive score of the most recent assessment at completed status for this assessment type for this site.
scoreValue	No	Number (10)	For assessments, this field contains the numeric score of the most recent assessment at completed status for this assessment type for this site. For audits, this field contains the numeric score of the most recent audit/visit at completed status for this audit template for this site. For supplier site score, this field contains the numeric value of the site's latest SE score.
auditOpenIssues	No	Number (10)	This field contains the count of open issues/non-conformances of the latest audit/visit based on the audit template for this site.
auditCompletedIssues	No	Number (10)	This field contains the count of completed issues/non-conformances of the latest audit/visit based on the audit template for this site.
auditTotalIssues	No	Number (10)	This field contains the count of total issues/non-conformances of the latest audit/visit based on the audit template for this site.

Table 5-196 TypeBusinessKeyDescription - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	String (255)	This field contains the Supplier Evaluation language code in which the translated text is maintained. The supported language codes are defined in SELG code type. Few examples are en_GB, en_US, sv, ru, nl, ja, zh_CN, it, pt_BR, de, fr, es. Any text in a non supported language will be ignored.

Table 5-196 (Cont.) TypeBusinessKeyDescription - Object. See list of elements for detail

Element Name	Required	Data Type	Description
typeDescription	Yes	String (255)	This field contains the translated value of the typeBusinessKeyDescription.

Table 5-197 TransactionsError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	This is the supplier site.
transactionType	Yes	String (255)	This field determines the type of transaction. This identifies the message type as Audit, Assessment, Summary, Document or Site Score.
evaluationType	Yes	String (255)	This field contains the code specific to the supplier evaluation type. The default set of suggested supplier evaluation types will be Ethical, Environmental and Governance, however the retailer/portal owner can configure their own set of values in addition or instead of these
typeBusinessKey	No	String (255)	This field contains the code of the assessment type or the audit template.
recordBusinessKey	No	String (255)	This field contains the name of the assessment or audit code.
errors	Yes	Array of String	Collection of validation errors for the supplier evaluation record.

Sample Input Message

```
{
  "items": [
    {
      "supplier": 30201,
      "transactionType": "Audit",
      "evaluationType": "ETHICAL",
      "typeBusinessKey": "SITE_AUDIT",
      "typeBusinessKeyDescription": [
        {
          "language": "es",
          "typeDescription": "Auditoría del sitio"
        }
      ],
      "recordBusinessKey": "AUD45",
      "createdOn": "2001-12-31T23:59:59.123Z",
      "dueDate": "2001-12-31",
      "status": "COMPLETED",
      "due": "true",
    }
  ]
}
```

```

        "overDue": "false",
        "completionDate": "2001-12-31",
        "score": "PASS",
        "scoreValue": 80,
        "auditOpenIssues": 1,
        "auditCompletedIssues": 7,
        "auditTotalIssues": 8
    }
}
}

```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-198 TransactionsError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	This is the supplier site.
transactionType	Yes	String (255)	This field determines the type of transaction. This identifies the message type as Audit, Assessment, Summary, Document or Site Score.
evaluationType	Yes	String (255)	This field contains the code specific to the supplier evaluation type. The default set of suggested supplier evaluation types will be Ethical, Environmental and Governance, however the retailer/portal owner can configure their own set of values in addition or instead of these
typeBusinessKey	No	String (255)	This field contains the code of the assessment type or the audit template.
recordBusinessKey	No	String (255)	This field contains the name of the assessment or audit code.
errors	Yes	Array of String	Collection of validation errors for the supplier evaluation record.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
}

```

```

    "businessError": [
      {
        "supplier": 30201,
        "transactionType": "Audit",
        "evaluationType": "ETHICAL",
        "typeBusinessKey": "SITE_AUDIT",
        "recordBusinessKey": "AUD45",
        "errors": [
          "Invalid evaluation type: ETHICAL"
        ]
      }
    ]
  }
}

```

Tax Rules

The following services are included in this functional area:

- [Tax Rule Download Service](#)
 - [Get Item Location Tax Rules](#)
 - [Get Store Tax Location](#)
 - [Get Tax Rule Groups](#)
- [Tax Rule Upload Service](#)

Tax Rule Download Service

The following services are included in this functional area:

- [Get Item Location Tax Rules](#)
- [Get Store Tax Location](#)
- [Get Tax Rule Groups](#)

Get Item Location Tax Rules

Functional Area

Global Tax Solution - Tax Services

Business Overview

This service retrieves the tax rules which determine how retail tax is calculated on a sales transaction for an item at the location level. This service will be used by external systems that require this information to obtain the tax rules for items being sold at a given store.

The service can retrieve tax rule details for all items at a given store, for all stores for a given item, or for all items at all stores depending on the provided input. For example, if a location is provided, the service will return information for the provided location.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_LOC

Additional business tables: GTS_RULES_CONFIG, VAT_CODES,
MERCHAPI_GTS_TAX_LOC, SYSTEM_OPTIONS

JSON cache table: MERCHAPI_GTS_TAX_LOC_ITEM

JSON generation view: V_MERCHAPI_GTS_TAX_RULE_JSON

The rule-detail portion of the payload comes from V_MERCHAPI_GTS_TAX_RULE_JSON, which is built from GTS_RULES_CONFIG and VAT_CODES.

Functionally, the GET query only returns rows whose item/location combination still exists in ITEM_LOC, and the package flow only runs when SYSTEM_OPTIONS.DEFAULT_TAX_TYPE = 'GTS'.

Delta and refresh processing use a multi-stage GTS calculation flow.

Initial Seeding and post enablement rebuild/repair

Mark cache rows deleted when a rule is no longer returned for an affected item in the current GTS calculation result.

Thread 1 refreshes MERCHAPI_GTS_TAX_LOC before item-level tax rows are refreshed.

ICL (Integration Change Log) table: ICL_RMS_ITEM_LOC plus the upstream tax-event feeder path (ICL consumer name = MERCHAPI)

Item/location changes enter through ICL_RMS_ITEM_LOC, and the upstream tax-event feeder path submits the calculation work used to refresh affected tax rows.

Item Approval Handling

On item approval, the main item publish flow writes store ITEM_LOC rows for approved transaction-level items to ICL_RMS_ITEM_LOC so this API is also queued when the item becomes approved.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_GTS_TAX_LOC_ITEM

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_GTS_TAX_LOC_ITEM_ADHOC_PROCESS ->
MERCHAPI_GTS_TAX_LOC_ITEM_ADHOC_JOB

Webhook configuration api name: foundation/taxLocationItem

Special handling

Additional GET query parameters: item, locationType, locationId.

The filters item, locationType, and locationId are not backed by dedicated cache-table indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Request query values are STORE|WAREHOUSE|EXTERNAL_FINISH, while payload values are S|W|E.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. Because this is the main shared GTS merchapi, the available configuration should be validated against the customer production item, location, and tax-rule profile, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing through the shared `API_GTS_TAX_LOC_ITEM` flow. Thread count is controlled through `API_NO_OF_THREADS`, and commit sizing is controlled through `API_COMMIT_MAX_COUNTER`.

Thread and commit settings should be chosen together based on expected item-location volume, average payload size, and the multi-stage shared tax calculation required to build and maintain the related GTS caches. Common starting points for large-volume runs are 8 or 16 threads. Higher settings such as 32 should be considered only after testing, and the delivered shared setting is currently 10.

`API_COMMIT_MAX_COUNTER` should be sized together with the thread setting. Typical values range from about 100 to 1,000 rows and can extend to 10,000 for simpler restartable workloads; the delivered shared setting is currently 10000 and should be changed only after validating restart behavior and database capacity. Where tax calculation cost is high, a lower thread count with a higher commit value may perform better.

REST fetch considerations

REST page size should be selected based on average payload size so the average response remains below about 10 MB per call. Common page-limit values are 1,000, 5,000, or 10,000 rows.

For full downloads, do not provide the `since` query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

`MerchIntegrations/services/foundation/taxLocationItem`

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
item	No	String	Item
locationType	No	String	Location Type
locationId	No	String (10)	Location
offsetkey	No	String	Offset key ([location type],[location], [item], Example: S,120091,Item-35)
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-199 PageResultsItemsMerchapiGtsTaxLocItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-199 (Cont.) PageResultsItemsMerchapiGtsTaxLocItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-200 MerchapiGtsTaxLocItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for tax location item details.
item	No	String (25)	The item ID of an approved sellable transaction-level item, including a vendor pack.
loctype	No	String (1)	The location type. Valid values are S - store, W - warehouse, E - external finisher.
location	No	Number(10,0)	The location ID, including stores, warehouses, and external finishers.
taxLocationId	No	Number(11,0)	A unique identifier of a group of tax locations.
ruleId	No	Number(15,0)	The tax rule unique ID.
taxRegionDestination	No	Number(4,0)	Destination Tax Region.
taxValuePerUnit	No	Number(20,10)	Tax value when tax calculation type is in per unit.
ruleName	No	String (300)	Tax rule name.
taxRegionSource	No	Number(4,0)	Source Tax Region.

Table 5-200 (Cont.) MerchapiGtsTaxLocItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
taxAuthorityId	No	String (60)	A specific tax authority that establishes the tax rate. The tax authority is normally a government body. For example, National, State, Province, City, and so on.
taxCalcType	No	String (6)	Define the tax calculation type. Possible values are 'P'percentage or 'U'nit'.
taxCode	No	String (6)	Tax code.
taxRate	No	Number(20,10)	Tax rate in percentage when tax calculation type is in percentage.
effectiveTaxRate	No	Number(20,10)	Tax over tax rate calculated based in predecessor tax rule when applied.
activeDate	No	date	Effective date when this tax rule becomes active.
endDate	No	date	End date when this tax rule expires.
currencyCode	No	String (3)	Contains the currency code for the tax rule. For unit tax application.
createDateTime	No	dateTime	This field holds the date and time the record was created.
updateDateTime	No	dateTime	This field holds the date and time the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the tax location item record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.
taxGroupId	No	String (60)	A unique identifier of a group of tax rules for a given tax location.

Table 5-201 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "item": "Item-35",
      "loctype": "S",
      "location": 120091,
      "taxLocationId": 10000000012,
      "ruleId": 1,
      "taxRegionDestination": 1,
      "taxValuePerUnit": 1.0,
      "ruleName": null,
      "taxRegionSource": 1,
      "taxAuthorityId": null,
      "taxCalcType": null,
      "taxCode": "S",
      "taxRate": 1,
      "effectiveTaxRate": 1.0,
      "activeDate": "2001-12-31",
      "endDate": "2001-12-31",
      "currencyCode": "USD",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z",
      "taxGroupId": null
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

```
]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_GTS_TAX_LOC_ITEM	Yes	No	No	No
MERCHAPI_GTS_TAX_LOC	Yes	No	No	No

Get Store Tax Location**Functional Area**

Global Tax Solution - Tax Services

Business Overview

This service can be used by external applications to obtain the tax location for a provided store or all stores.

In Merchandising, Tax (or VAT) Regions are associated with store definitions. These tax regions can be, or can have, a parent tax region. The tax region associated with a location must be at the lowest level, meaning a parent region cannot be assigned to a location. Store definitions may also have custom flexible attributes (CFAs) defined. If one or more of the CFAs associated with the store are flagged as taxable, it is possible to create tax rules that utilize these CFAs.

The Global Tax Solution (GTS) determines a store's tax location based on the store's tax region and the CFAs applied to the location and to tax rules. A location with CFAs used on a tax rule may have a different tax location from other locations that have the same tax region. If all stores are placed in the same tax region and there are no location level CFAs used on a GTS tax rule, all of these stores will have the same tax location, such as 10000001000, where 1000 is the tax region identifier. The CFA for locations is used in GTS as exception criteria for applying tax rules. For instance, all locations have the same tax region but one store qualifies for a benefit from the government to have a reduce rate. This store will have a CFA to reflect designation and rules to use the CFA. In this case, this store with the CFA will be placed in a different tax location than the other stores in the same tax region.

If a location is provided, the service will return that store's tax location, if no location is provided the system will return all stores and their associated tax location. In order to fetch new updates pass the parameter of since/before timestamp.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: STORE

Additional business tables: GTS_RULES_CFA, SYSTEM_OPTIONS

JSON cache table: MERCHAPI_GTS_TAX_LOC

JSON generation view: not applicable; cache rows are populated directly by package logic

Functionally, the current delta and refresh logic only builds store rows (LOC_TYPE = 'S') and only for stores with a non-null STORE.VAT_REGION.

Stores without a matching GTS CFA restriction are assigned a tax location ID based on tax region, while stores with matching CFA restrictions are assigned a location-specific tax location ID.

The package compares the derived tax-location set with existing MERCHAPI_GTS_TAX_LOC rows to determine insert, merge, and delete behavior, and rebuild refresh marks cache rows deleted when the store no longer exists in STORE.

Initial Seeding and post enablement rebuild/repair

This service relies on the shared initial-seeding and rebuild flow used by foundation/taxLocationItem.

Rebuild marks cache rows deleted when the store no longer exists in STORE.

ICL (Integration Change Log) table: ICL_RMS_STORE (ICL consumer name = MERCHAPI_3)

This service relies on the shared GTS change-capture and delta-processing flow used by foundation/taxLocationItem.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_GTS_TAX_LOC_ITEM

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_GTS_TAX_LOC_ITEM_ADHOC_PROCESS ->
MERCHAPI_GTS_TAX_LOC_ITEM_ADHOC_JOB

Webhook configuration api name: foundation/taxLocation

Special handling

Additional GET query parameters: locationType, locationId.

This service shares the same GTS process family as foundation/taxLocationItem and foundation/taxGroupRule.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/taxLocation

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)

Parameter Name	Required	Data Type	Description
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
locationType	No	String	Location Type
locationId	No	String (10)	Location
offsetkey	No	String	Offset key ([location type],[location],[tax location id] Example: S,120091,10000000012)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-202 PageResultsItemsMerchapiGtsTaxLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-203 MerchapiGtsTaxLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for tax location details.
location	No	Number(10,0)	The location ID of the store.
locationType	No	String (1)	The location type. Valid value is S - store.
taxLocationId	No	Number(11,0)	A unique identifier of a group of tax locations.
createDateTime	No	dateTime	This field holds the date and time the record was created.
updateDateTime	No	dateTime	This field holds the date and time the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the tax location record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-204 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "location": 120091,
      "locationType": "S",
      "taxLocationId": 10000001000,

```

```
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_GTS_TAX_LOC	Yes	No	No	No

Get Tax Rule Groups

Functional Area

Global Tax Solution - Tax Services

Business Overview

This service retrieves the tax groups and associated tax rules for use by external systems. Tax rule groups are determined by Merchandising based on how many tax rules are applied to a given item/tax region combination.

Tax rules are defined for tax regions and a set of items. Tax rules contain calculation details that inform the system how to calculate the tax. A given item/location may have one or more applicable tax rules.

Tax Rules

Tax Rule | Items | Tax Region | Tax Calculation\

Tax Rule A | 111, 222, 333 | Main Region | Standard VAT 16%\

Tax Rule B | 444, 555, 666 | Main Region | Reduced VAT 0%\

Tax Rule C | 111, 222, 333 | Border Region | Standard VAT 8%\

Tax Rule D | 444, 555, 666 | Border Region | Reduced VAT 0%\

Tax Rule E | 222, 444 | Main Region | IEPS Tax 2.00 per unit\

Tax Rule F | 333, 555 | Border Region | IEPS Tax 1%

Each store in Merchandising will have an associated tax region:

Store Definitions

Location | Tax Region\

Store A | Main Region\

Store B | Main Region\

Store C | Border Region\

Store D | Border Region

Merchandising will organize the tax rules into tax groups using the tax rule definitions and location's tax region associations.

For any case where one or more items have a unique tax rule, or a unique set of tax rules at one or more stores, the system will create a tax group for the tax rule or group of tax rules.

Steps to determine the Tax Group for a given item/location:

Find the store's tax location. The tax location identifier is derived based on the location's tax region and the association of the location's CFAs to one or more tax rules. A store with CFAs used on a tax rule may have a different tax location from other stores that are part of the same tax region.

Using the item and the store's tax region, find the applicable tax rules.

For each unique result, whether it is one tax rule or multiple, the system will assign a new tax group identifier.

The table below shows the associated tax group for Items 111, 222, 333, 444, 555, or 666 at either the Main Region or Border Region based on the Tax Rules and Store Definitions listed above. Both items 111 and 333 at stores associated with the Main tax region, only require Tax Rule A, this is Group 1. There is one item that requires Tax Rule B, and it is item 666 at stores in the Main Region, this is Group 2. Item 222 at stores in the Main tax region have both Tax Rule A and Tax Rule E, this is Group 5. The table below shows the tax groups for this set of items and tax regions. This service will return the Tax Groups, the Tax Regions (referred to as Tax Location in the output) and the associated tax rules.

Tax Groups

Item | Tax Region | Tax Rule or Rules Applied | Tax Group\

111, 333 | Main Region | Tax Rule A | Group 1\

666 | Main Region | Tax Rule B | Group 2\

111, 222 | Border Region | Tax Rule C | Group 3\

444, 666 | Border Region | Tax Rule D | Group 4\

222 | Main Region | Tax Rule A and Tax Rule E | Group 5\

444 | Main Region | Tax Rule B and Tax Rule E | Group 6\

333 | Border Region | Tax Rule C and Tax Rule F | Group 7\

555 | Border Region | Tax Rule D and Tax Rule F | Group 8

The Tax Authority provided in the output of this service is derived by Merchandising Tax Codes setup (vat_codes).

In order to fetch new updates pass the parameter of since/before timestamp.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: MERCHAPI_GTS_TAX_LOC_ITEM, GTS_RULES_CONFIG

Additional business tables: VAT_CODES

JSON cache table: MERCHAPI_GTS_TAX_GROUP_RULE

JSON generation view: not applicable; cache rows are populated directly by package logic

The package derives the published groups from MERCHAPI_GTS_TAX_LOC_ITEM, enriches rule names from GTS_RULES_CONFIG, and pulls rule-detail JSON from V_MERCHAPI_GTS_TAX_RULE_JSON, which is built from GTS_RULES_CONFIG and VAT_CODES.

Functionally, the service publishes one row per (tax_group_id, tax_loc_id, rule_id) and represents the unique set of tax rules that applies to an item/tax-location combination.

To build a tax group, the package concatenates the ordered rule IDs with LISTAGG, hashes that string with ORA_HASH to derive tax_group_id, concatenates the corresponding rule names into group_name, and assigns rule_sequence by ascending rule_id.

Initial Seeding and post enablement rebuild/repair

This service relies on the shared initial-seeding and rebuild flow used by foundation/taxLocationItem.

Mark cache rows deleted when a rule is no longer returned for an affected item in the current GTS calculation result.

Rebuild updates or inserts changed rows; no separate rebuild delete step was found in this package.

ICL (Integration Change Log) table: ICL_RMS_GTS_RULES_CONFIG delete events (ICL consumer name = MERCHAPI_3)

This service relies on the shared GTS change-capture and delta-processing flow used by foundation/taxLocationItem.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_GTS_TAX_LOC_ITEM

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_GTS_TAX_LOC_ITEM_ADHOC_PROCESS ->
MERCHAPI_GTS_TAX_LOC_ITEM_ADHOC_JOB

Webhook configuration api name: foundation/taxGroupRule

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/taxGroupRule

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key ([Tax Group Id],[Tax location], [Tax Rule Id] Example: 43242,5435345,12314254)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Response Code: 200 (Success)

Table 5-205 PageResultsItemsMerchapiGtsTaxGroupRule - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object

Table 5-205 (Cont.) PageResultsItemsMerchapiGtsTaxGroupRule - Object. See list of elements for detail

Element Name	Required	Data Type	Description
links	No	Collection of Object	List of Self and/or Next URL

Table 5-206 MerchapiGtsTaxGroupRule - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for tax group rule details.
taxGroupId	No	String (60)	A unique identifier of a group of tax rules for a given tax location.
groupName	No	String (254)	Tax group name.
taxLocationId	No	Number(11,0)	A unique identifier of a group of tax locations.
ruleId	No	Number(15,0)	Tax rule unique ID.
ruleSequence	No	Number(15,0)	A sequential number for the tax rule within the group.
ruleName	No	String (300)	Tax rule name.
taxRegionSource	No	Number(4,0)	Source Tax Region.
taxRegionDestination	No	Number(4,0)	Destination Tax Region.
taxAuthorityId	No	String (60)	A specific tax authority that establishes the tax rate. The tax authority is normally a government body. For example, National, State, Province, City, and so on.
taxCalcType	No	String (6)	Define the tax calculation type. Possible values are 'P'percentage or 'U'nit'.
taxCode	No	String (6)	Tax code.
taxRate	No	Number(20,10)	Tax rate in percentage when tax calculation type is in percentage.
effectiveTaxRate	No	Number(20,10)	Tax over tax rate calculated based in predecessor tax rule when applied.
activeDate	No	date	Effective date when this tax rule becomes active.

Table 5-206 (Cont.) MerchapiGtsTaxGroupRule - Object. See list of elements for detail

Element Name	Required	Data Type	Description
endDate	No	date	End date when this tax rule expires.
currencyCode	No	String (3)	Contains the currency code for the tax rule. For unit tax application.
taxValuePerUnit	No	Number(20,10)	Tax value when tax calculation type is in per unit.
createDateTime	No	dateTime	This field holds the date and time the record was created.
updateDateTime	No	dateTime	This field holds the date and time the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the tax group rule record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-207 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "taxGroupId": "43242",
      "groupName": "Tax Rule A|Tax Rule E",
      "taxLocationId": 10000001000,

```

```

"ruleId": 12314254,
"ruleSequence": 1,
"ruleName": "Tax Rule A",
"taxRegionSource": 1,
"taxRegionDestination": 1,
"taxAuthorityId": null,
"taxCalcType": null,
"taxCode": "S",
"taxRate": 1,
"effectiveTaxRate": 1.0,
"activeDate": "2001-12-31",
"endDate": "2001-12-31",
"currencyCode": "USD",
"taxValuePerUnit": 1.0,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than the limit defined, then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists, then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	“self”: This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Table 5-208 Items - Object. See list of elements for detail

Name	Data Type	Description
action	String	Action for tax group rules details. Valid Values are INSERT, UPSERT, UPDATE and DELETE.
taxGroupId	String	The tax group identifier is determined by Merchandising based on the criteria applied on the combinations of location tax identification, tax rules and items.
groupName	String	Group Name
taxLocationId	String	The tax location identifier is derived based on the location's tax region and the association of the location's CFAs to one or more tax rules. A store with CFAs used on a tax rule may have a different tax location from other stores that are part of the same tax region.
ruleId	String	Tax Rule identifier
ruleSequence	String	Sequence of the rule within the tax group
ruleName	String	Tax Rule Name
taxRegionSource	String	Source Tax Region.
taxRegionDestination	String	Destination Tax Region
taxAuthorityId	String	A specific tax authority that establishes the tax rate. The tax authority is normally a government body. For example, National, State, Province, City, and so on.
taxCalcType	String	The tax calculation type which indicates if the tax is calculated as a percentage of a value or as a monetary amount per unit. Possible values are Percentage (P) or Per Unit (U).
taxCode	String	Tax code
taxRate	String	Tax rate in percentage when tax calculation type is Percentage (P).
effectiveTaxRate	String	Tax over tax rate calculated based on predecessor tax rule when applied
taxValuePerUnit	String	Tax value when tax calculation type is Per Unit (U).
activeDate	String	Date upon which this tax rule became, or will become active.
endDate	String	Date upon which this tax rule expired or will expire.
currencyCode	String	Contains the currency code for the tax rule. For per unit tax application.
createDateTime	String	This field holds the record creation date.

Table 5-208 (Cont.) Items - Object. See list of elements for detail

Name	Data Type	Description
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when the tax group record was last maintained.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "taxGroupId": 2654787678,
      "groupName": "Rule 55003",
      "taxLocationId": 10000000011,
      "ruleId": "55003",
      "createDateTime": "2024-07-19T02:37:21.757Z",
      "updateDateTime": "2024-07-19T02:37:21.757Z",
      "cacheTimestamp": "2024-07-19T02:37:21.757Z",
      "ruleName": "Rule 55003",
      "taxRegionSource": 1,
      "taxRegionDestination": 11,
      "taxAuthorityId": null,
      "taxCalcType": "P",
      "taxCode": "MIV2",
      "taxRate": 0,
      "effectiveTaxRate": null,
      "taxValuePerUnity": null,
      "activeDate": "2024-04-23",
      "endDate": null,
      "currencyCode": null,
      "ruleSequence": "1"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/foundation/taxGroupRule?limit=1&since=1970-01-01T00:00:00.001Z&before=2024-07-24T17:26:51.588Z",
      "rel": "self"
    },
    {
      "href": "https://<server>:<port>/Application/services/foundation/taxGroupRule?offsetkey=2654787678%2C10000000011%2C55003&limit=1&since=1970-01-01T00:00:00.001Z&before=2024-07-24T17:26:51.588Z",
      "rel": "next"
    }
  ]
}
```

```
    ]
  }
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_GTS_TAX_GROUP_RULE	Yes	No	No	No

Tax Rule Upload Service**Functional Area**

Foundation - Tax Rules

Business Overview

The primary role of this service is to create, update or delete Merchandising's Tax Rules.

Create Tax Rule**Business Overview**

Create a new Tax Rule.

Service Type

POST

ReST URL

/foundation/taxRules/create

Input Parameters

Parameter Name	Required	Data Type	Description
ruleId	No	Number (15)	Tax rule unique ID.
ruleName	No	String (300)	Tax Rule Name.
taxRegionSource	No	Number (4)	Source Tax Region.
taxRegionDestination	No	Number (4)	Destination Tax Region.
activeDate	No	date	Effective date when this tax rule become active.
endDate	No	date	End date when this tax rule expired.
status	No	String (6)	Status of the Rule. Valid Values are 'AC' for Active, 'AP' for Approved, 'O' Obsoleted, 'S' for Submitted, 'U' for Updated, 'W' for Worksheet.
taxCode	No	String (6)	Tax code.

Parameter Name	Required	Data Type	Description
taxCalcType	No	String (6)	Define the tax calculation type. Possible values are 'P' for Percentage or 'U' for Unit'.
taxRate	No	Number (20,10)	Tax rate in percentage when tax calculation type is in percentage.
taxBasisType	No	String (6)	Indicate which value is used as base calculation to apply tax. Valid values are: 'C' for UNIT_COST, 'R' for UNIT_RETAIL, or 'B' for BOTH, 'N' ON_MERCH_COST.
taxBasisFactor	No	Number (20,10)	Contains a unique user specified code representing the non-merchandise code.
nonMerchCode	No	String (6)	Contains a unique user specified code representing the non-merchandise code.
uomType	No	String (4)	UOM type selected to be considered when tax calculation is in unit.
uomValue	No	Number (20,10)	UOM value selected to be considered when tax calculation is in unit.
currencyCode	No	String (3)	Contains the currency code for the tax rule. For unit tax application.
reverseTaxInd	No	String (1)	Reverse Tax Indicator: Valid values are 'Y' for Enable or 'N' for Disable.
reverseTaxThreshold	No	Number (20,10)	Threshold amount to be taken in consideration while returning the tax code. It will have value when REVERSE_TAX_IND is Y.
predecessorTaxRulesId	No	Number (15)	Holds the predecessor rule ID that the result of the tax calculated in the predecessor rule will be added to the tax basis of the current rule in order to address Tax over Tax calculation requirement.
regionExplodeType	No	String (6)	Tax over tax rate calculated based in predecessor tax rule when applied.
gtsRulesConfigTITbl	No	Collection of Object	Translation records associated with tax record.
gtsRulesConditionsTbl	No	Collection of Object	Conditions associated with tax record.
gtsRulesCfaTbl	No	Collection of Object	Rules restrictions associated with tax record.

Table 5-209 GtsRulesConfigTITbl - This table receive information regarding translations that may be necessary.

Parameter Name	Required	Data Type	Description
ruleName	No	String (300)	Translated Rule.
lang	No	Number (6)	Lang.

Table 5-210 GtsRulesConditionsTbl - This table receives the conditions that will be used in the processing.

Parameter Name	Required	Data Type	Description
conditionType	No	String (6)	Type of condition used. Valid values are '1' for All Departments,'2' for Department,'3' for Class,'4' for Subclass,'5' for Item,'6' for Parent/Diff,'7' for Item List and '8' for Upload List.
dept	No	Number (4)	Item Department.
class	No	Number (4)	Item Class.
subclass	No	Number (4)	Item subclass.
itemId	No	String (25)	Item ID.
itemList	No	Number (8)	This field contains Item List number that was used to create this item record.
excludeInd	No	String (1)	Indicates if the condition in exclude or include. Valid values are Y or N.
diffId	No	String (10)	Diff ID used when condition type is Parent/DIFF.

Table 5-211 GtsRulesCfaTbl - This table (CFA) receives and process the restrictions associated with the rules.

Element Name	Required	Data Type	Description
attributeId	No	Number (10)	This column holds the attribute id that this extended data is associated with.
attributeValue	No	String (250)	This column holds the CFA value selected for the rule.
restrictionLevel	No	String (6)	Restriction Level. Valid values are M for Merchandise Hierarchy, D for Destination Region and S for Source Region.

JSON Structure:

```
{
  "ruleId": 919875,
  "ruleName": "Tax Rules US-CA",
  "taxRegionSource": 1000,
  "taxRegionDestination": 200,
  "activeDate": "2022-09-01",
  "endDate": "2022-12-31",
  "status": "S",
  "taxCode": "S",
  "taxCalcType": "P",
  "taxRate": 35.00,
  "taxBasisType": "B",
  "taxBasisFactor": 11.00,
```

```
"nonMerchCode": "",
"uomType": "",
"uomValue": 68.00,
"currencyCode": "",
"reverseTaxInd": "",
"reverseTaxThreshold": 19.00,
"predecessorTaxRulesId": 514301,
"regionExplodeType": "A",
"gtsRulesConfigTlTbl": [
  {
    "ruleName": "Tax Rules TL US-CA",
    "lang": 2
  }
],
"gtsRulesConditionsTbl": [
  {
    "conditionType": "1"
  }
],
"gtsRulesCfaTbl": [
  {
    "attributeId": 4,
    "attributeValue": "A",
    "restrictionLevel": "S"
  }
]
}
```

Output

Element Name	Required	Data Type	Description
status	No	String (255)	Process Execution status.E – Execution finished with errors. S – Execution completed successfully.
errMsg	No	String (4000)	Error message, in case of error.
errTbl	No	Collection of Object	Error table in case of existing errors.

Table 5-212 ErrTbl - Table used to detail errors associated with execution.

Element Name	Required	Data Type	Description
columnName	No	String (255)	Column name associated with error message.
errorMsg	No	String (4000)	Error msg.

JSON Structure:

```
{
  "status": "E",
  "errMsg": ""
  "errTbl": [
    {
```

```
        "columnName": "STATUS",
        "errorMsg": "Invalid Status.", "
    }
  ]
}
```

Response Code: 400 (Error)
In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_GTS_RULES_CONFIG	Yes	Yes	No	No
SVC_GTS_RULES_CONFIG_TL	Yes	Yes	No	No
SVC_GTS_RULES_CFA	Yes	Yes	No	No
SVC_GTS_RULES_CONDITIONS	Yes	Yes	No	No
GTS_RULES_CONFIG	Yes	No	No	No
GTS_RULES_CONFIG_TL	Yes	No	No	No
GTS_RULES_CFA	Yes	No	No	No
GTS_RULES_CONDITIONS	Yes	No	No	No

Update Tax Rule

Business Overview
Update an existing Tax Rule.

Service Type
PUT

ReST URL

/foundation/taxRules/update

Input

Parameter Name	Required	Data Type	Description
ruleId	No	Number (15)	Tax rule unique ID.
ruleName	No	String (300)	Tax Rule Name.
taxRegionSource	No	Number (4)	Source Tax Region.
taxRegionDestination	No	Number (4)	Destination Tax Region.
activeDate	No	date	Effective date when this tax rule become active.
endDate	No	date	End date when this tax rule expired.
status	No	String (6)	Status of the Rule. Valid Values are 'AC' for Active, 'AP' for Approved, 'O' Obsolete, 'S' for Submitted, 'U' for Updated, 'W' for Worksheet.
taxCode	No	String (6)	Tax code.
taxCalcType	No	String (6)	Define the tax calculation type. Possible values are 'P' for Percentage or 'U' for Unit'.
taxRate	No	Number (20,10)	Tax rate in percentage when tax calculation type is in percentage.
taxBasisType	No	String (6)	Indicate which value is used as base calculation to apply tax. Valid values are: 'C' for UNIT_COST, 'R' for UNIT_RETAIL, or 'B' for BOTH, 'N' ON_MERCH_COST.
taxBasisFactor	No	Number (20,10)	Contains a unique user specified code representing the non-merchandise code.
nonMerchCode	No	String (6)	Contains a unique user specified code representing the non-merchandise code.
uomType	No	String (4)	UOM type selected to be considered when tax calculation is in unit.
uomValue	No	Number (20,10)	UOM value selected to be considered when tax calculation is in unit.
currencyCode	No	String (3)	Contains the currency code for the tax rule. For unit tax application.
reverseTaxInd	No	String (1)	Reverse Tax Indicator: Valid values are 'Y' for Enable or 'N' for Disable.
reverseTaxThreshold	No	Number (20,10)	Threshold amount to be taken in consideration while returning the tax code. It will have value when REVERSE_TAX_IND is Y.
predecessorTaxRulesId	No	Number (15)	Holds the predecessor rule ID that the result of the tax calculated in the predecessor rule will be added to the tax basis of the current rule in order to address Tax over Tax calculation requirement.
regionExplodeType	No	String (6)	Tax over tax rate calculated based in predecessor tax rule when applied.

Parameter Name	Required	Data Type	Description
gtsRulesConfigTITbl	No	Collection of Object	Translation records associated with tax record.
gtsRulesConditionsTbl	No	Collection of Object	Conditions associated with tax record.
gtsRulesCfaTbl	No	Collection of Object	Rules restrictions associated with tax record.

Table 5-213 GtsRulesConfigTITbl - This table receive information regarding translations that may be necessary.

Element Name	Required	Data Type	Description
ruleName	No	String (300)	Translated Rule.
lang	No	Number (6)	Lang.

Table 5-214 GtsRulesConditionsTbl - This table receives the conditions that will be used in the processing.

Element Name	Required	Data Type	Description
conditionType	No	String (6)	Type of condition used. Valid values are '1' for All Departments,'2' for Department,'3' for Class,'4' for Subclass,'5' for Item,'6' for Parent/Diff,'7' for Item List and '8' for Upload List.
dept	No	Number (4)	Item Department.
class	No	Number (4)	Item Class.
subclass	No	Number (4)	Item subclass.
itemId	No	String (25)	Item ID.
itemList	No	Number (8)	This field contains Item List number that was used to create this item record.
excludeInd	No	String (1)	Indicates if the condition in exclude or include. Valid values are Y or N.
diffId	No	String (10)	Diff ID used when condition type is Parent/DIFF.

Table 5-215 GtsRulesCfaTbl - This table (CFA) receives and process the restrictions associated with the rules.

Element Name	Required	Data Type	Description
attributeId	No	Number (10)	This column holds the attribute id that this extended data is associated with.
attributeValue	No	String (250)	This column holds the CFA value selected for the rule.

Table 5-215 (Cont.) GtsRulesCfaTbl - This table (CFA) receives and process the restrictions associated with the rules.

Element Name	Required	Data Type	Description
restrictionLevel	No	String (6)	Restriction Level. Valid values are M for Merchandise Hierarchy, D for Destination Region and S for Source Region.

JSON Structure:

```
{
  "ruleId": 919875,
  "ruleName": "Tax Rules US-CA",
  "taxRegionSource": 1000,
  "taxRegionDestination": 200,
  "activeDate": "2022-09-01",
  "endDate": "2022-12-31",
  "status": "W",
  "taxCode": "S",
  "taxCalcType": "P",
  "taxRate": 35.00,
  "taxBasisType": "B",
  "taxBasisFactor": 11.00,
  "nonMerchCode": "",
  "uomType": "",
  "uomValue": 68.00,
  "currencyCode": "",
  "reverseTaxInd": "",
  "reverseTaxThreshold": 19.00,
  "predecessorTaxRulesId": 514301,
  "regionExplodeType": "A",
  "gtsRulesConfigTlTbl": [
    {
      "ruleName": "Tax Rules TL US-CA",
      "lang": 2
    }
  ],
  "gtsRulesConditionsTbl": [
    {
      "conditionType": "1"
    }
  ],
  "gtsRulesCfaTbl": [
    {
      "attributeId": 4,
      "attributeValue": "A",
      "restrictionLevel": "S"
    }
  ]
}
```

Output

Element Name	Required	Data Type	Description
status	No	String (255)	Process Execution status.E – Execution finished with errors. S – Execution completed successfully.
errMsg	No	String (4000)	Error message, in case of error.
errTbl	No	Collection of Object	Error table in case of existing errors.

Table 5-216 ErrTbl - Table used to detail errors associated with execution.

Element Name	Required	Data Type	Description
columnName	No	String (255)	Column name associated with error message.
errorMsg	No	String (4000)	Error msg.

JSON Structure:

```
{
  "status": "E",
  "errMsg": " CANNOT_UPD_RULE ",
  "errTbl": [
    {
      "columnName": "",
      "errorMsg": " The rule was not submitted successfully."
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

JSON Structure:

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_GTS_RULES_CONFIG	Yes	Yes	No	No
SVC_GTS_RULES_CONFIG_TL	Yes	Yes	No	No
SVC_GTS_RULES_CFA	Yes	Yes	No	No
SVC_GTS_RULES_CONDITIONS	Yes	Yes	No	No
GTS_RULES_CONFIG	Yes	No	Yes	No
GTS_RULES_CONFIG_TL	Yes	No	Yes	No
GTS_RULES_CFA	Yes	No	Yes	No
GTS_RULES_CONDITIONS	Yes	No	Yes	No

Delete Tax Rule

Business Overview

Delete an existing Tax Rule

Service Type

DELETE

ReST URL

/foundation/taxRules/delete

Input

Parameter Name	Required	Data Type	Description
ruleId	No	Number (15)	Tax rule unique ID.
ruleName	No	String (300)	Tax Rule Name.
taxRegionSource	No	Number (4)	Source Tax Region.
taxRegionDestination	No	Number (4)	Destination Tax Region.
activeDate	No	date	Effective date when this tax rule become active.
endDate	No	date	End date when this tax rule expired.
status	No	String (6)	Status of the Rule. Valid Values are 'AC' for Active, 'AP' for Approved, 'O' Obsolete, 'S' for Submitted, 'U' for Updated, 'W' for Worksheet.
taxCode	No	String (6)	Tax code.
taxCalcType	No	String (6)	Define the tax calculation type. Possible values are 'P' for Percentage or 'U' for Unit'.
taxRate	No	Number (20,10)	Tax rate in percentage when tax calculation type is in percentage.
taxBasisType	No	String (6)	Indicate which value is used as base calculation to apply tax. Valid values are: 'C' for UNIT_COST, 'R' for UNIT_R'ETAIL, or 'B' for BOTH, 'N'ON_MERCH_COST.

Parameter Name	Required	Data Type	Description
taxBasisFactor	No	Number (20,10)	Contains a unique user specified code representing the non-merchandise code.
nonMerchCode	No	String (6)	Contains a unique user specified code representing the non-merchandise code.
uomType	No	String (4)	UOM type selected to be considered when tax calculation is in unit.
uomValue	No	Number (20,10)	UOM value selected to be considered when tax calculation is in unit.
currencyCode	No	String (3)	Contains the currency code for the tax rule. For unit tax application.
reverseTaxInd	No	String (1)	Reverse Tax Indicator: Valid values are 'Y' for Enable or 'N' for Disable.
reverseTaxThreshold	No	Number (20,10)	Threshold amount to be taken in consideration while returning the tax code. It will have value when REVERSE_TAX_IND is Y.
predecessorTaxRulesId	No	Number (15)	Holds the predecessor rule ID that the result of the tax calculated in the predecessor rule will be added to the tax basis of the current rule to address Tax over Tax calculation requirement.
regionExplodeType	No	String (6)	Tax over tax rate calculated based in predecessor tax rule when applied.
gtsRulesConfigTITbl	No	Collection of Object	Translation records associated with tax record.
gtsRulesConditionsTbl	No	Collection of Object	Conditions associated with tax record.
gtsRulesCfaTbl	No	Collection of Object	Rules restrictions associated with tax record.

Table 5-217 GtsRulesConfigTITbl - This table receive information regarding translations that may be necessary.

Element Name	Required	Data Type	Description
ruleName	No	String (300)	Translated Rule.
lang	No	Number (6)	Lang.

Table 5-218 GtsRulesConditionsTbl - This table receives the conditions that will be used in the processing.

Element Name	Required	Data Type	Description
conditionType	No	String (6)	Type of condition used. Valid values are '1' for All Departments,'2' for Department,'3' for Class,'4' for Subclass,'5' for Item,'6' for Parent/Diff,'7' for Item List and '8' for Upload List.
dept	No	Number (4)	Item Department.
class	No	Number (4)	Item Class.
subclass	No	Number (4)	Item subclass.
itemId	No	String (25)	Item ID.
itemList	No	Number (8)	This field contains Item List number that was used to create this item record.
excludeInd	No	String (1)	Indicates if the condition in exclude or include. Valid values are Y or N.
diffId	No	String (10)	Diff ID used when condition type is Parent/DIFF.

Table 5-219 GtsRulesCfaTbl - This table (CFA) receives and process the restrictions associated with the rules.

Element Name	Required	Data Type	Description
attributeId	No	Number (10)	This column holds the attribute id that this extended data is associated with.
attributeValue	No	String (250)	This column holds the CFA value selected for the rule.
restrictionLevel	No	String (6)	Restriction Level. Valid values are M for Merchandise Hierarchy, D for Destination Region and S for Source Region.

JSON Structure:

```
{
  "ruleId": 919875,
  "ruleName": "Tax Rules US-CA",
  "taxRegionSource": 1000,
  "taxRegionDestination": 200,
  "activeDate": "2022-09-01",
  "endDate": "2022-12-31",
  "status": "W",
  "taxCode": "S",
  "taxCalcType": "P",
  "taxRate": 35.00,
  "taxBasisType": "B",
  "taxBasisFactor": 11.00,
```

```

"nonMerchCode": "",
"uomType": "",
"uomValue": 68.00,
"currencyCode": "",
"reverseTaxInd": "",
"reverseTaxThreshold": 19.00,
"predecessorTaxRulesId": 514301,
"regionExplodeType": "A",
"gtsRulesConfigTlTbl": [
  {
    "ruleName": "Tax Rules TL US-CA",
    "lang": 2
  }
],
"gtsRulesConditionsTbl": [
  {
    "conditionType": "1"
  }
],
"gtsRulesCfaTbl": [
  {
    "attributeId": 4,
    "attributeValue": "A",
    "restrictionLevel": "S"
  }
]
}

```

Output

Element Name	Required	Data Type	Description
status	No	String (255)	Process Execution status.E – Execution finished with errors. S – Execution completed successfully.
errMsg	No	String (4000)	Error message, in case of error.
errTbl	No	Collection of Object	Error table in case of existing errors.

Table 5-220 ErrTbl - Table used to detail errors associated with execution.

Element Name	Required	Data Type	Description
columnName	No	String (255)	Column name associated with error message.
errorMsg	No	String (4000)	Error msg.

JSON Structure:

```

{
  "status": "E",
  "errMsg": " CANNOT_UPD_RULE ",
  "errTbl": [
    {

```

```

        "columnName": "",
        "errorMsg": " The rule was not submitted successfully."
    }
  ]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_GTS_RULES_CONFIG	Yes	Yes	No	No
SVC_GTS_RULES_CONFIG_TL	Yes	Yes	No	No
SVC_GTS_RULES_CFA	Yes	Yes	No	No
SVC_GTS_RULES_CONDITIONS	Yes	Yes	No	No
GTS_RULES_CONFIG	Yes	No	No	Yes
GTS_RULES_CONFIG_TL	Yes	No	No	Yes
GTS_RULES_CFA	Yes	No	No	Yes
GTS_RULES_CONDITIONS	Yes	No	No	Yes

Franchise Management

The following services are included in this functional area:

- [Customer Credit Status Upload Service](#)
 - [Update Customer Credit Status](#)
- [Franchise Order Upload Service](#)
 - [Create Franchise Orders](#)
 - [Manage Franchise Orders](#)

Customer Credit Status Upload Service

The following services are included in this functional area:

- [Update Customer Credit Status](#)

Update Customer Credit Status

Functional Area

Franchise Management

Business Overview

This service provides a way for an external source, usually a financials system, to update the credit status for a franchise customer in Merchandising. This status is used when determining whether a franchisee order can be approved. Valid values are Y (credit is good) and N (credit issues). For each collection of customer and customer group passed into the service, the credit flag will be updated with the value indicated in the service call.

Merchandising returns failure status as part of the response object in the web service call if credit flag is not updated due to validation errors.

This service supports a collection of record to be submitted. The service either processes all the records or rejects all the record if there is a one or more bad record.

Service Type

PUT

ReST URL

MerchIntegrations/services/franchiseManagement/customerCreditStatus/update

Input Payload Details

Table 5-221 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection. This should match with the number of records in the input message.
items	No	Collection of Object	Collection of Customer records.

Table 5-222 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerId	Yes	Number (10)	The unique customer identifier.
customerGroupId	Yes	Number (10)	Customer Group to which the customer belongs to.

Table 5-222 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
creditInd	Yes	String (1)	Determine if the customer has good credit. valid values 'Y' and 'N'.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "customerId": 10061,
      "customerGroupId": 1006,
      "creditInd": "Y"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Franchise Order Upload Service

The following services are included in this functional area:

- [Create Franchise Orders](#)
- [Manage Franchise Orders](#)

Create Franchise Orders

Functional Area

Franchise Management

Business Overview

Deprecated in v24.1.301.0: Scheduled for Removal in 2 Years.

Use `MerchIntegrations/services/franchiseManagement/franchiseOrder/manage`

This service is used from an external source, usually an order management system, to create franchise orders in Merchandising. This service accepts a collection of franchise order and will return success or failure through the service response object. The franchise order uploaded through this service will be created with an order type of 'EDI'. A linked transfer, PO or store order will be created for the approved franchise orders.

Franchise orders created via this service will be systematically approved if the customer is setup for auto approval, provided the customer has valid credit. If the order does not meet the criteria for auto-approval, Merchandising will create the franchise order in Input or Require Credit Approval and will provide the reason of approval failure in the response object. Franchise orders from customers that are not identified for 'Auto Approval' are uploaded into Merchandising in Input status. Such orders will need to be manually approved in Merchandising to be considered active.

The service allows upload of one or more franchise orders in a single service call. Each request is treated as a single unit of work and if there are no validation errors or business validation errors, all the franchise orders in the message will be created and the service will return success through the service response object. In case of one or more validation errors, the service call will be rejected, and the error response will contain the details of the validation errors.

Item Validation

- Packs are allowed for warehouse sourced or supplier sourced franchise orders only. If the pack added is a buyer pack with an 'order as' type of eaches, then the pack is exploded to its components.
- Deposit container items cannot be added to franchise orders. When the deposit content item is added to an order, the associated container item gets added as well. Deposit item content and container items must have the same costing location.
- Transformable orderable items can be added to an order, however, transformable sellable items are not allowed, as these are not inventoried.
- There cannot be multiple detail record for the same source location/item/customer location combination within a franchise order.
- Multiple order detail lines with the same source location, item and customer location combination in the franchise order is not allowed.

Source Validations

- For supplier sourced orders, the supplier must be an active supplier and allows direct store deliveries.
- For supplier sourced franchise order, items must be 'Active' at the franchise store in order to be included on the order.
- The supplier must belong to the same org unit as the costing location.
- For warehouse or company store sourced order, the items must not be in 'Delete' status at the source location and must be in either 'Active' or 'Discontinued' status at the franchise store.
- If the source for franchise order is not specified, then the costing location for the item and franchise store will be used as the source location.
- If the source location is a physical warehouse, then the service will use distribution rules to determine from which virtual warehouse the inventory will be pulled, similar to distribution rules used for non-franchise transfers.

Customer Location Validation

- If item/franchise store relationship does not exist, the franchise store would be auto ranged as part of franchise order creation.
- For source location as a store, the customer store cannot be non-stockholding.
- When a franchise order is uploaded through this service, having a future cost record for the item/franchise store combination is not necessary. Whether or not the cost template relationship is defined for the item/franchise store, the order can still be approved based on the provided fixed cost. However, if the fixed cost is also not provided, then the order creation would fail.

Inventory Validations

The inventory validation for Franchise orders sourced from company locations is subject to the setting of the 'Validate Availability for External Franchise Orders' system option. If set to No, Franchise Order creation would be carried out without checking for inventory availability. If set to Yes, then the requested quantity is validated against the available inventory at location for the warehouse or store sourced franchise orders. For the warehouse sourced orders, if the requested quantity exceeds the available stock at the warehouse, then Merchandising checks whether the item is on the Store Order replenishment at the same sourcing warehouse for the franchise store. If so, then an order is placed for the sourcing warehouse from the supplier for the unavailable quantity and the sourcing for the franchise location is handled via transfers created through Merchandising replenishment process. When the source location is a store, requested inventory must be available at the location.

Order Date Validations

- Multiple franchise orders with the same source location, item, customer location and need date combination are not allowed in Merchandising.
- For the supplier sourced franchise orders, the not after date must be greater than supplier lead time.
- For the warehouse sourced franchise orders, if the franchise order has a need date that is less than the order lead days in the future, the franchise order is fulfilled with the available warehouse inventory. Any remaining is fulfilled via Store Order replenishment if the item is on Store Order replenishment at the same sourcing warehouse for the franchise store.
- For store sourced franchise orders, the need date must be within order lead days.

Item Validation

- Packs are allowed for warehouse sourced or supplier sourced franchise orders only. If the pack added is a buyer pack with an 'order as' type of eaches, then the pack is exploded to its components.
- Deposit container items cannot be added to franchise orders. When the deposit content item is added to an order, the associated container item gets added as well. Deposit item content and container items must have the same costing location.
- Transformable orderable items can be added to an order, however, transformable sellable items are not allowed, as these are not inventoried.
- There cannot be multiple detail record for the same source location/item/customer location combination within a franchise order.
- Multiple order detail lines with the same source location, item and customer location combination in the franchise order is not allowed.

Source Validations

- For supplier sourced orders, the supplier must be an active supplier and allows direct store deliveries.
- For supplier sourced franchise order, items must be 'Active' at the franchise store in order to be included on the order.
- The supplier must belong to the same org unit as the costing location.
- For warehouse or company store sourced order, the items must not be in 'Delete' status at the source location and must be in either 'Active' or 'Discontinued' status at the franchise store.
- If the source for franchise order is not specified, then the costing location for the item and franchise store will be used as the source location.
- If the source location is a physical warehouse, then the service will use distribution rules to determine from which virtual warehouse the inventory will be pulled, similar to distribution rules used for non-franchise transfers.

Customer Location Validation

- If item/franchise store relationship does not exist, the franchise store would be auto ranged as part of franchise order creation.
- For source location as a store, the customer store cannot be non-stockholding.
- When a franchise order is uploaded through this service, having a future cost record for the item/franchise store combination is not necessary. Whether or not the cost template relationship is defined for the item/franchise store, the order can still be approved based on the provided fixed cost. However, if the fixed cost is also not provided, then the order creation would fail.

Inventory Validations

The inventory validation for Franchise orders sourced from company locations is subject to the setting of the 'Validate Availability for External Franchise Orders' system option. If set to No, Franchise Order creation would be carried out without checking for inventory availability. If set to Yes, then the requested quantity is validated against the available inventory at location for the warehouse or store sourced franchise orders. For the warehouse sourced orders, if the requested quantity exceeds the available stock at the warehouse, then Merchandising checks whether the item is on the Store Order replenishment at the same sourcing warehouse for the franchise store. If so, then an order is placed for the sourcing warehouse from the supplier for the unavailable quantity and the sourcing for the franchise location is handled via transfers created through Merchandising replenishment process. When the source location is a store, requested inventory must be available at the location.

Order Date Validations

- Multiple franchise orders with the same source location, item, customer location and need date combination are not allowed in Merchandising.
- For the supplier sourced franchise orders, the not after date must be greater than supplier lead time.
- For the warehouse sourced franchise orders, if the franchise order has a need date that is less than the order lead days in the future, the franchise order is fulfilled with the available warehouse inventory. Any remaining is fulfilled via Store Order replenishment if the item is on Store Order replenishment at the same sourcing warehouse for the franchise store.
- For store sourced franchise orders, the need date must be within order lead days.

Service Type

POST

ReST URL

MerchIntegrations/services/franchiseManagement/franchiseOrder/create

Input Payload Details

Table 5-223 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of franchise order creation details.

Table 5-224 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerId	Yes	Number (10)	This value should be a valid customer id.
orderReferenceNo	Yes	String (20)	This is an external reference number that would be provided by the franchisee for their tracking purposes.
currencyCode	Yes	String (3)	This field represents the currency of the order, which may or may not be different from the primary currency in the system. Valid values for this field are based on the currency codes held in the CURRENCIES table.
exchangeRate	No	Number (20,10)	This field represents the exchange rate between the primary currency and the franchise order currency. If this is not provided, it is defaulted based on the conversion type set at system level.
freight	No	Number (20,4)	This field represents any freight charges associated to the franchise order.

Table 5-224 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otherCharges	No	Number (20,4)	This field represents other miscellaneous charges associated to the franchise order.
defaultBillingLocation	No	Number (10)	A customer's location where the billing for the entire order is sent. If blank, each location is billed.
billToAddressType	No	String (2)	This field represents the address type for the default billing location. This field is defaulted to Invoice address.
comments	No	String (2000)	Free form comments associated with the franchise order.
details	Yes	Collection of Object	Details of the orders. At least one detail item is mandatory.

Table 5-225 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This is the item which is on the Franchise Order. This should be an approved, inventory, orderable and transaction level item. The item should be ranged to both customer location and source location and must not be a consignment/concession item.
customerLocation	Yes	Number (10)	This field holds a valid franchise location number.
sourceLocationType	No	String (2)	Contains the source entity type from where the items will originate. Valid values are ST - Store, WH - Warehouse, SU - Supplier. If this field is populated, the source location should also be populated.
sourceLocation	No	Number (10)	Contains the location id from where this item will be sourced from. The id will correspond to a company store, warehouse or supplier depending on the source type value.
requestedQuantity	Yes	Number (12,4)	This field represents the quantity of the item on this record being ordered. This value is always written in the standard UOM for the item.
unitOfPurchase	No	String (3)	This field contains the unit of purchase information. Must be the standard unit of measure or a valid pallet name/case name/inner name for the item/supplier.

Table 5-225 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
needDate	Yes	date	This date represents the initial date by which the franchisee wants the item on this order. The date format should be YYYY-MM-DD. The need date can be business date (VDATE) or a date in future.
notAfterDate	Yes	date	This date represents the last date by which the franchisee will accept the item on this order. The date format should be YYYY-MM-DD. This date should be after need date.
fixedCost	No	Number (20,4)	This is a user defined cost which will override the customer cost for the item on this order if populated. This should be a positive numeric value. This must be provided if there are no cost templates associated with the item.

Table 5-226 CreateError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerId	Yes	Number (10)	Input Customer Id
orderReferenceNo	Yes	String (20)	Input order reference number
item	No	String (25)	Input item
customerLocation	No	Number (10)	Input franchise location number
sourceLocationType	No	String (2)	Input source location type
sourceLocation	Yes	Number (10)	Input source location id
errors	Yes	Array of String	List of errors identified during business data processing of the request

Sample Input Message

```
{
  "items": [
    {
      "customerId": 1001,
      "orderReferenceNo": "1001-A",
      "currencyCode": "USD",
      "exchangeRate": 52.5,
      "freight": 23.5,
      "otherCharges": 2.58,
      "defaultBillingLocation": 100123,
      "billToAddressType": "01",
      "comments": "Franchise order 1001-A",
      "details": [
        {
```

```

        "item": "104300083",
        "customerLocation": 100123,
        "sourceLocationType": "ST",
        "sourceLocation": 909090,
        "requestedQuantity": 9000,
        "unitOfPurchase": "EA",
        "needDate": "2001-12-31",
        "notAfterDate": "2001-12-31",
        "fixedCost": 95
      }
    ]
  }
}

```

Response Code: 200 (Success)

Table 5-227 CreateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	References a collection of franchise orders created.

Table 5-228 CreateResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
franchiseOrder	Yes	Number (10)	This is the unique identifier of the franchise order that is generated by Merchandising.
customerId	Yes	Number (10)	Input Customer Id
orderReferenceNo	Yes	String (20)	Input order reference number
status	Yes	String (1)	This contains the status in which the Franchise order was created. This can be I - Input or A - Approved.
autoApproveErrors	No	Array of String	This will be populated with the reason (like customer fails credit check) why the order could not be approved and ended up getting created in Input status. Using the UI, the franchise order should be approved after fixing the issue.
transactions	No	Collection of Object	References a collection of transfers or purchase orders created.

Table 5-229 Transactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerLocation	Yes	Number (10)	This field holds a valid franchise location number.

Table 5-229 (Cont.) Transactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sourceLocation	Yes	Number (10)	Contains the location id from where this item will be sourced from. The id will correspond to a company store, warehouse or supplier depending on the source type value.
sourceLocationType	Yes	String (2)	Contains the source entity type from where the items will originate. Valid values are ST - Store, WH - Warehouse, SU - Supplier. If this field is populated, the source location should also be populated.
needDate	Yes	date	This date represents the initial date by which the franchisee wants the item on this order. The date format should be YYYY-MM-DD. The need date can be business date (VDATE) or a date in future.
documentNo	Yes	Number (12)	This contains the purchase order or transfer number created.
documentType	Yes	String (1)	This contains the document type generated. This can be P - Purchase Order or T - Transfer.

Sample Response Message

```
{
  "items": [
    {
      "franchiseOrder": 30001,
      "customerId": 1001,
      "orderReferenceNo": "1001-A",
      "status": "A",
      "autoApproveErrors": [
        "null"
      ],
      "transactions": [
        {
          "customerLocation": 100123,
          "sourceLocation": 909090,
          "sourceLocationType": "ST",
          "needDate": "2001-12-31",
          "documentNo": 22145453,
          "documentType": "T"
        }
      ]
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-230 CreateError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerId	Yes	Number (10)	Input Customer Id
orderReferenceNo	Yes	String (20)	Input order reference number
item	No	String (25)	Input item
customerLocation	No	Number (10)	Input franchise location number
sourceLocationType	No	String (2)	Input source location type
sourceLocation	Yes	Number (10)	Input source location id
errors	Yes	Array of String	List of errors identified during business data processing of the request

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "customerId": 1001,
      "orderReferenceNo": "1001-A",
      "item": "104300083",
      "customerLocation": 100123,
      "sourceLocationType": "ST",
      "sourceLocation": 909090,
      "errors": [
        "Customer Location passed in franchise order is not a valid franchise
store."
      ]
    }
  ]
}
```

Manage Franchise Orders

Functional Area

Franchise Management

Business Overview

This service is used from an external source, usually an order management system, to create and maintain (that is, update, delete and cancel) franchise orders in Merchandising. The service allows upload of one or more franchise orders in a single service call. Each request is treated as a single unit of work and if there are no validation errors or business validation errors, all the franchise orders in the message will be created/updated/deleted and the service will return success through the service response object. In case of one or more validation errors, the service call will be rejected, and the error response will contain the details of the validation errors.

Order Creation

Franchise orders created through this service will have an order type of EDI. A linked transfer, PO or store order will be created for the approved franchise orders.

Franchise orders created through this service will be systematically approved if the customer is set up for auto-approval, provided the customer has valid credit. If the order does not meet the criteria for auto-approval, Merchandising will create the franchise order in Input or Require Credit Approval and will provide the reason of approval failure in the response object. Franchise orders from customers that are not set up for auto-approval are uploaded into Merchandising in Input status. Such orders will need to be manually approved in Merchandising to be considered active.

Item Validation

- Packs are allowed for warehouse sourced or supplier sourced franchise orders only. If the pack added is a buyer pack with an order as type of eaches, then the pack is exploded to its components.
- Deposit container items cannot be added to franchise orders. When the deposit content item is added to an order, the associated container item gets added as well. Deposit item content and container items must have the same costing location.
- Transformable orderable items can be added to an order, however, transformable sellable items are not allowed, as these are not inventoried.
- There cannot be multiple detail record for the same source location/item/customer location combination within a franchise order.
- Multiple order detail lines with the same source location, item and customer location combination in the franchise order is not allowed.

Source Validation

- For supplier sourced orders, the supplier must be an active supplier and allows direct store deliveries.
- For supplier sourced franchise order, items must be Active at the franchise store in order to be included on the order.
- The supplier must belong to the same org unit as the costing location.
- For warehouse or company store sourced order, the items must not be in Delete status at the source location and must be in either Active or Discontinued status at the franchise store.

- If the source for franchise order is not specified, then the costing location for the item and franchise store will be used as the source location.
- If the source location is a physical warehouse, then the service will use distribution rules to determine from which virtual warehouse the inventory will be pulled, similar to distribution rules used for non-franchise transfers.

Customer Location Validation

- If item/franchise store relationship does not exist, the franchise store would be auto ranged as part of franchise order creation.
- For source location as a store, the customer store cannot be non-stockholding.
- When a franchise order is uploaded through this service, having a future cost record for the item/franchise store combination is not necessary. Whether or not the cost template relationship is defined for the item/franchise store, the order can still be approved based on the provided fixed cost. However, if the fixed cost is also not provided, then the order creation would fail.

Inventory Validations

The inventory validation for Franchise orders sourced from company locations is subject to the setting of the **Validate Availability for External Franchise Orders** system option. If set to **No**, Franchise Order creation would be carried out without checking for inventory availability. If set to **Yes**, then the requested quantity is validated against the available inventory at location for the warehouse or store sourced franchise orders. For the warehouse sourced orders, if the requested quantity exceeds the available stock at the warehouse, then Merchandising checks whether the item is on the Store Order replenishment at the same sourcing warehouse for the franchise store. If so, then an order is placed for the sourcing warehouse from the supplier for the unavailable quantity and the sourcing for the franchise location is handled through transfers created through the Merchandising replenishment process. When the source location is a store, requested inventory must be available at the location.

Order Date Validations

- Multiple franchise orders with the same source location, item, customer location and need date combination are not allowed in Merchandising.
- For the supplier sourced franchise orders, the not after date must be greater than supplier lead time.
- For the warehouse sourced franchise orders, if the franchise order has a need date that is less than the order lead days in the future, the franchise order is fulfilled with the available warehouse inventory. Any remaining is fulfilled through Store Order replenishment if the item is on Store Order replenishment at the same sourcing warehouse for the franchise store.
- For store sourced franchise orders, the need date must be within order lead days.

Order Updates

For franchise orders Input status, the Update action will support the updating of a franchise order header and the adding, updating, or deleting of a franchise order detail. Header level updates includes changes such as customer order reference no, freight, and other charges as well as requested quantity, fixed cost, need date, and not after date at the detail level. Besides this, billing address details such as default bill to location, and bill to address can be updated. A new item can be added, and an existing item can be deleted using a Update header level action and a detail level Create or Delete level action respectively.

For an Approved franchise orders, only the requested quantity and fixed cost can be updated via this service. When the requested quantity is decremented, a line level cancellation reason must be provided.

Order Deletes and Cancellations

A franchise order can be deleted by sending an Update action with the status of the order set to Delete (D). Only orders in Input status can be deleted in Merchandising. If the user sends an Update action, with the status Delete (D) for a franchise order that is not in Input status, it will be treated as a cancellation and the quantity on all the order lines will be set to zero. A warning will be logged for the process to indicate to the user that the order was cancelled as opposed to being deleted.

Line level cancellations can be performed on approved franchise orders by using a header level action of Update and an order detail level action of Update with either the cancelled quantity field populated with the quantity to be cancelled; or with a header level action of Update and an order detail level action of Delete for the desired line, in which case the entire open quantity on the line will be cancelled. As part of cancellation, franchise orders in Approved or In progress status will get cancelled by providing the appropriate cancellation reason. Cancellation reasons are configured under the Franchise Order Cancel Reasons (WFC0) code type.

Service Type

PUT

ReST URL

MerchIntegrations/services/franchiseManagement/franchiseOrder/manage

Input Payload Details

Table 5-231 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	
items	Yes	Collection of Object	

Table 5-232 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Contains the intended action for the franchise order. Valid values are CREATE, UPDATE and DELETE.
franchiseOrder	No	Number (15)	Contains a unique identifier for the franchise order.
customerId	No	Number (10)	Contains the numeric identifier of the customer requesting the franchise order.
orderReferenceNo	No	String (20)	This field holds the ID for the franchise order used in the external system. This a reference number that would be provided by the franchisee which will help in tracking the franchise order created in Merchandising (RMS).
currencyCode	No	String (3)	Contains the currency of the franchise order.

Table 5-232 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
exchangeRate	No	Number (20,10)	Contains the exchange rate associated with the franchise order.
freight	No	Number (20,4)	Contains the freight charge associated with the franchise order.
otherCharges	No	Number (20,4)	Contains other miscellaneous charges associated with the franchise order.
billToAddressType	No	String (2)	Contains the address type of the default billing address of the billing location.
defaultBillingLocation	No	Number (10)	Contains the default billing location of the order. It will hold the customer's location where the billing for the entire order is sent.
billToIndividualshipmentLocation	No	String (1)	This field indicates if the franchise order is created for billing to individual shipment location.
orderCancelReason	No	String (6)	This field indicates the reason for order cancellation.
comments	No	String (2000)	Contains the comments associated with the franchise order.
status	No	String (1)	Contains the status of the franchise order.
details	No	Collection of Object	Details of the orders. At least one detail item is mandatory for header action CREATE otherwise ignored.

Table 5-233 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Contains the intended action for the franchise order. Valid values are CREATE, UPDATE and DELETE.
item	Yes	String (25)	Contains the item on the franchise order.
customerLocation	Yes	Number (10)	Contains the franchise store requesting the item.
sourceLocationType	Yes	String (2)	Contains the source entity type from where the items will originate. Valid values are ST - Store, WH - Warehouse, SU - Supplier.
sourceLocation	Yes	Number (10)	Contains the location Id from where this item will be sourced from.
requestedQuantity	No	Number (12,4)	Contains the quantity of the item on this record being ordered. The value is in the mentioned UOP.
unitOfPurchase	No	String (3)	Contains the unit of purchase of the requested quantity.

Table 5-233 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fixedCost	No	Number (20,4)	Contains the cost which will be charged to the customer for the item on the franchise order instead of using pricing cost.
needDate	No	date	Contains the initial date by which the item is needed in the franchise store.
notAfterDate	No	date	Contains the last date after which the item may no longer be accepted for a franchise store.
itemCancelReason	No	String (6)	This field represents the reason an item was cancelled from the franchise order. The cancellation of the item record results in the reduction of the overall order quantity.

Table 5-234 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerId	Yes	Number (10)	Input Customer Id
orderReferenceNo	Yes	String (20)	Input order reference number
item	No	String (25)	Input item
customerLocation	No	Number (10)	Input franchise location number
sourceLocationType	No	String (2)	Input source location type
sourceLocation	Yes	Number (10)	Input source location id
errors	Yes	Array of String	List of errors identified during business data processing of the request

Sample Input Message

```
{
  "collectionSize": 1000,
  "items": [
    {
      "action": "CREATE",
      "franchiseOrder": 584081,
      "customerId": 1001,
      "orderReferenceNo": "1001-A",
      "currencyCode": "USD",
      "exchangeRate": 52.5,
      "freight": 23.5,
      "otherCharges": 2.58,
      "billToAddressType": "01",
      "defaultBillingLocation": 100123,
      "billToIndividualshipmentLocation": "Y",
      "orderCancelReason": "NS",
      "comments": "Franchise order 1001-A",
      "status": "A",
      "details": [
```

```

    {
      "action": "CREATE",
      "item": "104300083",
      "customerLocation": 100123,
      "sourceLocationType": "ST",
      "sourceLocation": 909090,
      "requestedQuantity": 9000,
      "unitOfPurchase": "EA",
      "fixedCost": 95,
      "needDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "itemCancelReason": "ED"
    }
  ]
}

```

Response Code: 200 (Success)

Table 5-235 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
franchiseOrders	Yes	Collection of Object	

Table 5-236 FranchiseOrders - Object. See list of elements for detail

Element Name	Required	Data Type	Description
franchiseOrder	Yes	Number (10)	
customerId	Yes	Number (10)	Input Customer Id.
orderReferenceNo	Yes	String (20)	Input order reference number.
status	Yes	String (1)	This contains the status in which the Franchise order was created. This can be I - Input or A - Approved.
processedStatus	Yes	String (20)	Status of the request action.
autoApproveErrors	No	Array of String	This will be populated with the reason (like customer fails credit check) why the order could not be approved and ended up getting created in Input status. Using the UI, the franchise order should be approved after fixing the issue.
transactions	No	Collection of Object	References a collection of transfers or purchase orders created.

Table 5-237 Transactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customerLocation	Yes	Number (10)	This field holds a valid franchise location number.

Table 5-237 (Cont.) Transactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sourceLocation	Yes	Number (10)	Contains the location id from where this item will be sourced from. The id will correspond to a company store, warehouse or supplier depending on the source type value.
sourceLocationType	Yes	String (2)	Contains the source entity type from where the items will originate. Valid values are ST - Store, WH - Warehouse, SU - Supplier. If this field is populated, the source location should also be populated.
needDate	Yes	date	This date represents the initial date by which the franchisee wants the item on this order. The date format should be YYYY-MM-DD. The need date can be business date (VDATE) or a date in future.
documentNo	Yes	Number (12)	This contains the purchase order or transfer number created.
documentType	Yes	String (1)	This contains the document type generated. This can be P - Purchase Order or T - Transfer.

Sample Response Message

```
{
  "franchiseOrders": [
    {
      "franchiseOrder": 584081,
      "customerId": 1001,
      "orderReferenceNo": "1001-A",
      "status": "A",
      "processedStatus": "CREATED",
      "autoApproveErrors": [
        "null"
      ],
    },
    "transactions": [
      {
        "customerLocation": 100123,
        "sourceLocation": 909090,
        "sourceLocationType": "ST",
        "needDate": "2001-12-31",
        "documentNo": 22145453,
        "documentType": "T"
      }
    ]
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service. The element `businessError` will be present if the payload passes schema validation but an exception is caught while processing the business logic.

Table 5-238 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
<code>customerId</code>	Yes	Number (10)	Input Customer Id
<code>orderReferenceNo</code>	Yes	String (20)	Input order reference number
<code>item</code>	No	String (25)	Input item
<code>customerLocation</code>	No	Number (10)	Input franchise location number
<code>sourceLocationType</code>	No	String (2)	Input source location type
<code>sourceLocation</code>	Yes	Number (10)	Input source location id
<code>errors</code>	Yes	Array of String	List of errors identified during business data processing of the request

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "customerId": 1001,
      "orderReferenceNo": "1001-A",
      "item": "104300083",
      "customerLocation": 100123,
      "sourceLocationType": "ST",
      "sourceLocation": 909090,
      "errors": [
        "Customer Location passed in franchise order is not a valid franchise
store."
      ]
    }
  ]
}
```

Import Management

The following services are included in this functional area:

- [Actual Landed Cost](#)
 - [Actual Landed Cost Upload Service](#)
 - * [Finalize Actual Landed Cost](#)
- [Obligations](#)
 - [Obligations Upload Services](#)

Actual Landed Cost

The following services are included in this functional area:

- [Actual Landed Cost Upload Service](#)
 - [Finalize Actual Landed Cost](#)

Actual Landed Cost Upload Service

The following services are included in this functional area:

- [Finalize Actual Landed Cost](#)

Finalize Actual Landed Cost

Functional Area

Import Management - Actual Landed Cost

Business Overview

This service is used to finalize the Actual Landed Cost (ALC) for a list of purchase orders or purchase order/shipment combinations matching the provided collection of purchase order, ASN, obligation, and/or customs entry input records. The available inputs for this service are purchase order, ASN, obligation, customs entry, method of finalization and the Back Out Estimates with No Actuals flag. Success or failure of the operation will be returned as part of the response object. The value of the Allocation Shipment Level Obligations Method system option will determine if orders or order/shipment combinations are processed as follows:

- When the Allocate Shipment Level Obligations Using system option is set to ASN (ASN), the service processes order/shipment combinations matching the input provided.
- When the Allocate Shipment Level Obligations Using system option is set to Transportation (TRNSPRT), the service processes purchase orders matching the input provided; the service will not accept ASN as an input in this case.

This service compares the actual value from obligations or customs entries with the estimated value on the purchase order and posts records to transaction data (transaction code 20A), update weighted average cost (WAC) and open to buy (OTB) with the difference between the estimate and the actual.

For the pending ALC to be finalized, the associated obligation or customs entry must be approved or confirmed respectively. Further, when Allocate Shipment Level Obligations Using is set to Transportation, orders cannot be finalized until the purchase order has been closed. When the Allocate Shipment Level Obligations Using is set to ASN, order/shipments must be received before the ALC can be finalized, and for ALC from Purchase Order Header or Purchase Order/Item level obligations, the associated purchase order must be closed before the associated ALC can be finalized.

When the provided method of finalization is No Finalization, the pending ALC records are updated to have a status of No Finalization, which means that the actual values are ignored and the estimated values that were posted at the time of receipt will be retained. Subsequent

finalizations of the same purchase orders or purchase order/shipments will only attempt to finalize new ALC records in Pending status; the records previously marked as No Finalization or Processed will not be processed during future finalizations.

Service Type

PUT

ReST URL

MerchIntegrations/services/importManagement/actualLandedCost/finalize

Input Payload Details

Table 5-239 Finalize - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	Collection of the input details for which the ALC Finalization needs to be done.

Table 5-240 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	No	Number (12)	The unique identifier for the Purchase Order.
asn	No	String (30)	The ASN linked to the Shipment.
obligation	No	Number (10)	The unique identifier for the Obligation.
customsEntryNo	No	String (40)	The Customs Entry Number.
methodOfFinalization	Yes	String (6)	The method of ALC Finalization to be used.
backOutEstimatesWithNoActualsInd	No	String (1)	Indicates whether the Order Estimates, which are not having Actuals, need to be Backed out or not.

Table 5-241 FinalizeError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
alcFinalizeError	Yes	Collection of Object	

Table 5-242 AlcFinalizeError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	
asn	Yes	String (30)	
obligation	Yes	Number (10)	
customsEntryNo	Yes	String (40)	

Table 5-242 (Cont.) AlcFinalizeError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
errorMessages	Yes	Array of String	List of errors identified during business data processing of the request.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 100000,
      "asn": "ASN_1234",
      "obligation": 56789,
      "customsEntryNo": "CE_1234",
      "methodOfFinalization": "W",
      "backOutEstimatesWithNoActualsInd": "N"
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-243 FinalizeResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	

Table 5-244 FinalizeResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	
asn	Yes	String (30)	
obligation	Yes	Number (10)	
customsEntryNo	Yes	String (40)	
alcStatus	Yes	String (250)	The status of the ALC records, after the successful completion of the ALC Finalization.

Sample Response Message

```
{
  "items": [
    {
      "orderNo": 120711,
      "asn": "String",
      "obligation": 389535,
```

```

        "customsEntryNo": "String",
        "alcStatus": "String"
    }
]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` is present when the input payload or input parameters do not match the schema definition of this service. The element `businessError` is present if the payload passes schema validation but an exception is caught while processing the business logic.

Table 5-245 FinalizeError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
alcFinalizeError	Yes	Collection of Object	

Table 5-246 AlcFinalizeError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	
asn	Yes	String (30)	
obligation	Yes	Number (10)	
customsEntryNo	Yes	String (40)	
errorMessages	Yes	Array of String	List of errors identified during business data processing of the request.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "alcFinalizeError": [
        {
          "orderNo": 120711,
          "asn": "String",
          "obligation": 389535,
          "customsEntryNo": "String",
          "errorMessages": [
            "String"
          ]
        }
      ]
    }
  ]
}

```

```
}  
  }  
    ]  
  }  
    ]  
  }
```

Obligations

The following services are included in this functional area:

- [Obligations Upload Services](#)

Obligations Upload Services

This section describes the Obligations Upload Services.

Function Area

Import Management - Obligations

Business Overview

The primary role of these services is to support requests from trading partners or suppliers for bulk uploads of obligations in Merchandising.

Create Obligation

Business Overview

This service creates obligations by calling the SVCPROV_OBLIGATION package, then calling the core obligation package to validate and insert data to the Merchandising tables.

This service creates obligations with required fields. At least one component is needed to create a successful obligation. If the obligation level is Purchase Order Header (P0) or Purchase Order/Items (P0IT) then component locations can be added if desired. Allocation to Actual Landed Cost (ALC) will be performed after obligation creation.

When creating a new Purchase Order Header (P0) or Purchase Order/Item (P0IT) level obligation with a provided status of Approved, if the purchase order has a single location or if the component/location level details are provided on the obligation, you are able to specify whether the corresponding non-merchandise invoice should be generated and sent to Invoice Matching (REIMCS) upon obligation approval, or if the system should wait until the purchase order is closed. To have the system generate the non-merchandise invoice upon obligation approval, the Create Non-Merchandise Invoice upon Obligation Approval flag (createInvoiceOnObligationApproval) should be passed in as Yes (Y); otherwise you can send a value of No (N). If no value is provided for the flag, the system will assume a value of No (N). If the purchase order has multiple locations, the non-merchandise invoice will not be generated or sent to Invoice Matching until the purchase order is closed. The reason for waiting until closure in this case is because this allows the system to know the final receipt quantities in order to determine the location allocation of the expenses on the obligation.

When creating new obligations of any level besides Purchase Order Header or Purchase Order/Item, non-merchandise invoices will be generated and sent to Invoice Matching (REIMCS) upon obligation approval.

Purchase Order and Purchase Order/Item level obligations are intended to be used for a full purchase order, meaning the system expects that a given expense component will only appear

on one obligation for a given purchase order or for a given purchase order/item. For example, if you are being charged a freight expense for each of two shipments for a single purchase order. To use a Purchase Order Header level obligation, you would need to total the freight expense for the two shipments and enter the total value on a single Purchase Order Header level obligation with the total quantity on the order. You cannot create two separate Purchase Order Header level obligations for the same Freight expense component for a partial quantity. If you want to create separate obligations for a subset of the total ordered quantity, you need to use one of the transportation or ASN level obligations.

Service Type

Post

ReST URL

/ObligationUpload/createObligation

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compId	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal
amt	BigDecimal

JSON Structure

invoiceDate and paidDate will take input in strings the 'YYYY-MM-DD' format that will be converted to Timestamp format and inserted into tables.

```
[{
  "obligationKey": null,
  "obligationLevel": null,
  "keyValue1": null,
  "keyValue2": null,
  "keyValue3": null,
  "keyValue4": null,
  "keyValue5": null,
  "keyValue6": null,
  "status": null,
  "partnerType": null,
  "partnerId": null,
  "supplierSite": null,
  "invoice": null,
  "invoiceDate": null,
  "qty": null,
  "qtyUom": null,
  "exchangeRate": null,
  "currency": null,
  "paymentMethod": null,
  "checkAuthNo": null,
  "paidAmt": null,
  "paidDate": null,
  "comments": null,
  "obligationCompTbl" : [{
    "compId": null,
    "allocateToAlc": null,
```

```

        "allocationType":null,
        "allocationBasisUom": null,
        "amt":null,
        "rate":null,
        "perCount":null,
        "perCountUom":null,
        "obligationCompLocTbl":[{"action":null,
        "locType":null,
        "locId": null,
        "qty":null,
        "amt":null}]
    }
}

```

Output**RestObligationStatusRecRDO**

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request including validation error, if any.

The service will insert data into the Obligation (obligation), Obligation Component (obligation_comp) and Obligation Component Location (obligation_comp_loc) tables based on obligation levels. Obligation amounts will be allocated to each order/item/location represented on the obligation and will be recorded in the Actual Landed Cost (ALC) tables. Non-merchandise invoices generated for approved obligations will be inserted into invoice staging tables to be loaded into Invoice Matching.

For Success:

```

{
  "successObligationTbl": [

```

```

    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

For Failure:

```

{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
      "errorMessage": null,
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	No	Yes	No	No
OBLIGATION_COMP	No	Yes	No	No
OBLIGATION_COMP_LOC	No	Yes	No	No
ORDHEAD	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC	Yes	No	No	No
TRANSPORTATION	Yes	No	No	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
CE_HEAD	Yes	No	No	No
ELC_COMP	Yes	No	No	No
RTM_UNIT_OPTIONS	Yes	No	No	No
ALC_HEAD	Yes	Yes	Yes	No
ALC_COMP_LOC	Yes	Yes	Yes	No
INVC_HEAD	No	Yes	No	No
INVC_NON_MERCH	No	Yes	No	No
INVC_XREF	No	Yes	No	No

Create Obligation Component

Business Overview

This service creates obligation components for existing obligations in Pending status by calling the SVCPROV_OBLIGATION package, then calling the core obligation package to validate and insert data to the Merchandising tables.

This service accepts an obligation key, a supplier/invoice combination, or a partner type/partner/invoice combination to identify the existing obligation. Expense component details can be added.. Allocation to Actual Landed Cost (ALC) will be done after component creation.

Service Type

Post

ReST URL

/ObligationUpload/createObligationComp

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
Status	String
partnerType	String
partnerId	String

Parameter Name	Data Type
supplierSite	String
invoice	String
invoiceDate	Timestamp
Qty	BigDecimal
qtyUom	String
Currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
Comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compld	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
Amt	BigDecimal
Rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
Action	String
locType	String
locId	BigDecimal
Qty	BigDecimal
Amt	BigDecimal

JSON Structure: The same RDO 'RestObligationRecRDO' will be used for createObligationComp, but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

```
[{
  "obligationKey": null,
  "partnerType": null,
  "partnerId": null,
  "supplierSite": null,
  "invoice": null,
```

```

    "obligationCompTbl" : [
      {
        "compId":null,
        "allocateToAlc":null,
        "allocationType":null,
        "allocationBasisUom": null,
        "amt":null,
        "rate":null,
        "perCount":null,
        "perCountUom":null,
        "obligationCompLocTbl":[
          {
            "action":null,
            "locType":null,
            "locId": null,
            "qty":null,
            "amt":null}}
        ]
      }
    ]
  }
}

```

Output**RestObligationStatusRecRDO**

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request including validation error, if any.

The service will insert data into the Obligation Component (obligation_comp) table, and for Purchase Order Header or Purchase Order/Item obligations with location details, into the Obligation Component Location (obligation_comp_loc) table. Obligation amounts will be allocated to each order/item/location represented on the obligation and will be recorded in the Actual Landed Cost (ALC) tables.

For Success:

```
{
  "successObligationTbl": [
    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

For Failure:

```
{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
      "errorMessage": null,
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	No
OBLIGATION_COMP	Yes	Yes	No	No
OBLIGATION_COMP_LOC	Yes	Yes	No	No
ORDHEAD	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC	Yes	No	No	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
V_PACKSKU_QTY	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
ELC_COMP	Yes	No	No	No
ALC_HEAD	Yes	Yes	Yes	No
ALC_COMP_LOC	Yes	Yes	Yes	No

Create Obligation Component Location

Business Overview

This service creates obligation component locations for Purchase Order Header or Purchase Order/Item level obligations by calling the SVCPROV_OBLIGATION package and then calling the core obligation package to validate and insert data into the Merchandising tables.

For Purchase Order Header (P0) and Purchase Order/Item (P0IT) obligation levels, a location-level breakdown of the obligation component amounts can be added. This service accepts an obligation key, supplier/invoice or partner type/partner/invoice combination to identify the existing obligation. Allocation to Actual Landed Cost (ALC) will be done after successful location creation.

Service Type

Post

ReST URL

/ObligationUpload/createObligationCompLoc

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String

Parameter Name	Data Type
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compId	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal
amt	BigDecimal

JSON Structure: Same RDO 'RestObligationRecRDO' will be used for createObligationComp, but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

```
[{
  "obligationKey": null,
  "partnerType": null,
  "partnerId": null,
  "supplierSite": null,
  "invoice": null,
  "obligationCompTbl" : [{
    "compId": null,
```

```

        "obligationCompLocTbl": [{
            "action": null,
            "locType": null,
            "locId": null,
            "qty": null,
            "amt": null}]
    }]
}

```

Output

RestObligationStatusRecRDO

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request, including validation errors, if any.

For Success:

```

{
  "successObligationTbl": [
    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
}

```

```

    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

For Failure:

{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
      "errorMessage": null,
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	No
OBLIGATION_COMP	Yes	No	No	No
OBLIGATION_COMP_LOC	Yes	Yes	No	No
ORDHEAD	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC	Yes	No	No	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
ALC_HEAD	Yes	Yes	Yes	No
ALC_COMP_LOC	Yes	Yes	Yes	No

Modify Obligation

Business Overview

This service updates obligation header record by calling the SVCPR0V_OBLIGATION package and then calling the core obligation package to validate and update data to the Obligation table.

This service accepts an obligation key, a supplier/invoice combination or a partner type/partner/invoice combination to identify the existing obligation.

When approving an existing Purchase Order Header (PO) or Purchase Order/Item (POIT) level obligation, if purchase order has a single location or if component/location level details are provided on the obligation, you are able to specify whether the corresponding non-merchandise invoice should be generated and sent to Invoice Matching (REIMCS) upon obligation approval, or if the system should wait until the purchase order is closed. To have the system generate the non-merchandise invoice upon obligation approval, the Create Non-Merchandise Invoice upon Obligation Approval flag (`createInvoiceOnObligationApproval`) should be passed in as Yes (Y), otherwise you can send a value of No (N). If no value is provided for the flag, the system will assume a value of No (N). If the purchase order has multiple locations, the non-merchandise invoice will not be generated or sent to Invoice Matching until the purchase order is closed. The reason for waiting until closure in this case is because this allows the system to know the final receipt quantities in order to determine the location allocation of the expenses on the obligation.

When approving obligations of any level besides Purchase Order Header or Purchase Order/Item, non-merchandise invoices will be generated and sent to Invoice Matching (REIMCS) upon obligation approval.

When modifying and/or approving a pending Purchase Order Header (PO) or Purchase Order/Item (POIT) level obligation, if purchase order has a single location or if component/location level details are provided on the obligation, you are able to specify whether the corresponding non-merchandise invoice should be generated and sent to Invoice Matching (REIMCS) upon obligation approval, or if the system should wait until the purchase order is closed. To have the system generate the non-merchandise invoice upon obligation approval, the Create Non-Merchandise Invoice upon Obligation Approval flag (`createInvoiceOnObligationApproval`) should be passed in as Yes (Y), otherwise you can send a value of No (N). If no value is provided for the flag, the system will assume a value of No (N). If the purchase order has multiple locations, the non-merchandise invoice will not be generated or sent to Invoice Matching until the purchase order is closed. The reason for waiting until closure in this case is because this allows the system to know the final receipt quantities in order to determine the location allocation of the expenses on the obligation.

When approving existing obligations of any level besides Purchase Order Header or Purchase Order/Item, non-merchandise invoices will be generated and sent to Invoice Matching (REIMCS) upon obligation approval.

The following fields can be updated when the obligation is in Pending status.

- Status
- Partner Type
- Partner ID
- Supplier Site
- Invoice
- Invoice Date
- Quantity
- Quantity UOM
- Currency
- Exchange Rate

The below fields, which are not mandatory, can be updated in pending as well as approved status.

- Payment Method
- Check Authorization No.
- Amount Paid
- Paid Date
- Comments

Service Type

Post

ReST URL

/ObligationUpload/modifyObligation

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compId	String
allocateToAlc	String

Parameter Name	Data Type
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal
amt	BigDecimal

JSON Structure: Same RDO 'RestObligationRecRDO' will be used for modifyObligation but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

invoiceDate and paidDate will take input in a string with the 'YYYY-MM-DD' format, which will be converted to Timestamp format and insert into the tables.

```
[{
  "obligationKey": null,
  "status": null,
  "partnerType": null,
  "partnerId": null,
  "supplierSite": null,
  "invoice": null,
  "invoiceDate": null,
  "qty": null,
  "qtyUom": null,
  "exchangeRate": null,
  "currency": null,
  "paymentMethod": null,
  "checkAuthNo": null,
  "paidAmt": null,
  "paidDate": null,
  "comments": null
}]
```

Output

RestObligationStatusRecRDO

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal

Parameter Name	Data Type
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request, including validation errors, if any.

NULL values will be ignored for mandatory fields. Non-mandatory fields can be updated to NULL.

For Success:

```
{
  "successObligationTbl": [
    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

For Failure:

```
{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
```

```

        "errorMessage": null,
        "status": null,
        "invoice": null,
        "obligationKey": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    },
    "failObligationCount": null,
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	No
OBLIGATION_COMP	Yes	No	No	No
OBLIGATION_COMP_LOC	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC	Yes	No	No	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
ALC_HEAD	Yes	Yes	Yes	No
ALC_COMP_LOC	Yes	Yes	Yes	No
INVC_HEAD	No	Yes	No	No
INVC_NON_MERCH	No	Yes	No	No
INVC_XREF	No	Yes	No	No

Modify Obligation Comp

Business Overview

This service updates existing obligation component records which are in pending status by calling the SVCPROV_OBLIGATION package and then calling the core obligation package to validate and update data in the obligation_comp table.

This service accepts an obligation key, a supplier/invoice combination, or a partner type/partner/invoice combination to identify the existing obligation. The user can modify component level fields on pending obligations. Reallocation to ALC will be done after update.

Service Type

Post

ReST URL

/ObligationUpload/modifyObligationComp

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compld	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String

Parameter Name	Data Type
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal
amt	BigDecimal

JSON Structure: The same RDO 'RestObligationRecRDO' will be used for modifyObligationComp, but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

```
[{
  "obligationKey": null,
  "partnerType":null,
  "partnerId":null,
  "supplierSite":null,
  "invoice":null,
  "obligationCompTbl" : [{"compId":null,
    "allocateToAlc":null,
    "allocationType":null,
    "allocationBasisUom": null,
    "amt":null,
    "rate":null,
    "perCount":null,
    "perCountUom":null,
  }
}]
```

Output

RestObligationStatusRecRDO

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request, including validation errors, if any.

For Success:

```
{
  "successObligationTbl": [
    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

For Failure:

```
{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
      "errorMessage": null,
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```


Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	No
OBLIGATION_COMP	Yes	No	No	No
OBLIGATION_COMP_LOC	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC	Yes	No	No	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
ALC_HEAD	Yes	Yes	Yes	No
ALC_COMP_LOC	Yes	Yes	Yes	No

Modify Obligation Component Location**Business Overview**

This service updates existing obligation component records which are in pending status by calling the SVCPROV_OBLIGATION package and then calling the core obligation package to validate and update data in the obligation_comp table.

This service accepts an obligation key, a supplier/invoice combination, or a partner type/partner/invoice combination to identify the obligation. Enter an expense component for which you want to modify the location. Reallocation to ALC will be done after update.

Service Type

Post

ReST URL

/ObligationUpload/modifyObligationCompLoc

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String

Parameter Name	Data Type
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compld	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal
amt	BigDecimal

JSON Structure: The same RDO 'RestObligationRecRDO' will be used for modifyObligationCompLoc but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

```
[{
  "obligationKey": null,
  "partnerType": null,
  "partnerId": null,
  "supplierSite": null,
  "invoice": null,
  "obligationCompTbl" : [{
    "compId": null,
    "obligationCompLocTbl": [{
      "action": null,
      "locType": null,
      "locId": null,
      "qty": null,
      "amt": null}]
  }]
}]
```

Output

RestObligationStatusRecRDO

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request, including validation errors, if any.

The action field is mandatory here. Valid value for this are MOD/DEL. MOD is to update a location and DEL to delete a location.

For Success:

```
{
  "successObligationTbl": [
    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

For Failure:

```
{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
      "errorMessage": null,
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	No
OBLIGATION_COMP	Yes	No	Yes	No
OBLIGATION_COMP_LOC	Yes	No	Yes	No
ORDHEAD	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC	Yes	No	No	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
ALC_HEAD	Yes	No	Yes	No
ALC_COMP_LOC	Yes	No	Yes	No

Delete Obligation

Business Overview

This service deletes existing obligation records with pending status by calling the SVCPROV_OBLIGATION package and then calling the core obligation package to validate and delete entire obligation records from Merchandising tables.

This service accepts an obligation key, a supplier/invoice combination, or a partner type/partner/invoice combination to identify the obligation. This service will delete the entire obligation record.

Service Type

Post

ReST URL

/ObligationUpload/deleteObligation

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String

Parameter Name	Data Type
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compld	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal
amt	BigDecimal

JSON Structure: The same RDO 'RestObligationRecRDO' will be used for deleteObligation but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

```
[{
  "obligationKey": null,
  "partnerType":null,
  "partnerId":null,
  "supplierSite":null,
  "invoice":null }]
```

Output

RestObligationStatusRecRDO

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>

Parameter Name	Data Type
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request, including validation errors, if any.

For Success:

```
{
  "successObligationTbl": [
    {
      "status": null,
      "invoice": null,
      "obligationKey": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "successObligationCount": null,
  "failObligationTbl": [],
  "failObligationCount": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

For Failure:

```
{
  "successObligationTbl": [],
  "successObligationCount": null,
  "failObligationTbl": [
    {
      "errorMessage": null,

```

```

        "status": null,
        "invoice": null,
        "obligationKey": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    },
    "failObligationCount": null,
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	Yes
OBLIGATION_COMP	Yes	No	No	Yes
OBLIGATION_COMP_LOC	Yes	No	No	Yes
ALC_HEAD	Yes	No	No	Yes
ALC_COMP_LOC	Yes	No	No	Yes

Delete Obligation Component

Business Overview

This service deletes obligation components for existing obligations by calling the SVCPROV_OBLIGATION package and then calling the core obligation package to validate and delete data from Merchandising tables.

This service accepts an obligation key, a supplier/invoice combination or a partner type/partner/invoice combination to identify the obligation. Pass the expense component to be deleted. This service will delete the component if the obligation is in Pending status and the component is not the only component on the obligation. If locations are attached to the components then that will be deleted as well. Reallocation to ALC will be done after deletion.

Service Type

Post

ReST URL

/ObligationUpload/deleteObligationComp

Input Parameters

RestObligationRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
obligationLevel	String
keyValue1	String

Parameter Name	Data Type
keyValue2	String
keyValue3	String
keyValue4	String
keyValue5	String
keyValue6	String
status	String
partnerType	String
partnerId	String
supplierSite	String
invoice	String
invoiceDate	Timestamp
qty	BigDecimal
qtyUom	String
currency	String
exchangeRate	BigDecimal
checkAuthNo	String
paidAmt	BigDecimal
paidDate	Timestamp
comments	String
obligationCompTbl	List<RestObligationCompRecRDO>

RestObligationCompRecRDO

Parameter Name	Data Type
compId	String
allocateToAlc	String
allocationType	String
allocationBasisUom	String
amt	BigDecimal
rate	BigDecimal
perCount	BigDecimal
perCountUom	String
obligationCompLocTbl	List<RestObligationCompLocRecRDO>

RestObligationCompLocRecRDO

Parameter Name	Data Type
action	String
locType	String
locId	BigDecimal
qty	BigDecimal

Parameter Name	Data Type
amt	BigDecimal

JSON Structure: The same RDO 'RestObligationRecRDO' will be used for deleteObligationComp, but only the below parameters will be considered for the request. The rest of the parameters will be ignored.

```
[{
  "obligationKey": null,
  "partnerType": null,
  "partnerId": null,
  "supplierSite": null,
  "invoice": null,
  "obligationCompTbl" : [{
    "compId": null
  }]
}]
```

Output

RestObligationStatusRecRDO

Parameter Name	Data Type
successObligationTbl	List<RestObligationSuccessRecRDO>
successObligationCount	BigDecimal
failObligationTbl	List<RestObligationFailRecRDO>
failObligationCount	BigDecimal

RestObligationSuccessRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String

RestObligationFailRecRDO

Parameter Name	Data Type
obligationKey	BigDecimal
status	String
invoice	String
errorMessage	String

The output will contain the status of the request, including validation errors, if any.

For Success:

```
{
  "successObligationTbl": [
    {
      "status": null,
```

```

        "invoice": null,
        "obligationKey": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    },
    ],
    "successObligationCount": null,
    "failObligationTbl": [],
    "failObligationCount": null,
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}

```

For Failure:

```

{
    "successObligationTbl": [],
    "successObligationCount": null,
    "failObligationTbl": [
        {
            "errorMessage": null,
            "status": null,
            "invoice": null,
            "obligationKey": null,
            "links": [],
            "hyperMediaContent": {
                "linkRDO": []
            }
        }
    ],
    "failObligationCount": null,
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
OBLIGATION	Yes	No	No	Yes
OBLIGATION_COMP	Yes	No	No	Yes
OBLIGATION_COMP_LOC	Yes	No	No	Yes
ORDHEAD	Yes	No	No	Yes
ORDSKU	Yes	No	No	Yes
ORDLOC	Yes	No	No	Yes
SHIPMENT	Yes	No	No	Yes
SHIPSKU	Yes	No	No	Yes
V_PACKSKU_QTY	Yes	No	No	Yes
ITEM_SUPP_COUNTRY	Yes	No	No	Yes
ORDLOC_EXP	Yes	No	No	Yes
ALC_HEAD	Yes	No	Yes	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
ALC_COMP_LOC	Yes	No	Yes	Yes

Inventory

The following services are included in this functional area:

- [Appointments](#)
 - [Appointment Upload Service](#)
 - * [Create Appointment](#)
 - * [Delete Appointment](#)
 - * [Delete Appointment Details](#)
 - * [Manage Appointment Details](#)
 - * [Update Appointment](#)
- [Inventory Download Service](#)
 - [Get Item Available Inventory at Customer Orderable Locations](#)
 - [Get Item Location Inventory](#)
- [Inventory Adjustments](#)
 - [Inventory Adjustment Upload Service](#)
 - * [Create Inventory Adjustment for Items at a Location](#)
 - * [Create Inventory Adjustments](#)
 - [Item Transformation Upload Service](#)
 - * [Create Item Transformations](#)
- [Returns To Vendor](#)
 - [Return to Vendor Download Service](#)
 - * [RTV Publish Services](#)
 - [Return to Vendor Upload Service](#)
 - * [Manage Returns to Vendor](#)
- [Shipments and Receipts](#)
 - [Receiver Unit Adjustment Download Service](#)
 - * [Get Receiver Unit Adjustment](#)
 - [Shipment and Receipt Download Service](#)
 - * [Get Shipment Detail](#)
 - [Shipment and Receipt Upload Service](#)
 - * [Create DSD Receipts](#)
 - * [Create Outbound ASNs](#)
 - * [Delete Inbound ASNs](#)
 - * [Manage Inbound ASN](#)

- * [Manage Purchase Order and Stock Order Receipts](#)
- * [Manage Purchase Order Receipts](#)
- * [Manage Stock Order Receipts](#)
- * [Update DSD Receipts](#)
- [Stock Counts](#)
 - [Stock Count Download Service](#)
 - * [Get Stock Count Detail](#)
 - [Stock Count Upload Service](#)
 - * [Create Stock Count Schedules](#)
 - * [Delete Stock Count Schedule Locations](#)
 - * [Delete Stock Count Schedules](#)
 - * [Update Stock Count Schedules](#)
- [Transfers and Allocations](#)
 - [Allocation Download Service](#)
 - * [Get Allocation Details for Provided Allocation](#)
 - * [Get Details for an Allocation](#)
 - [Allocation Number Download Service](#)
 - * [Get Next Allocation Numbers Service](#)
 - [Allocation Upload Service](#)
 - * [Close Allocations](#)
 - * [Create Allocation Details](#)
 - * [Create Allocations](#)
 - * [Delete Allocation Details](#)
 - * [Update Allocation Details](#)
 - * [Update Allocations](#)
 - [Mobile Workflow Transfer Download Service](#)
 - * [Create Transfer - Get From Locations for Transfer](#)
 - * [Create Transfer - Get Items for Transfer](#)
 - * [Create Transfer - Get Next Transfer Numbers](#)
 - * [Create Transfer - Get To Locations for Transfer](#)
 - * [Create Transfer - Refresh Transfer Items](#)
 - * [Create Transfer - Refresh Transfer Locations](#)
 - * [Mobile Workflow - Get Locations on Existing Transfers](#)
 - * [Mobile Workflow - Get Transfer Statuses](#)
 - * [Mobile Workflow - Get Transfer Summary](#)
 - * [Mobile Workflow - Get Transfer Types](#)
 - * [Mobile Workflow - Get Transfer Users](#)
 - * [Mobile Workflow - Get Transfers](#)

- * [Mobile Workflow - Refresh Transfer Locations](#)
- [Mobile Workflow Transfer Upload Service](#)
 - * [Create Transfer](#)
 - * [Mobile Workflow - Update Transfer Status](#)
- [Stock Order Status Upload Service](#)
 - * [Manage Stock Order Status](#)
- [Transfer Detail Service](#)
- [Transfer Download Service](#)
 - * [Get Transfer](#)
 - * [Get Transfer Details for Provided Transfer](#)
- [Transfer Upload Service](#)
 - * [Create Book Transfer](#)
 - * [Create Transfer Details](#)
 - * [Create Transfers](#)
 - * [Delete Transfer Details](#)
 - * [Delete Transfers](#)
 - * [Update Transfer Details](#)
 - * [Update Transfers](#)
- [Work Order Status Upload Service](#)
 - * [Update Work Orders Status](#)
- [Work Order Download Service](#)
 - * [Purchase Order Work Order Publish Services](#)
 - * [Transfer Work Order Publish Services](#)

Appointments

The following services are included in this functional area:

- [Appointment Upload Service](#)
 - [Create Appointment](#)
 - [Delete Appointment](#)
 - [Delete Appointment Details](#)
 - [Manage Appointment Details](#)
 - [Update Appointment](#)

Appointment Upload Service

The following services are included in this functional area:

- [Create Appointment](#)
- [Delete Appointment](#)
- [Delete Appointment Details](#)

- [Manage Appointment Details](#)
- [Update Appointment](#)

Create Appointment

Functional Area

Inventory - Appointments

Business Overview

This service can be used to create a new appointment in Merchandising. An appointment is sent to Merchandising when merchandise is about to arrive at a location. After performing the business level validations, appointments will be created through this message in the Scheduled status.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/appointments/create

Input Payload Details

Table 5-247 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of appointments.

Table 5-248 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field contains the location where the merchandise has been sent.
appointmentNo	Yes	Number (9)	This field contains the unique number generated by warehouse while creating an appointment.
appointmentDateTime	No	date-time	The date and time of the scheduled appointment, i.e. when the goods are expected to arrive at the receiving location.
action	Yes	String (2)	The status of the Appointment. Valid values include: SC - Scheduled, MS - Modified Scheduled, AR - Arrived, AC - Closed.
details	No	Collection of Object	References a collection of appointment details.
receiptDateTime	No	date-time	The actual date and time upon which the trailer arrived at the receiving location for the appointment.

Table 5-249 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains the items shipped to the location.
unitQuantity	Yes	Number (12,4)	This field contains the quantity of the item slated to be sent to the location.
documentNo	Yes	Number (12)	This field contains the purchase order, transfer or allocation corresponding to the shipped merchandise.
documentType	Yes	String (1)	This field indicates the type of document corresponding to the shipped merchandise. Possible choices are Purchase Order (P), Transfer (T), or Allocation (A). Types 'D' and 'V' are also valid document types that are treated the same as transfers.
asnNo	No	String (30)	This field contains the advance shipping notice number associated with the appointment. It is populated only when the appointment is based on an ASN.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "location": 2,
      "appointmentNo": 200911,
      "appointmentDateTime": "2001-12-31T23:59:59.123Z",
      "action": "SC",
      "details": [
        {
          "item": "101200001",
          "unitQuantity": 3,
          "documentNo": 100000135000,
          "documentType": "T",
          "asnNo": null
        }
      ],
      "receiptDateTime": "2001-12-31T23:59:59.123Z"
    }
  ]
}
```


Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Appointment

Functional Area

Inventory - Appointments

Business Overview

This service can be used to delete existing appointments in Merchandising. In addition, Merchandising attempts to close the document that is related to the appointment, when applicable. For more details on Appointments, see the Create Appointment service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/appointments/delete

Input Payload Details

Table 5-250 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of appointments.

Table 5-251 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentNo	Yes	Number (9)	This field contains the unique number generated by warehouse while creating an appointment.
location	Yes	Number (10)	This field contains the location where the merchandise has been sent.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "appointmentNo": 200911,
      "location": 2
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Appointment Details

Functional Area

Inventory - Appointments

Business Overview

This service is used to delete appointment details from Merchandising. To know more about appointment details, see the Manage Appointment Details service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/appointment/details/delete

Input Payload Details

Table 5-252 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of appointment details.

Table 5-253 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentNo	Yes	Number (9)	This field contains the unique number generated by warehouse while creating an appointment.
location	Yes	Number (10)	This field contains the location where the merchandise has been sent.
details	No	Collection of Object	References a collection of appointment details.

Table 5-254 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains the items shipped to the location.
documentNo	Yes	Number (12)	This field contains the purchase order, transfer or allocation corresponding to the shipped merchandise.
asnNo	No	String (30)	This field contains the advance shipping notice number associated with the appointment. It is populated only when the appointment is based on an ASN.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
```

```

    "appointmentNo": 200911,
    "location": 2,
    "details": [
      {
        "item": "101200001",
        "documentNo": 100000135000,
        "asnNo": null
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Manage Appointment Details

Functional Area

Inventory - Appointments

Business Overview

This service updates an appointment detail record that was previously sent, such as an update to the quantity for an item. If the record doesn't already exist, it is added.

An appointment is sent to Merchandising when merchandise is about to arrive at a location. These appointment messages are processed by Merchandising and it attempts to receive against the appointment and close it, wherever it is possible. This message contains the item information associated with an appointment, including the ASN and the document number (PO, transfer, or allocation)

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/appointment/details/manage

Input Payload Details

Table 5-255 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of appointments.

Table 5-256 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field contains the location where the merchandise has been sent.
appointmentNo	Yes	Number (9)	This field contains the unique number generated by warehouse while creating an appointment.
action	Yes	String (2)	The status of the Appointment. Valid values include: SC - Scheduled, MS - Modified Scheduled, AR - Arrived, AC - Closed.
details	No	Collection of Object	References a collection of appointment details.

Table 5-257 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains the items shipped to the location.
unitQuantity	Yes	Number (12,4)	This field contains the quantity of the item slated to be sent to the location.
documentNo	Yes	Number (12)	This field contains the purchase order, transfer or allocation corresponding to the shipped merchandise.
documentType	Yes	String (1)	This field indicates the type of document corresponding to the shipped merchandise. Possible choices are Purchase Order (P), Transfer (T), or Allocation (A). Types 'D' and 'V' are also valid document types that are treated the same as transfers.

Table 5-257 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
asnNo	No	String (30)	This field contains the advance shipping notice number associated with the appointment. It is populated only when the appointment is based on an ASN.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "location": 2,
      "appointmentNo": 200911,
      "action": "SC",
      "details": [
        {
          "item": "101200001",
          "unitQuantity": 12,
          "documentNo": 100000135000,
          "documentType": "T",
          "asnNo": null
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
    }
  ]
}
```

```
        "inputValue": "X"
      }
    ]
  }
```

Update Appointment

Functional Area

Inventory - Appointments

Business Overview

This service is used to edit an existing appointment in Merchandising. An update message updates the status of an existing appointment if already exists. Valid values for the status column include:

- SC–Scheduled
- MS–Modified Scheduled
- AR–Arrived
- AC–Closed

For more details about Appointments, see the Create Appointment service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/appointments/update

Input Payload Details

Table 5-258 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of appointments.

Table 5-259 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field contains the location where the merchandise has been sent.
appointmentNo	Yes	Number (9)	This field contains the unique number generated by warehouse while creating an appointment.
appointmentDateTime	No	date-time	The date and time of the scheduled appointment,i.e. when the goods are expected to arrive at the receiving location.

Table 5-259 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (2)	The status of the Appointment. Valid values include: SC - Scheduled, MS - Modified Scheduled, AR - Arrived, AC - Closed.
receiptDateTime	No	date-time	The actual date and time upon which the trailer arrived at the receiving location for the appointment.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "location": 2,
      "appointmentNo": 200911,
      "appointmentDateTime": "2001-12-31T23:59:59.123Z",
      "action": "SC",
      "receiptDateTime": "2001-12-31T23:59:59.123Z"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```


Inventory Download Service

The following services are included in this functional area:

- [Get Item Available Inventory at Customer Orderable Locations](#)
- [Get Item Location Inventory](#)

Get Item Available Inventory at Customer Orderable Locations

Functional Area

Inventory

Business Overview

This service publishes available inventory data to downstream consuming systems. It is intended for sellable inventory visibility and returns only rows that are currently sellable, belong to a stockholding and customer-orderable store or warehouse, and are either warehouse inventory or non-pack store inventory. The payload exposes available quantity and stock on hand for the qualifying item/location, and available quantity is calculated at read time from the cached inventory buckets, with warehouse calculations also including pack-component quantities.

This web service is used to fetch available inventory at customer orderable store or warehouse locations. The service considers only approved transaction level items which are sellable, inventoried and ranged to stockholding and customer orderable locations. To keep the data volume manageable for performance consideration, the service publishes inventory availability for item/locations which currently have non-zero inventory position, or had non-zero inventory position in the last 7 days when the current inventory position is zero.

Initial seeding is narrower than steady-state updates. Refresh seeds approved transaction-level inventoried items that changed recently or still carry non-zero inventory in tracked buckets, while delta updates are driven by ITEM_LOC_SOH quantity changes. Changes to store or warehouse customer-orderable or stockholding flags are handled through targeted rebuild requests so the published population stays aligned with location eligibility.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_LOC_SOH

Additional business tables: ITEM_MASTER, STORE, WH

JSON cache table: MERCHAPI_ITEM_LOC_SOH

JSON generation view: not applicable; cache rows are populated directly by package logic

The runtime response is read directly from that cache table, and the cache is populated from ITEM_LOC_SOH, ITEM_MASTER, STORE, and WH through the available-inventory cache processing flow.

Functionally, it serves sellable inventory visibility for customer-orderable locations and returns only rows that are currently sellable, belong to a stockholding and customer-orderable store or warehouse, and are either warehouse inventory or non-pack store inventory.

The payload exposes available quantity and stock on hand for the qualifying item/location, and available quantity is calculated at read time from the cached inventory buckets, with warehouse calculations also including pack-component quantities.

Initial Seeding and post enablement rebuild/repair

Refresh seeds approved transaction-level inventoried items where `ITEM_MASTER.STATUS = 'A'`, `ITEM_MASTER.ITEM_LEVEL = ITEM_MASTER.TRAN_LEVEL`, and `ITEM_MASTER.INVENTORY_IND = 'Y'`.

Refresh keeps rows changed in the last 7 days or rows that still have a non-zero quantity in one of the tracked inventory buckets, and `DATA_REFRESH/REBUILD` marks cache rows deleted when the item/location no longer exists in `ITEM_LOC_SOH`.

ICL (Integration Change Log) table: `ICL_RMS_ITEM_LOC_SOH` (ICL consumer name = `MERCHAPI`)

Triggers on `ITEM_LOC_SOH` stage `ICL_RMS_ITEM_LOC_SOH` updates for store and warehouse `ITEM_LOC_SOH` rows whenever tracked inventory buckets change.

Batch-driven bulk rebuild handling for specific events

Store stockholding indicator change, store customer-orderable indicator change, warehouse stockholding indicator change, and warehouse customer-orderable indicator change do not rely only on `ITEM_LOC_SOH` delta staging.

Those source events insert a record into `MERCHAPI_ASYNC_REQUEST` for inventory/available with `REQUEST_TYPE = DATA_REFRESH`, `REFRESH_TYPE = REBUILD`, and `REQUEST_PARAMETERS` set to the affected store or warehouse so the rebuild stays scoped to that location.

These requests are processed through `MERCHAPI_DATA_REBUILD_ADHOC_PROCESS` -> `MERCHAPI_DATA_REBUILD_ADHOC_JOB` and realign the `MERCHAPI_ITEM_LOC_SOH` cache when location eligibility changes can add or remove a large inventory population at once.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: `API_ITEM_LOC_SOH`

To enable via Background engine set `BACKGROUND_RUN_FLAG` to Y and review the `BACKGROUND_RUN_INTERVAL_SECOND` parameter.

POM Batch schedule process and job: `MERCHAPI_DELTA_ITEM_LOC_SOH_PROCESS` -> `MERCHAPI_ITEM_LOC_SOH_ADHOC_JOB`

Webhook configuration api name: `inventory/available`

Special handling

Additional GET query parameters: `locationType`, `location`.

Request windows greater than 30 days use the full-download access path, while shorter windows use the delta access path.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. Because this service is driven primarily by ITEM_LOC_SOH row volume and inventory-bucket churn rather than by complex source SQL, the available configuration should be validated against the customer production inventory profile, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through API_NO_OF_THREADS, and commit sizing is controlled through API_COMMIT_MAX_COUNTER.

Because the cache is populated directly from ITEM_LOC_SOH together with a small set of eligibility tables, runtime is usually driven more by row volume and inventory-change frequency than by join complexity. Common starting points for large-volume runs are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment.

API_COMMIT_MAX_COUNTER should be sized together with the thread setting. Typical values range from about 1,000 rows and can extend to 10,000 for simpler restartable workloads. Lower commit values may perform better when sustained inventory-change spikes need to drain in smaller units of work.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, DROP_INDEX_ON_REFRESH can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes. GATHER_STATS_ON_REFRESH should normally remain enabled unless testing shows a specific reason to change it.

REST fetch considerations

Because the payload is relatively narrow, larger page limits are often practical after testing. REST page size should still be selected so the average response remains below about 10 MB per call, using common page-limit values such as 1,000, 5,000, or 10,000 rows.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the LAST_UPDATE_DATETIME index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping BUILD_HISTOGRAM enabled allows the service to support recovery through smaller, more selective polling windows using the optimalBefore query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides optimalBefore and leaves before empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-

window delta pattern. Hourly histogram processing groups LAST_UPDATE_DATETIME values into buckets of 100,000 rows in MERCHAPI_TIMESTAMP_HISTOGRAM using API specific HISTOGRAM_ROW_COUNT configuration.

Handling Bulk Publish on Store/Warehouse attribute change

This API also uses batch-driven REBUILD handling when store or warehouse stockholding or customer-orderable attributes change.

Those source changes insert MERCHAPI_ASYNC_REQUEST REBUILD requests for inventory/available with the affected location in REQUEST_PARAMETERS. A single store or warehouse eligibility change can therefore republish the full qualifying inventory population for that location rather than only the base inventory rows that changed.

A bulk location stockholding or customer-orderable change can result in a very large number of inventory records being published to downstream systems and should be planned with the downstream performance implication of that message volume in mind.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/available

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
locationType	No	String	Allowed value for locationType is 'S' for Store or 'W' for Warehouse. If not defined, fetches details for all locations. If defined, fetches for the specified location type.
location	No	String (10)	Should be a valid store or a virtual warehouse if inventory position for a particular location has to be fetched. locationType is mandatory if this is populated
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-260 PageResultsItemsMerchapItemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-261 MerchapItemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	This field contains the action for available inventory.

Table 5-261 (Cont.) MerchapiltemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationType	No	String (1)	This field contains the type of location in the location field. This will be either S for store or W for warehouse.
location	No	Number(10,0)	This field contains the numeric identifier of the location in which the item is to be found.
item	No	String (25)	This field contains the unique alphanumeric value that identifies the item. This will be an approved sellable transaction-level item. For warehouse locations, pack items are also included.
availableQuantity	No	Number(12,4)	This field represents the available inventory of an item in a particular location. This is calculated as stock on hand minus the sum of non-sellable, transfer reserved, customer order reserved, RTV, and in-progress sales quantity. For warehouse locations, the inventory of an item as a component of a pack is also included in the available inventory calculation.
stockOnHand	No	Number(12,4)	This field contains the current stock on hand for the item.
standardUom	No	String (4)	This field contains the unit of measure in which stock of the item is tracked at a corporate level.
physicalWarehouse	No	Number(10,0)	This field contains the physical warehouse for a warehouse location.

Table 5-261 (Cont.) MerchapiltemLocSoh - Object. See list of elements for detail

Element Name	Required	Data Type	Description
channelId	No	Number(4,0)	This field contains the channel ID for the store, or the channel of the virtual warehouse when the location type is warehouse.
cacheTimestamp	No	dateTime	This field holds the database timestamp when the message was prepared or cached to serve this service for an item/location combination. The message serving this service is cached for an item/location when inventory position is changed, and this will continue to hold the same timestamp value until the next inventory change. The consumer can use this field for sequencing records for an item/location. UTC timestamp.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-262 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "locationType": null,
      "location": 6000,
      "item": "100100076",
      "availableQuantity": 3,
      "stockOnHand": 12,
      "standardUom": "EA",
      "physicalWarehouse": 11,
      "channelId": 1,
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	“self” : This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Example JSON Input

```
{
  "items": [{
    "action": "INSERT",
    "locationType": "W",
    "location": "13",
    "item": "102900026",
    "availableQuantity": 1,
    "stockOnHand": 1,
    "standardUom": "EA",
    "physicalWarehouse": 1,
    "channelId": 1,
    "cacheTimestamp": "2022-07-11T15:57:42.147Z"
  }],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [{
    "href": "http://<hostname>:<portnumber>/MerchIntegrations/services/
inventory/available?
locationType=W&limit=1&since=1970-01-01T00:00:00.001Z&before=2022-07-18T23:27:
26.547666+00:00",
    "rel": "self"
  },
  {
    "href": "http://<hostname>:<portnumber>/MerchIntegrations/services/
inventory/available?
locationType=W&offsetkey=W%2C13%2C102900026&limit=1&since=1970-01-01T00:00:00.
001Z&before=2022-07-18T23:27:26.547666+00:00",
    "rel": "next"
  }
]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
STORE	Yes	No	No	No
WH	Yes	No	No	No
MERCHAPI_ITEM_LOC_SOH	Yes	No	No	No

Get Item Location Inventory

This section describes the Item Loc Inventory Detail service.

Business Overview

Item Loc Inventory Detail service allows user to retrieve Item Location and Item Location Stock on Hand information for a selected item and location. If location and location type are not specified, all locations for the item will be retrieved. If location type is specified but not the location, all locations for the item and location type will be retrieved.

Service Type

Get

ReST URL

```
ItemlocInvDtl/itemlocInvDetail?  
item={itemNumber}&location={locationNumber}&locationType={locationType}
```

Input Parameters

Parameter Name	Required	Description
Item	Yes	Item ID
Location	No	Location ID.
Location Type	No	Location Type.

Output

RestItemlocInvDtlRecRDO

Parameter Name	Data Type
item	String
itemParent	String
loc	BigDecimal
locType	String
unitRetail	BigDecimal
sellingUom	String
clearInd	String
taxableInd	String
localItemDesc	String
status	String
primarySupp	BigDecimal
primaryCntry	String
avCost	BigDecimal
unitCost	BigDecimal
stockOnHand	BigDecimal
sohUpdateDatetime	Timestamp
inTransitQty	BigDecimal
packCompSoh	BigDecimal

Parameter Name	Data Type
packCompResv	BigDecimal
packCompExp	BigDecimal
rtvQty	BigDecimal
customerResv	BigDecimal
sellingUnitRetail	BigDecimal
localShortDesc	String
packCompIntran	BigDecimal
tsfReservedQty	BigDecimal
tsfExpectedQty	BigDecimal
nonSellableQty	BigDecimal
customerBackorder	BigDecimal
packCompCustResv	BigDecimal
packCompCustBack	BigDecimal
packCompNonSellable	BigDecimal
inProgressSalesQty	BigDecimal
firstReceived	Timestamp
lastReceived	Timestamp

JSON Structure

```
{
  "item": null,
  "itemParent": null,
  "loc": null,
  "locType": null,
  "unitRetail": null,
  "sellingUom": null,
  "clearInd": null,
  "taxableInd": null,
  "localItemDesc": null,
  "status": null,
  "primarySupp": null,
  "primaryCntry": null,
  "avCost": null,
  "unitCost": null,
  "stockOnHand": null,
  "sohUpdateDatetime": null,
  "inTransitQty": null,
  "packCompSoh": null,
  "packCompResv": null,
  "packCompExp": null,
  "rtvQty": null,
  "customerResv": null,
  "sellingUnitRetail": null,
  "localShortDesc": null,
  "packCompIntran": null,
  "tsfReservedQty": null,
  "tsfExpectedQty": null,
```

```

    "nonSellableQty": null,
    "customerBackorder": null,
    "packCompCustResv": null,
    "packCompCustBack": null,
    "packCompNonSellable": null,
    "inProgressSalesQty": null,
    "firstReceived": null,
    "lastReceived": null,
    "links": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Inventory Adjustments

The following services are included in this functional area:

- [Inventory Adjustment Upload Service](#)
 - [Create Inventory Adjustment for Items at a Location](#)
 - [Create Inventory Adjustments](#)
- [Item Transformation Upload Service](#)
 - [Create Item Transformations](#)

Inventory Adjustment Upload Service

The following services are included in this functional area:

- [Create Inventory Adjustment for Items at a Location](#)
- [Create Inventory Adjustments](#)

Create Inventory Adjustment for Items at a Location

This section describes the Inventory Adjustment service.

Functional Area

Inventory

Business Overview

Deprecated in v23.1.401.0: Scheduled for Removal in 2 Years. Use MerchIntegrations/services/inventory/inventoryAdjustments/create

The primary role of this service is to create inventory adjustment and send them to Merchandising.

Inventory Adjustment

Business Overview

Deprecated in v23.1.401.0: Scheduled for Removal in 2 Years. Use MerchIntegrations/services/inventory/inventoryAdjustments/create

This service creates inventory adjustment by calling the package SVCPROV_INVADJ to load input data to the staging tables and then calling the core inventory adjustment package to validate and insert data to the Merchandising tables.

Service Type

Post

ReST URL

Invadj/createInvadj

Input Parameters

SvcprovInvadjdescRecRDO

Parameter Name	Data Type
location	BigDecimal
Invadjdtl	List<SvcprovInvadjdescdtlRecRDO>

SvcprovInvadjdescdtlRecRDO

Parameter Name	Data Type
unitQty	BigDecimal
toDisposition	String
adjReasonCode	BigDecimal
docType	String
toWipCode	String
item	String
poNbr	String
auxReasonCode	String
weight	BigDecimal
toTroubleCode	String
fromWipCode	String
weightUom	String
unitCost	BigDecimal
fromTroubleCode	String
transshipmentNumber	String
fromDisposition	String
transactionCode	BigDecimal
adjTranDate	Date (Format: YYYY-MM-DD)
userId	String

JSON Structure

```
[
  {
    "location":null,
    "invadjdtl":[
      {
        "unitQty":null,
        "toDisposition":null,
        "adjReasonCode":null,
        "docType":null,
        "toWipCode":null,
        "item":null,
        "poNbr":null,
        "auxReasonCode":null,
        "weight":null,
        "toTroubleCode":null,
        "fromWipCode":null,
        "weightUom":null,
        "unitCost":null,
        "fromTroubleCode":null,
        "transshipmentNumber":null,
        "fromDisposition":null,
        "transactionCode":null,
        "adjTranDate":null,
        "userId":null,
      }
    ]
  }
]
```

Output

SvcprovInvadjStatusRecRDO

Parameter Name	Data Type
statusMsg	String
invadjErrTbl	List< SvcprovFailInvadjRecRDO >

SvcprovFailInvadjRecRDO

Parameter Name	Data Type
location	BigDecimal
unitQty	BigDecimal
adjReasonCode	BigDecimal
item	String
errorMsg	String

The output will contain the status of the request including validation errors, if any.

JSON Structure

```

{
  "statusMsg": null,
  " invadjErrTbl ": [
    {
      " location": null,
      "unitQty": null,
      " adjReasonCode": null,
      " item": null,
      "errorMsg": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
INV_ADJ	Yes	Yes	No	No
ITEM_LOC_SOH	Yes	Yes	Yes	No
SVC_PROCESS_TRACKER	Yes	Yes	Yes	No
SVC_INV_ADJ	Yes	Yes	Yes	Yes
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Inventory Adjustments**Functional Area**

Inventory - Inventory Adjustments

Business Overview

Merchandising receives requests for inventory adjustments from an integration subsystem through this service. The request contains information about the item and location whose inventory is being adjusted, the quantity to adjust, a 'from' and a 'to' disposition code, and the reason for the adjustment. Merchandising uses the information in these requests to perform the following actions :

Adjust overall quantities of stock on hand for the item and location combination

Adjust the availability of item-location quantities based on status.

After initial processing and validation, Merchandising performs the following tasks:

The item/location is ranged if it does not already exist.

For total stock on hand adjustments:

Stock on hand is updated for the item at the location, for total stock on hand adjustments.

Stock adjustment is recorded to the Merchandising transaction level stock ledger.

For status-based adjustments:

Quantities by inventory status are adjusted for the item/location combination.

Non-sellable quantity is updated for the item/location.

For both types, an audit trail is created for the inventory adjustment by item, location, inventory status and reason.

Note: An adjustment can impact both total stock on hand and inventory status at the same time.

Inventory Adjustment Transaction Codes

Whenever the status or quantity of inventory changes, Merchandising writes transaction codes to adjust inventory values in the stock ledger. The types of inventory adjustment transaction codes are:

- Tran code 22 - adjustments where positive and negative adjustments are made to total stock on hand using a reason code with the COGS indicator = N. In this case, a transaction is inserted to the transaction level stock ledger for both the retail and cost value of the adjustment.
- Tran code 23 - adjustments where positive and negative adjustments are made to total stock on hand using a reason code with the COGS indicator = Y. In this case, a transaction is inserted to the transaction level stock ledger for both the retail and cost value of the adjustment.
- Tran code 25 - adjustments to inventory status, where inventory is moved to or from an unavailable or non-sellable status.

Other Notes

One or both of the 'from' disposition and 'to' disposition fields must have values. Both cannot be empty. If both contain equal values, no adjustment will be made.

The item must be inventoried and approved.

If the item is a simple pack catch weight item, then both the weight and weight UOM need to be defined or NULL. The weight UOM must be of the type 'Mass'.

The item should be a transaction-level or a reference item. When a reference item is passed in, its parent item (the transaction level item) has its inventory adjusted.

If adjusting a pack at a warehouse, the pack item must have its inventory tracked at the pack level (receive as type = Pack for the item/warehouse).

If the location is a warehouse, then either a virtual or physical warehouse can be supported. If it is a virtual warehouse, it must be a stockholding warehouse. If it is a physical warehouse, then the adjusted quantity is distributed among the virtual locations of the physical location.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/inventoryAdjustments/create

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Details - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Item Transformation Upload Service

The following services are included in this functional area:

- [Create Item Transformations](#)

Create Item Transformations

Functional Area

Inventory - Inventory Adjustments

Business Overview

The service provides the ability to process inventory adjustments and WAC recalculation in Merchandising for item transformations from an external system such as the Fresh Product Management System. The service accepts the list of input & output items involved in the transformation and carry out inventory adjustments to account for inventory conversions in the transformation. Based on the product transformations details, the inventory of the input item(s) will be decremented, and the inventory of the output item(s) will be incremented.

An item transformation may also include the costs involved in labor and packaging. In order to communicate the conversion cost to Merchandising as this will have impact on Stock Ledger postings and WAC calculation, the transformation request can also carry activity codes for each transformation activity involved in producing the output items. The cost associated with each of these activities can be predefined in Merchandising or can be interfaced in the transformation message with the activity code.

The service supports a collection of item transformation request and will return success or failure through the service response object. If there are upload validation errors, the record will be rejected, and a failure status will be returned as part of the response object in the web service call.

Service Type

POST

ReST URL

MerchIntegrations/services/item/transformations/create

Input Payload Details

Table 5-263 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	Collection of item transformations.

Table 5-264 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transformationId	Yes	String (15)	The external reference id corresponding to a transformation. This should be unique at LOCATION and TRANSFORMATION DATE level.
location	Yes	Number (10)	The stockholding company store or the stockholding virtual warehouse at which the transformation was performed.
locationType	Yes	String (1)	The type of the location. Valid values are S - Store and W - Warehouse.
transformationDate	No	date	The date of transformation. This can be a past or present date in reference to the Vdate. Valid format is 'yyyy-mm-dd'.
workOrderActivity	No	Collection of Object	Collection of all the work order activities involved in the transformation.
transformationInput	Yes	Collection of Object	Collection of the input item details involved in the transformation.
transformationOutput	Yes	Collection of Object	Collection of the output item details involved in the transformation.

Table 5-265 WorkOrderActivity - Object. See list of elements for detail

Element Name	Required	Data Type	Description
activityCode	Yes	String (10)	The activity code corresponding to the work order activity involved in transformation such as labor and packaging cost.
activityCost	No	Number (20,4)	The cost involved in the work order activity.
currencyCode	No	String (3)	The currency in which the activity cost is provided. If the value is null and an activity cost is provided, then it is assumed that the given activity cost is in the location's currency.

Table 5-266 TransformationInput - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The transaction level or below transaction level item which was used as an input item or as an output item in the transformation activity.

Table 5-266 (Cont.) TransformationInput - Object. See list of elements for detail

Element Name	Required	Data Type	Description
disposition	No	String (10)	This value is used to determine if the quantity of the input/output item consumed/generated in the transformation is from available or unavailable inventory buckets based on the dispositions INV_STATUS value in the INV_STATUS_CODES table.
quantity	Yes	Number (12,4)	The quantity consumed for the input item or the quantity generated for the output item.
quantityUom	No	String (4)	The UOM corresponding to the quantity value. It should be a UOM from the same uom class as that of the uom class of the Standard UOM of the item.
weightType	No	String (6)	Indicates whether the WEIGHT is the Total weight (T) or the Weight per unit (WPU). Valid values are defined in code detail with code type INVW.
weight	No	Number (12,4)	Weight of the item utilized (input) or generated (output). For weight type Total, the given weight is the Total weight for the quantity utilized or generated.
weightUom	No	String (4)	Unit of Measure of the weight.
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during Item Transformation message. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on item transformation messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Table 5-267 CreateError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemTransformationError	Yes	Collection of Object	Collection of Item transformation error records.

Table 5-268 ItemTransformationError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transformationId	Yes	String (15)	The external reference id corresponding to a transformation. This should be unique at LOCATION and TRANSFORMATION DATE level.
location	Yes	Number (10)	The stockholding company store or the stockholding virtual warehouse at which the transformation was performed.
locationType	Yes	String (1)	The type of the location. Valid values are S - Store and W - Warehouse.
transformationDate	Yes	date	The date of transformation. This can be a past or present date in reference to the Vdate. Valid format is 'yyyy-mm-dd'.
errors	Yes	Array of String	Collection of error messages.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "transformationId": "10001",
      "location": 1531,
      "locationType": "S",
      "transformationDate": "2001-12-31",
      "workOrderActivity": [
        {
          "activityCode": "12122",
          "activityCost": 10,
          "currencyCode": "USD"
        }
      ],
      "transformationInput": [
        {
          "item": "101050103",
          "disposition": "ATS",
          "quantity": 2,
          "quantityUom": "EA",
          "weightType": "WPU",
          "weight": 2.5,
          "weightUom": "KG",
          "inventoryIdentifierType": "L",
          "inventoryId": "5346"
        }
      ],
      "transformationOutput": [
        {
          "item": "101050103",
```

```

        "disposition": "ATS",
        "quantity": 2,
        "quantityUom": "EA",
        "weightType": "WPU",
        "weight": 2.5,
        "weightUom": "KG",
        "inventoryIdentifierType": "L",
        "inventoryId": "5346"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-269 CreateError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemTransformationError	Yes	Collection of Object	Collection of Item transformation error records.

Table 5-270 ItemTransformationError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transformationId	Yes	String (15)	The external reference id corresponding to a transformation. This should be unique at LOCATION and TRANSFORMATION DATE level.
location	Yes	Number (10)	The stockholding company store or the stockholding virtual warehouse at which the transformation was performed.
locationType	Yes	String (1)	The type of the location. Valid values are S - Store and W - Warehouse.
transformationDate	Yes	date	The date of transformation. This can be a past or present date in reference to the Vdate. Valid format is 'yyyy-mm-dd'.
errors	Yes	Array of String	Collection of error messages.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [

```

```

    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "itemTransformationError": [
        {
          "transformationId": "String",
          "location": 190104,
          "locationType": "?",
          "transformationDate": "2001-12-31",
          "errors": [
            "String"
          ]
        }
      ]
    }
  ]
}

```

Returns To Vendor

The following services are included in this functional area:

- [Return to Vendor Download Service](#)
 - [RTV Publish Services](#)
- [Return to Vendor Upload Service](#)
 - [Manage Returns to Vendor](#)

Return to Vendor Download Service

The following services are included in this functional area:

- [RTV Publish Services](#)

RTV Publish Services

Endpoints

MerchIntegrations/services/inventory/rtv

MerchIntegrations/services/inventory/rtv/{rtvOrderNo}

Functional Area

Inventory – Returns to Vendor

Business Overview

This service publishes return-to-vendor data to downstream consuming systems. It publishes the RTV header together with RTV detail lines, warehouse and supplier enrichment, and header custom flex attributes. Functionally, the view does not apply a direct RTV status filter,

but it only publishes RTVs that currently have at least one detail row. The payload also derives the published location fields from the header so location and locationType resolve to either the store or warehouse side of the RTV.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: RTV_HEAD, RTV_DETAIL

Additional business tables: WH, SUPS, RTV_HEAD_CFA_EXT

JSON cache table: MERCHAPI_EXT_RTV

JSON generation view: V_MERCHAPI_EXT_RTV_JSON

Builds the RTV header from RTV_HEAD, uses aggregated line data from RTV_DETAIL, enriches the payload with warehouse information from WH and supplier information from SUPS, and adds header custom flex attributes from MERCHAPI_EXT_RTV_HEAD_CFA.

The view does not apply a direct RTV status filter, but it only publishes RTVs that currently have at least one detail row.

The payload also derives several location fields from the header: location is either RTV_HEAD.WH or RTV_HEAD.STORE, locationType is S or W based on the same warehouse/store split, and physicalLocation resolves to WH.PHYSICAL_WH or the store.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_RTV_JSON and apply RTV_HEAD.STATUS_IND <> 5.

Mark cache rows deleted when the RTV no longer exists in V_MERCHAPI_EXT_RTV_JSON.

ICL (Integration Change Log) table: ICL_RMS_RTV (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on RTV_HEAD, RTV_DETAIL, and RTV_HEAD_CFA_EXT.

Queue detail changes only when the related header status = 10.

Queue CFA changes only when RTV status not in (5,12,15).

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_RTV

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job: MERCHAPI_DELTA_EXT_RTV_ADHOC_PROCESS -> MERCHAPI_EXT_RTV_ADHOC_JOB

Webhook configuration api name: inventory/rtv

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. Because this service is built primarily from RTV header and detail data with a smaller supporting-table footprint, runtime is usually driven more by transaction volume and detail density than by source SQL complexity, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through `API_NO_OF_THREADS`, commit sizing through `API_COMMIT_MAX_COUNTER`, and long-running initial seeding can also be segmented through `API_EXEC_SECONDS`.

Common starting points for higher-volume runs are 8 or 16 threads, while higher settings such as 32 should be considered only after testing. This service is delivered seeded at 1 thread and should be increased gradually after validating throughput and restart behavior.

`API_COMMIT_MAX_COUNTER` should be sized together with the thread setting. Typical values range from about 1,000 rows and can extend to 10,000 for simpler restartable workloads. Lower commit values may perform better when RTV detail changes arrive in short-term spikes.

`API_EXEC_SECONDS` is mainly relevant for initial seeding. A value of about 30 minutes helps the batch pause and resume cleanly so tuning changes can be applied between runs if needed.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes. `GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a specific reason to change it.

REST fetch considerations

REST page size should be selected based on average payload size so the average response remains below about 10 MB per call. Common page-limit values are 1,000, 5,000, or 10,000 rows.

For full downloads, do not provide the `since` query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned `before` timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/rtv

MerchIntegrations/services/inventory/rtv/{rtvOrderNo}

Input Parameters for MerchIntegrations/services/inventory/rtv

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-271 Input Parameter for "MerchIntegrations/services/inventory/rtv/{rtvOrderNo}"

Parameter Name	Required	Data Type	Description
rtvOrderNo	Yes	String	RTV Order Number

Table 5-271 (Cont.) Input Parameter for "MerchIntegrations/services/inventory/rtv/{rtvOrderNo}"

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/inventory/rtv/{rtvOrderNo}

Parameter Name	Required	Data Type	Description
rtvOrderNo	Yes	Number	This number uniquely identifies the return to vendor within the system.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - RTV Head

Table 5-272 PageResultsItemsMerchApiRTV - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-273 MerchApiRTV - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for RTV details.
rtvOrderNo	No	Number(10,0)	This field contains the number that uniquely identifies the return to vendor within the system.
supplier	No	Number(10,0)	This field contains the unique identifier for the supplier or supplier site to which the merchandise is being returned.
supplierName	No	String (240)	This field contains the supplier's trading name.
status	No	Number(2,0)	This field contains the current status of the return. Valid values include 10 - Approved, 12 - In Progress, 15 - Shipped, and 20 - Cancelled.
location	No	Number(10,0)	This field contains the unique identifier for the location where the items are being returned from.
locationType	No	String (1)	This field contains the type of location from which the items are being returned.
physicalLocation	No	Number(10,0)	This field contains the store number or the physical warehouse from which the items are being returned.
totalOrderAmount	No	Number(20,4)	This field contains the total amount of the return. This field is stored in the supplier's currency.
shipToAddress1	No	String (240)	This field contains the first line of the supplier's address for returns.
shipToAddress2	No	String (240)	This field contains the second line of the supplier's address for returns.

Table 5-273 (Cont.) MerchApiRTV - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shipToAddress3	No	String (240)	This field contains the third line of the supplier's address for returns.
shipToCity	No	String (120)	This field contains the city name of the supplier's address for returns.
state	No	String (3)	This field contains the state of the supplier's address for returns.
shipToCountry	No	String (3)	This field contains the country ID of the supplier's address for returns.
shipToPostalCode	No	String (30)	This field contains the zip code of the supplier's address for returns.
returnAuthorizationNo	No	String (12)	This field contains the number that the supplier provides when the decision is made that an order may be returned.
returnCourier	No	String (250)	This field contains the courier name if the supplier requires that returns be sent by a designated carrier.
freightCharge	No	Number(20,4)	This field contains the freight cost associated with the vendor return, if applicable, and is stored in the supplier's currency.
creationDate	No	date	This field contains the date on which the RTV order was created.
completedDate	No	date	This field contains the date the RTV order was completed.

Table 5-273 (Cont.) MerchApiRTV - Object. See list of elements for detail

Element Name	Required	Data Type	Description
restockingPercentage	No	Number(12,4)	This field contains the restocking fee percentage applicable to the order. This is the fee as a percentage of the total return value charged by the supplier for recovering the expenses that may have to be incurred while restocking them at the supplier's location.
restockingCost	No	Number(20,4)	This field contains the restocking fee value. This is calculated based on the restocking percentage by multiplying it by the total return value.
externalReferenceNo	No	String (30)	This field contains an audit trail reference to an external system when an RTV is created by external store or warehouse systems.
comments	No	String (2000)	This field contains any comments associated with the return.
notAfterDate	No	date	This field contains the last date that an RTV can be sent to a supplier.
massReturnTransferNo	No	Number(10,0)	This field holds the unique identifier for the mass return transfer if the RTV is created through a mass return transfer.
shipToJurisdictionCode	No	String (10)	This field identifies the jurisdiction code for the country-state relationship.
rtvOriginatedInd	No	String (6)	This field identifies the source of the return present in the table.
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).

Table 5-273 (Cont.) MerchApiRTV - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).
details	No	Collection of Object	This list holds attributes for RTV detail records.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the RTV.
cacheTimestamp	No	dateTime	This field specifies the date and time when the RTV record was first picked for processing from the ICL table.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-274 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	This field contains the unique identifier for the item being returned to the supplier.
shipment	No	Number(12,0)	This field is no longer used by RMS.
invStatus	No	Number(2,0)	This field contains the number representing the inventory status code. This field is only required if the reason field has a value of U for unavailable inventory.
quantityRequested	No	Number(12,4)	This field contains the quantity of this item requested to be returned to the supplier.

Table 5-274 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitCost	No	Number(20,4)	This field contains the cost per unit for the item being returned that is expected to be recovered from the supplier. This field is stored in the supplier's currency.
reason	No	String (6)	This field contains the value identifying the reason for the return. Valid values for this field are defined on CODE_DETAIL where CODE_TYPE = 'RTVR'.
sequenceNo	No	Number(4,0)	This field contains a sequence number which makes the record unique when combined with the RTV order number.
quantityCancelled	No	Number(12,4)	This field contains the difference between quantity requested and quantity returned. This is updated after the RTV has been shipped.
quantityReturned	No	Number(12,4)	This field contains the quantity of this item that was shipped to the supplier on this RTV.
restockPercentage	No	Number(12,4)	This field contains the restocking fee percentage applicable to the item. This is the fee as a percentage of the total return value charged by the supplier for recovering the expenses that may have to be incurred while restocking them at the supplier's location.
originalUnitCost	No	Number(20,4)	This field contains the current weighted average cost for the item at the return location.

Table 5-274 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updatedByRmsInd	No	String (1)	This field indicates if the last update to RTV_DETAIL was done by RMS or an external application. If by RMS, the indicator will be set to Y and a message will be published to external systems. If by an external application, the indicator will be set to N and a message will not be published to external systems.
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).

Table 5-275 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-276 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "rtvOrderNo": 1,
      "supplier": 1,
      "supplierName": null,
      "status": 1,
      "location": 1,
      "locationType": null,
      "physicalLocation": 1,
      "totalOrderAmount": 1.0,
      "shipToAddress1": null,
      "shipToAddress2": null,
      "shipToAddress3": null,
      "shipToCity": "Minneapolis",
      "state": "MN",
      "shipToCountry": "US",
      "shipToPostalCode": null,
      "returnAuthorizationNo": null,
      "returnCourier": null,
      "freightCharge": 1.0,
      "creationDate": "2001-12-31",
      "completedDate": "2001-12-31",
      "restockingPercentage": 1.0,
      "restockingCost": 1.0,
      "externalReferenceNo": null,
      "comments": "commentDesc",
      "notAfterDate": "2001-12-31",
      "massReturnTransferNo": 1,
      "shipToJurisdictionCode": null,
      "rtvOriginatedInd": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "details": [
        {
          "item": null,
          "shipment": 1,

```

```

    "invStatus": 1,
    "quantityRequested": 3,
    "unitCost": 9.65,
    "reason": null,
    "sequenceNo": 1,
    "quantityCancelled": 1.0,
    "quantityReturned": 1.0,
    "restockPercentage": 1.0,
    "originalUnitCost": 1.0,
    "updatedByRmsInd": null,
    "createDateTime": "2001-12-31T23:59:59.000Z",
    "updateDateTime": "2001-12-31T23:59:59.000Z"
  },
  ],
  "customFlexAttribute": [
    {
      "name": "attribute1",
      "value": "value1",
      "valueDate": "2001-12-31",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ],
  "cacheTimestamp": "2001-12-31T23:59:59.000Z",
  "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
  {
    "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
    "rel": "next"
  }
]
]
}

```

Output - RTV Details

Name	Data Type	Description
item	String	Unique identifier for the item.
shipment	Number	The RTV shipment number.
invStatus	String	The inventory status code. Only applicable if the reason field has a value of 'U' for unavailable inventory.
quantityRequested	Number	The quantity of items requested to be returned to the supplier.
unitCost	Number	The cost per unit for the SKU being returned in the supplier's currency.
Reason	String	The cost per unit for the SKU being returned in the supplier's currency.

Name	Data Type	Description
sequenceNo	Number	The sequence number of the RTV detail record that is being deleted.
quantityCancelled	Number	The difference between qty_requested and qty_returned. This is updated after the RTV has been shipped.
quantityReturned	Number	This field contains the quantity of this item that was shipped to the supplier on this RTV.
restockPercentage	Number	This field contains the Restocking Fee percentage applicable to the item. This is the fee as percentage of the total return value charged by the supplier for recovering the expenses that may have to be incurred while restocking them at suppliers location.
originalUnitCost	Number	This field contains the current weighted average cost for the item at the return location.
updatedByRmsInd	String	This field indicates if the last update to RTV_DETAIL table is done by RMS or an external application. If by RMS, then the indicator will be set to Y and a message will be published to external systems; otherwise, the indicator will be set to N and a message will not be published to external systems.
createDateTime	String	Date time when record was inserted.
updateDateTime	string	Date time when record was updated.

Output - Custom Flex Attributes

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.

Elements	Description
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```
{
  "items": [
    {
      "action": "NO_CHANGE",
      "rtvOrderNo": 0,
      "supplier": 0,
      "supplierName": "string",
      "status": 0,
      "location": 0,
      "locationType": "string",
      "physicalLocation": 0,
      "totalOrderAmount": 0,
      "shipToAddress1": "string",
      "shipToAddress2": "string",
      "shipToAddress3": "string",
      "shipToCity": "string",
      "state": "string",
      "shipToCountry": "string",
      "shipToPostalCode": "string",
      "returnAuthorizationNo": "string",
      "returnCourier": "string",
      "freightCharge": 0,
      "creationDate": "string",
      "completedDate": "string",
      "restockingPercentage": 0,
      "restockingCost": 0,
      "externalReferenceNo": "string",
      "comments": "string",
      "notAfterDate": "string",
      "massReturnTransferNo": 0,
      "shipToJurisdictionCode": "string",
      "originInd": "string",
      "createDateTime": "string",
      "updateDateTime": "string",
      "details": [
        {
          "item": "string",
          "shipment": "string",
          "invStatus": "string",
          "quantityRequested": 0,

```

```

        "unitCost": 0,
        "reason": "string",
        "sequenceNo": 0,
        "quantityCancelled": 0,
        "quantityReturned": 0,
        "restockPercentage": 0,
        "originalUnitCost": 0,
        "updatedByRmsInd": "string",
        "createDateTime": "string",
        "updateDateTime": "string"
      }
    ],
    "customFlexAttribute": [
      {
        "name": "string",
        "value": "string",
        "valueDate": "string",
        "createDateTime": "string",
        "updateDateTime": "string"
      }
    ],
    "rtvOriginatedInd": "string",
    "cacheTimestamp": "string"
  }
],
"hasMore": true,
"limit": 0,
"count": 0,
"links": [
  "SELF_LINK"
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_RTV	Yes	No	Yes	Yes
MERCHAPI_EXT_RTV	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_RTV_JSON	Yes	No	No	No

Return to Vendor Upload Service

The following services are included in this functional area:

- [Manage Returns to Vendor](#)

Manage Returns to Vendor

Functional Area

Inventory - Returns to Vendor

Business Overview

Merchandising subscribes to return-to-vendor (RTV) messages when an RTV is shipped out from a warehouse or store. This shipment could be for an RTV that was initially created in Merchandising, or one initiated at the store or warehouse. The RTV information is sent from a warehouse management system (WMS), such as Oracle WMS Cloud or the store inventory solutions, such as Oracle Retail Store Inventory and Operations Cloud Service (SIOCS) when the RTV is created or shipped out of the location. Additionally, RTV information may also come from an external application where RTVs are planned. The external system can create, cancel or update RTVs using this service in the similar manner these are created in Merchandising.

This service allows the above listed systems to create, update or cancel RTVs in Merchandising. Create, update and cancel messages can be sent through this single service.

This service accepts the following status:

A - Approved

P - In-Progress

S - Shipped

C - Cancelled

Create RTVs

If the message contains a new RTV generated in the store or warehouse, then it must contain both header and detail information. RTVs from external systems can be created in Approved, In-Progress or Shipped status. RTVs created in Approved status will be editable through Merchandising screen where details can be updated including an approved RTVs being shipped and cancelled. RTVs created in Approved status via this service will be integrated to downstream systems.

Update RTVs

RTVs in Input (created via Merchandising UI), In-Progress, Approved and Shipped status can be updated using this service. To update an RTV, you can send either only the header information or both header and detail level information. The most common update is to ship a previously created RTV. While updating an approved or in-progress RTV to the Shipped status, if only the header information is provided, then the requested quantity of all the items that are present in the approved RTV will be shipped. Note: Once RTVs are Shipped, they cannot be changed back to the approved status. Alternatively, approved RTVs can also be set to Cancelled status by updating the quantities in case they cannot be shipped for some reason.

Cancel RTV

An RTV can be cancelled without specifying the individual item quantities by sending the header information with Cancelled status. Only Input and Approved RTVs can be cancelled.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/returnsToVendor/manage

Input Payload Details

Table 5-277 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of return to vendor details.

Table 5-278 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rtvNo	No	Number (10)	Contains the number that uniquely identifies the return to vendor within the system. If both the rtvNo and extRefNo are provided, it is assumed that an existing externally generated RTV that contains the same rtvNo/ extRefNo combination needs to be updated. If no match is found, then the service will return an error. In order to create a new return to vendor, the rtvNo field must be blank. When updating a return to vendor that was initiated in Merchandising, both the rtvNo and extRefNo must be provided but only the rtvNo is expected to match since the extRefNo is blank for these types of RTVs.
location	Yes	Number (10)	Contains the location number that the items are returned from. If the location is a store, the STORE column gets populated with the location value, and the WH column is defaulted to -1. If the location is a warehouse, the WH column gets populated.
externalReferenceNo	No	String (30)	Audit trail reference to external system when an external transaction initiates master record creation in Merchandising.
returnAuthorizationNo	No	String (12)	Contains the number that the supplier provides when the decision is made that an order may be returned. This value is not always required and is determined by vendor level data.
supplier	Yes	Number (10)	Contains the supplier ID to which the merchandise is being returned.
shipToAddress1	No	String (240)	Contains the first line of the supplier's address for returns.
shipToAddress2	No	String (240)	Contains the second line of the supplier's address for returns.
shipToAddress3	No	String (240)	Contains the third line of the supplier's address for returns.

Table 5-278 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shipToState	No	String (3)	Contains the state of the supplier's address for returns.
shipToCity	No	String (120)	Contains the city name of the supplier's address for returns.
shipToPostalCode	No	String (30)	Contains the zip code of the supplier's address for returns.
shipToCountry	No	String (3)	Contains the country ID of the supplier's address for returns.
createdDate	No	date	Contains the date the vendor return was created.
status	No	String (1)	This value is used to determine the current status of the return. If this is 'A', Merchandising will set the status to 10 (Approved). If this is 'P', Merchandising will set the status to 12 (In Progress). If this is 'S' or null, it will be set to 15 (Shipped). If this is 'C', Merchandising will set the status to 20 (Cancelled).
comments	No	String (2000)	Contains any comments associated with the return.
details	No	Collection of Object	References a collection of items on the RTV. If no details are provided for the RTV status being updated from Approved to Shipped, then the quantity requested for all the items that are present in the RTV will be shipped.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
consignmentVendorRtvInd	No	String (1)	This indicator determine whether the RTV being interfaced is a Consignment Vendor RTV.

Table 5-279 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Unique identifier for the item on the RTV.

Table 5-279 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitQuantity	Yes	Number (12,4)	Contains the item quantity that is being returned to the supplier under this RTV number. When updating the quantity in an RTV, the value of the return/requested quantity will be increased if the unit quantity provided is positive whereas a negative unit quantity will increase the quantity cancelled. This quantity cannot be negative for RTVs in shipped status, and should always be greater than zero when creating a new RTV detail record.
fromDisposition	No	String (4)	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.
unitCost	No	Number (20,4)	Contains the cost per unit for the SKU being returned. This field is stored in the supplier's currency.
reason	No	String (6)	Contains the value identifying the reason for the return. Valid values are: Q - QC Failed, U - Unavailable Inventory. Other values may also be entered via the codes table under code type 'RTVR', however, the two codes listed previously may never be deleted. Note that any value added can only be related to RTV from existing inventory (e. g. overstock).
weight	No	Number (12,4)	Contains the actual weight shipped for the items on the RTV. This should not be provided for RTVs in the approved status.
weightUom	No	String (4)	Unit of Measure for the weight (for example, pounds, kilograms).
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during Return to Vendor Integration. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on inventory transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Table 5-280 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "rtvNo": 188295,
      "location": 1521,
      "externalReferenceNo": "RETURN01",
      "returnAuthorizationNo": "5522014",
      "supplier": 2400,
      "shipToAddress1": "2243 Pennys Way",
      "shipToAddress2": null,
      "shipToAddress3": null,
      "shipToState": "ORE",
      "shipToCity": "Woodburn",
      "shipToPostalCode": "4522",
      "shipToCountry": "US",
      "createdDate": "2001-12-31",
      "status": "A",
      "comments": "creating RTV via ReST service",
      "details": [
        {
          "item": "100100068",
          "unitQuantity": 3,
          "fromDisposition": "ATS",
          "unitCost": 9.65,
          "reason": "0",
          "weight": 1,
          "weightUom": "KG",
          "inventoryIdentifierType": "L",
          "inventoryId": "5346"
        }
      ],
      "customFlexAttribute": [
        {
          "name": "COMMENT",
          "value": "CFA Comment creating RTV via ReST service",
          "valueDate": "2001-12-31"
        }
      ],
      "consignmentVendorRtvInd": "Y"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Shipments and Receipts

The following services are included in this functional area:

- [Receiver Unit Adjustment Download Service](#)
 - [Get Receiver Unit Adjustment](#)
- [Shipment and Receipt Download Service](#)
 - [Get Shipment Detail](#)
- [Shipment and Receipt Upload Service](#)
 - [Create DSD Receipts](#)
 - [Create Outbound ASNs](#)
 - [Delete Inbound ASNs](#)
 - [Manage Inbound ASN](#)
 - [Manage Purchase Order and Stock Order Receipts](#)
 - [Manage Purchase Order Receipts](#)
 - [Manage Stock Order Receipts](#)
 - [Update DSD Receipts](#)

Receiver Unit Adjustment Download Service

The following services are included in this functional area:

- [Get Receiver Unit Adjustment](#)

Get Receiver Unit Adjustment

Endpoint

MerchIntegrations/services/inventory/receiverunitadj

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service publishes receiver unit adjustment data to downstream consuming systems. It publishes incremental receiver-unit-adjustment transactions together with the ASN reference, receiving location, adjusted unit quantity, and from/to disposition movement. Functionally, the upstream receiver-unit-adjustment logic writes to RUA_RIB_INTERFACE when the calling program is REIM or RECUTADJ, and only when the adjusted quantity is non-zero.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: RUA_RIB_INTERFACE

Additional business tables: SHIPMENT

JSON cache table: MERCHAPI_EXT_RCV_UNIT_ADJ

JSON generation view: not applicable; cache rows are populated directly from ICL_RMS_RCV_UNIT_ADJ and SHIPMENT by the delta-processing logic

Functionally, the upstream receiver-unit-adjustment logic only writes to RUA_RIB_INTERFACE when the calling program is REIM or RECUTADJ, and only when the adjusted quantity is non-zero.

The published asnNo is derived as the first available value of SHIPMENT.EXT_REF_NO_IN or SHIPMENT.ASN), so SIM's external receipt number takes precedence over the shipment ASN when it exists.

This API is delta-only by design.

Initial Seeding and post enablement rebuild/repair

This API has no initial load or rebuild; the refresh step only performs API-disable handling for the cache table.

ICL (Integration Change Log) table: ICL_RMS_RCV_UNIT_ADJ (ICL consumer name = MERCHAPI)

The delta async process reads unprocessed ICL_RMS_RCV_UNIT_ADJ rows for consumer MERCHAPI, inserts one cache row per event, and deletes the processed ICL rows after publication.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_RUA

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job: MERCHAPI_DELTA_EXT_RUA_ADHOC_PROCESS - > MERCHAPI_EXT_RUA_ADHOC_JOB

Webhook configuration api name: inventory/receiverunitadj

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/receiverunitadj

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Sequence Number)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-281 PageResultsItemsMerchApiRua - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-281 (Cont.) PageResultsItemsMerchApiRua - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-282 MerchApiRua - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for receiver unit adjustment details.
location	No	Number(10,0)	This field contains the unique identifier for the location for which the receiver unit adjustment is performed. The location will be a store if the location type is S, and a warehouse if the location type is W.
locationType	No	String (1)	This field contains the type of the location for which the receiver unit adjustment is performed. Valid values are S for store and W for warehouse.
orderNo	No	Number(12,0)	This field contains the unique identifier for the order against which a receiver unit adjustment was performed.

Table 5-282 (Cont.) MerchApiRua - Object. See list of elements for detail

Element Name	Required	Data Type	Description
asnNo	No	String (30)	This field contains the advance shipment notification number associated with the shipment for which a receiver unit adjustment was performed.
item	No	String (25)	This field contains the unique identifier for the item for which the receiver unit adjustment is being made.
containerId	No	String (30)	This field contains the unique identifier for the carton that contained the item against which the receiver unit adjustment was performed.
unitQuantity	No	Number(12,4)	This field contains the quantity of the receiver unit adjustment.
fromDisposition	No	String (4)	This value, along with the to disposition value, is used to determine if the inventory is available or unavailable.
toDisposition	No	String (4)	This field contains the value used to determine if the inventory is available or unavailable based on the value of the INV_STATUS column in the INV_STATUS_CODES table.
cacheTimestamp	No	dateTime	This field specifies the date and time when the record was first picked for processing from ICL table (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-283 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "location": 1111,
      "locationType": "S",
      "orderNo": 545005,
      "asnNo": "ASN_11223344_2",
      "item": "106400029",
      "containerId": "CTN_1_ASN_11223344_2",
      "unitQuantity": 3,
      "fromDisposition": "ATS",
      "toDisposition": "ATS",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Table Impact

Table Impact: The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	Yes	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_RCV_UNIT_ADJ	Yes	No	Yes	Yes
MERCHAPI_EXT_RCV_UNIT_ADJ	No	Yes	No	Yes
SYSTEM_OPTIONS	Yes	No	No	No

Shipment and Receipt Download Service

The following services are included in this functional area:

- [Get Shipment Detail](#)

Get Shipment Detail

This section describes the Shipment Detail Service.

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service allows users to retrieve shipment and shipment item details for a given distro (transfer or allocation) or purchase order (PO).

Service Type

Get

ReST URL

Shipment/shipmentDetail?
orderNumber={orderNumber}&distroNumber={distroNumber}&distroType={distroType}

Input Parameters

Parameter Name	Required	Description
orderNumber	No	Order Number. If none is specified, then Distro Number and Distro Type are required.
distroNumber	No	Distro Number. If none is specified, then Order Number is required.
distroType	No	Distro Type. If none is specified, then Order Number is required.

Output

RestShipmentRecRDO

Parameter Name	Data Type
shipment	BigDecimal
bolNo	String
asn	String
shipDate	Timestamp
receiveDate	Timestamp

Parameter Name	Data Type
estArrDate	Timestamp
shipOrigin	String
statusCode	String
toLoc	BigDecimal
toLocType	String
fromLoc	BigDecimal
fromLocType	String
parentShipment	BigDecimal
seqNo	BigDecimal
item	String
refItem	String
carton	String
invStatus	BigDecimal
shipskuStatusCode	String
qtyReceived	BigDecimal
unitCost	BigDecimal
unitRetail	BigDecimal
qtyExpected	BigDecimal
adjustType	String
actualReceivingStore	BigDecimal
reconcileUserId	String
reconcileDate	Timestamp
tamperedInd	String
dispositionedInd	String

JSON Structure

```
{
  "shipment":null,
  "bolNo":null,
  "asn":null,
  "shipDate":null,
  "receiveDate":null,
  "estArrDate":null,
  "shipOrigin":null,
  "statusCode":null,
  "toLoc":null,
  "toLocType":null,
  "fromLoc":null,
  "fromLocType":null,
  "parentShipment":null,
  "seqNo":null,
  "item":null,
  "refItem":null,
  "carton":null,
```

```

    "invStatus":null,
    "shipskuStatusCode":null,
    "qtyReceived":null,
    "unitCost":null,
    "unitRetail":null,
    "qtyExpected":null,
    "adjustType":null,
    "actualReceivingStore":null,
    "reconcileUserId":null,
    "reconcileDate":null,
    "tamperedInd":null,
    "dispositionedInd":null,
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_SHIPMENT	Yes	No	No	No
V_SHIPSKU	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Shipment and Receipt Upload Service

The following services are included in this functional area:

- [Create DSD Receipts](#)
- [Create Outbound ASNs](#)
- [Delete Inbound ASNs](#)
- [Manage Inbound ASN](#)
- [Manage Purchase Order and Stock Order Receipts](#)
- [Manage Purchase Order Receipts](#)
- [Manage Stock Order Receipts](#)
- [Update DSD Receipts](#)

Create DSD Receipts

Functional Area

Inventory - Shipments and Receipts

Business Overview

Direct store delivery (DSD) is the delivery of merchandise to and/or the performance of services in a store without the benefit of a pre-approved purchase order. When the delivery or service occurs, the store inventory system (for example, SIOCS) informs Merchandising of the receipt, which also generates the purchase order at the same time. The receipt can include both the merchandise item as well as the non-merchandise information associated with the

order. This information works in conjunction with payment details sent through Sales Audit if payment was made a part of the delivery or service at the store.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/dsd/create

Input Payload Details

Table 5-284 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of DSD Receipts.

Table 5-285 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	This field contains the unique identifying number for a supplier for the receipt.
originCountry	Yes	String (3)	This field contains the identifier of the country from which the item is being sourced.
store	Yes	Number (10)	This field contains the location where the items were delivered.
dept	No	Number (4)	The department in which all the items on the order belong.
currencyCode	Yes	String (3)	This field contains a code identifying the currency the supplier uses for business transactions.
paidInd	Yes	String (1)	This field indicates if the invoice has already been paid. Valid values are Y (invoice has already been paid) or N (invoice should be paid in accounts payable system).
externalReferenceNo	No	String (30)	If the invoice indicator is Y (invoice has been created), the external reference number, proof of delivery number, or payment reference number must be provided.
proofOfDeliveryNo	No	String (30)	This field contains the proof of delivery or service number given at the time of receipt at the store. This field will also be included when the invoice is interfaced through Sales Audit.

Table 5-285 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
paymentReferenceNo	No	String (16)	This field contains the reference number attached to the invoice payment, used when the invoice is paid from the POS system and interfaced through Sales Audit.
paymentDate	No	date	This field contains the date when the invoice was paid from the POS system. This field will be populated when the invoice is interfaced through Sales Audit.
invoiceInd	Yes	String (1)	This field indicates whether an invoice was created for this receipt by the supplier. Valid values are Yes (Y) and No (N).
dealsInd	Yes	String (1)	This field indicates whether deals need to be applied to the DSD Purchase Order or not. Valid values are Yes (Y) and No (N).
externalReceiptNo	No	String (17)	This field holds the external transaction sequence number for the receipt.
receiptDate	No	date	This field contains the date of the receipt.
details	No	Collection of Object	References a collection of DSD Receipt detail records.
nonMerchDetails	No	Collection of Object	References a collection of DSD Non Merchandise detail records.

Table 5-286 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains the item in the receipt. The item must be an approved item that is at transaction level or above.
quantityReceived	Yes	Number (12,4)	This field contains the number of items received for the item/shipment combination. The value must be greater than 0.
unitCost	No	Number (20,4)	This field contains the cost of the item from the supplier/origin country in the supplier's currency.
weight	No	Number (12,4)	This field contains the weight of the item in the receipt, if it is a catch weight item.
weightUom	No	String (4)	This field contains the unit of measure of the received weight.

Table 5-286 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during a DSD receipt message. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on inventory transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Table 5-287 NonMerchDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nonMerchCode	Yes	String (6)	This field contains the non-merchandising code that will be added to an invoice for services or other non-merchandise costs associated with the order.
nonMerchAmount	Yes	Number (20)	This field contains the amount of the non-merchandise cost that was invoiced. This field will be held in the invoice currency.
vatCode	No	String (6)	This field contains the code identifying the VAT rate that should be applied to the non-merchandise amount entered.
servicePerformedInd	Yes	String (1)	Indicates if a service non-merchandise cost has actually been performed. Valid values are 'Y' (service has been performed) or 'N' (service has not been performed or non-merchandise cost is not a service cost).

Sample Input Message

```
{
  "items": [
    {
      "supplier": 2400,
      "originCountry": "US",
      "store": 1531,
      "dept": 1117,
      "currencyCode": "USD",
      "paidInd": "Y",
      "externalReferenceNo": "REF1531",
      "proofOfDeliveryNo": null,
      "paymentReferenceNo": null,
      "paymentDate": "2001-12-31",

```

```

    "invoiceInd": "Y",
    "dealsInd": "Y",
    "externalReceiptNo": "RECREF1531",
    "receiptDate": "2001-12-31",
    "details": [
      {
        "item": "100750001",
        "quantityReceived": 3,
        "unitCost": 9.65,
        "weight": 79.00,
        "weightUom": null,
        "inventoryIdentifierType": "L",
        "inventoryId": "5346"
      }
    ],
    "nonMerchDetails": [
      {
        "nonMerchCode": "B720",
        "nonMerchAmount": 123456,
        "vatCode": "S",
        "servicePerformedInd": "N"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Outbound ASNs

Functional Area

Inventory - Shipments and Receipts

Business Overview

Merchandising receives advanced shipping notifications (ASNs), also known as a bill of lading (BOL) messages, from a warehouse management system, like Oracle Warehouse Management Cloud, or a store inventory system like Oracle Retail Store Inventory and Operations Cloud Service (SIOCS).

These ASNs are notifications to Merchandising that inventory is moving from one location to another and they contain data that is used by Merchandising to create or modify a shipment record. ASNs are received for:

- Pre-existing allocations
- Pre-existing transfers
- Externally generated transfers created in the store or warehouse (created as transfer type of EG within Merchandising).

An ASN message may contain details related to multiple transfers or allocations, and would enable the shipment record in Merchandising to reflect these multiple movements of the merchandise. The BOL number present on the shipment record is a way to track one or more transfers and allocations through their respective stock order records. Shipments for customer orders, franchise orders, and franchise returns are also managed through this service. If the receiving location is a non-stockholding location, like in the case of a warehouse shipment to a non-stockholding franchise store, or a warehouse shipment direct to a customer (that is processed through a non-stockholding store), then the shipment will be automatically received when processed by Merchandising.

Note

ASNs related to a purchase order from a supplier are classified as an Inbound ASNs. Details for those types of expected shipments are found in the ASN In Subscription service documentation.

Other Notes

- For customer order fulfillment, SIOCS will send an ASN Out message that does not include a ship-to location. Such messages are ignored by Merchandising.
- Store to customer fulfillment requests will not have associated transfers in Merchandising. When Oracle Retail Store Inventory and Operations Cloud Service (SIOCS) ships the customer order, then SIOCS will generate an Outbound ASN message with an empty 'To Location' or with the Location Type set as Customer (C). Since there are no associated transfers within Merchandising, it will not process these Outbound ASN messages. The reserved inventory will be backed out when Merchandising processes the related SALES transaction.
- Messages received through this service can create new shipments or update existing shipments. A new shipment record will be created in Merchandising in the 'Input' status if the BOL number is not yet associated to any shipment record. If the BOL number is already associated to a shipment record, the shipment record will be updated accordingly.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/outboundAsn/create

Input Payload Details

Table 5-288 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of ASN details.

Table 5-289 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	No	Number (10)	Contains the location that the shipment will be delivered to.
fromLocation	Yes	Number (10)	For transfer and allocation shipments, this field will hold the location from which the shipment was sourced.
cartonQuantity	No	Number (6)	Contains the number of boxes associated with the shipment.
asnNo	No	String (30)	Holds the bill of lading number associated with a shipment. This field will be persisted in the bol_no column of the Shipment table in Merchandising.
bolNo	Yes	String (17)	Holds the transaction sequence number on the message from the transfer shipment confirmation process. This field will be persisted in the ext_ref_no_out column of the Shipment table in Merchandising.
shipDate	No	date	This field contains the date the transfer or PO was shipped.
estimatedArrivalDate	No	date	This field contains the estimated arrival date of a vendor PO shipment. It is updated by EDIUP856. It is used for vendor/lead time analysis.
comments	No	String (2000)	Contains any miscellaneous comments about the shipment.
carrierCode	No	String (4)	Contains the courier that will deliver the shipment.
shipmentDetails	No	Collection of Object	Description is not available.

Table 5-290 ShipmentDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
distroNo	Yes	Number (12)	Specifies the transfer or allocation number associated with the shipment/item/carton.
distroDocumentType	Yes	String (1)	Specifies what the distro_nbr field corresponds to in Merchandising: V, D, and T specify transfer, A specifies Allocation.
customerOrderNo	No	String (48)	This is the customer order number that was generated by OMS and contains multiple fulfillment numbers.
fulfillOrderNo	No	String (48)	Based on a customer order - OMS will generate fulfillments to specific locations based on availability. RMS will generate Transfers based on the fulfillment request.
comments	No	String (2000)	Not used by RMS.
cartons	Yes	Collection of Object	Description is not available.

Table 5-291 Cartons - Object. See list of elements for detail

Element Name	Required	Data Type	Description
carton	Yes	String (30)	Identifies the UCC-128 carton number for shipments originating from the Advance Shipment Notification process as carton shipments. This field will be zero for all shipments that are not at a carton level.
weight	No	Number (12,4)	Actual weight shipped for the container.
weightUom	No	String (4)	Unit of measurement for weight (e. g. pounds, kilograms) that was shipped.
items	Yes	Collection of Object	Description is not available.

Table 5-292 Cartons.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Unique identifier for the item.
unitQuantity	Yes	Number (12,4)	Contains the number of items expected to be received based on the supplier's advance shipment notification for this item/shipment combination.

Table 5-292 (Cont.) Cartons.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromDisposition	No	String (4)	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.
unitCost	No	Number (20,4)	Contains the unit cost of the item in the shipment.
baseCost	No	Number (20,4)	This value will be used to get the base cost (BC) from RFM for a transfer, which will flow into RMS.
weight	No	Number (12,4)	Actual weight shipped.
weightUom	No	String (4)	Unit of measurement for weight (for example, pounds, kilograms) shipped.
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during shipment of transfer/allocations. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on shipment transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "toLocation": 1221,
      "fromLocation": 1151,
      "cartonQuantity": 5,
      "asnNo": "ASN-100100990",
      "bolNo": "BOL-100100990",
      "shipDate": "2001-12-31",
      "estimatedArrivalDate": "2001-12-31",
      "comments": "Shipment for transfer 100100990",
      "carrierCode": "FDX",
      "shipmentDetails": [
        {
          "distroNo": 100100990,
          "distroDocumentType": "T",
          "customerOrderNo": null,
          "fulfillOrderNo": null,
          "comments": "Shipment for transfer 100100990",
          "cartons": [
            {
              "carton": "CRTN10001",

```

```

        "weight": 3,
        "weightUom": "KG",
        "items": [
          {
            "item": "100100991",
            "unitQuantity": 10,
            "fromDisposition": "ATS",
            "unitCost": 9.65,
            "baseCost": 10.65,
            "weight": 3,
            "weightUom": "KG",
            "inventoryIdentifierType": "L",
            "inventoryId": "5346"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Delete Inbound ASNs

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service is used to delete shipments from Merchandising that have not been received. Only ASN shipments (Ship Origin = 0) and ASN UCC-128 shipments (Ship Origin = 6) can be deleted using this service.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/inboundAsn/delete

Input Payload Details

Table 5-293 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of ASN details for deletion.

Table 5-294 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
asnNo	Yes	String (30)	Not used in Merchandising.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "asnNo": "ASN-464690028"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Manage Inbound ASN

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service creates a new shipment record or updates an existing shipment for ASNs that have not been received once the supplier or consolidator sends an advanced shipping notice (ASN) to Merchandising. Merchandising subscribes to this ASN information and stores the information in the related tables depending upon the validity of the records enclosed within the ASN message. If an existing ASN number and carton combination already exists, the shipment details (shipment date, carrier code, unit quantity, etc.) will be updated with the values in the request, otherwise a new shipment record will be created.

The ASN message will consist of a header record, a series of order records, carton records, and item records. For each message, the header, order and item records are required while the carton portion of the record is optional. However, if a carton record is present or the ASN type is 'C', then the carton details must be present. The header record will contain information about the shipment, such as its destination location and estimated arrival date. The order records will identify the purchase orders that are associated with the shipment. If the shipment is packed in cartons, then the carton records will identify the items present in the cartons. The item records will contain details on the items that are present on the shipments, along with the quantity that will be shipped.

The location that is contained on the ASN will represent the expected receiving location for the order. If the location is set up as a non-stockholding store in Merchandising, then the shipment will also be automatically received when the ASN is processed. The two types of non-stockholding stores/orders for which orders are supported in this integration are Franchise stores and Customer orders.

Once the ship quantity is matched, an invoice is generated for Invoice Matching.

Note

This message can also be used by stores and warehouses for inbound transfer and allocation shipments. However, for Merchandising, all those shipments are processed as Outbound ASNs.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/inboundAsn/manage

Input Payload Details

Table 5-295 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of ASN details.

Table 5-296 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	No	Number (10)	Contains the location that the shipment will be delivered to. For purchase orders this will always be either a store or a physical warehouse.
asnNo	Yes	String (30)	Not used in Merchandising.
shipDate	Yes	date	This field contains the date the transfer or PO was shipped.
estimatedArrivalDate	No	date	This field contains the estimated arrival date of a vendor PO shipment. It is updated by EDIUP856. It is used for vendor/lead time analysis.
comments	No	String (2000)	Contains any comments about the shipment.
carrierCode	No	String (4)	This column contains a code that indicates the carrier that is involved in the shipment.
asnType	Yes	String (1)	This field is used to determine the ship origin. If asn_type is 'C', ship_origin will be set to '6' (ASN UCC-128) upon insert to the SHIPMENT table. Otherwise, SHIP_ORIGIN is defaulted to '0' (ASN Shipment).
supplier	Yes	Number (10)	Contains the supplier who will provide the merchandise specified in the order. This element is used to validate the order number(s) in ASNInPO (ASNInPO, po_nbr). The order number must be in the table ORDHEAD, in either the ORDER_NO column or VENDOR_ORDER_NO column.

Table 5-296 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shipPayMethod	No	String (2)	This column indicates the payment terms for freight charges associated with the order. Valid values are found in code type SHMT. Examples are Collect (CC), Collected Freight Credited Back to Cust (CF) and Defined by Buyer and Seller (DF).
asnDetails	Yes	Collection of Object	References a collection of ASN details.

Table 5-297 AsnDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	Identifies the order number which relates to the goods delivered in the shipment. Validated against the ORDHEAD table, and also the SHIPMENT table if the message applies to an existing shipment.
notAfterDate	No	date	This field contains the last date that delivery of the order will be accepted.
containers	No	Collection of Object	References a collection of shipment cartons.
items	No	Collection of Object	References a collection of items and item details.

Table 5-298 Containers - Object. See list of elements for detail

Element Name	Required	Data Type	Description
containerId	Yes	String (30)	Holds the UCC-128 carton number.
finalLocation	Yes	Number (10)	This will be the final destination of the carton. For a cross-dock order this will be the allocation location, otherwise it will be the direct to order location.
items	Yes	Collection of Object	References a collection of items and item details.
trackingNbr	No	String (120)	This is a unique tracking number that is used to track containers through a carriers system. Not used in Merchandising.

Table 5-299 Containers.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Unique identifier for the item. Either vpn, item_id, or ref_item must be specified in the message.
unitQuantity	Yes	Number (12,4)	This column contains the quantity of the item that is expected to be received.
vpn	No	String (30)	This column contains the vendor product number used to find the item number. Either VPN, item number, or reference item must be specified in the message.
referenceItem	No	String (25)	The column contains a bar code or reference item. Either VPN, item number, or reference item must be specified in the message.
uins	No	Collection of Object	Collection of Universal Identification Numbers. Not used in Merchandising.

Table 5-300 Uins - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uin	Yes	String (128)	Universal Identification Number

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "toLocation": 2,
      "asnNo": "ASN-464690028",
      "shipDate": "2001-12-31",
      "estimatedArrivalDate": "2001-12-31",
      "comments": "commentDesc",
      "carrierCode": "FDX",
      "asnType": "C",
      "supplier": 2400,
      "shipPayMethod": "PO",
      "asnDetails": [
        {
          "orderNo": 464690028,
          "notAfterDate": "2001-12-31",
          "containers": [
            {
              "containerId": "00152100-1",
              "finalLocation": 1521,
              "items": [
                {
                  "item": "103900095",
                  "unitQuantity": 3,

```

```

        "vpn": "vpn-00152100-1-103900095",
        "referenceItem": null,
        "uins": [
            {
                "uin": "103-64323123-12"
            }
        ]
    },
    "trackingNbr": "1234567A89"
},
"items": [
    {
        "item": "103900095",
        "unitQuantity": 3,
        "vpn": "vpn-00152100-1-103900095",
        "referenceItem": null,
        "uins": [
            {
                "uin": "103-64323123-12"
            }
        ]
    }
]
}
]
}
]
}
]
}
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

```
]
}
```

Manage Purchase Order and Stock Order Receipts

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service processes receipts that Merchandising receives against purchase orders, transfers, and allocations received at a store, warehouse or finisher. Purchase orders can be received at the item level only using this message, while transfers and allocations, collectively referred to as stock orders, may be received at the bill of lading (BOL) level, where the entire shipment is received without including details, carton level, or item level.

When a transfer, PO or allocation is received at a location, Merchandising will update the appropriate tables, including the shipment, the transfer, allocation, purchase order, stock on hand, as well as other inventory buckets (for example, in-transit). A record of the receipt is also made in the Stock Ledger.

Receiving Exceptions

Merchandising has the ability to automatically resolve several common exceptions that can occur during the receiving process. The following exceptions can be automatically processed when a stock order is received through this service:

Receipt Against the Wrong BOL

In this case, the receiving location should send a carton status of Dummy (D) or Overage (O), indicating that a dummy BOL number was used. But, even if that status is not used, this exception processing can still take place. Merchandising will attempt to match the contents of the receipt to a valid BOL as follows:

If the carton belongs to a valid BOL at the given location, Merchandising receives the carton against the intended BOL at the given location.

If the carton belongs to a valid BOL at a related walk-through store, Merchandising receives the carton against the intended BOL at the intended location.

If the carton belongs to a valid BOL at an unrelated location, Merchandising uses the wrong store receiving process.

Walk-through Store Receiving

If you have configured two or more stores as 'walk-through' locations, via attribution at on the store table in Merchandising, then if the BOL was intended to be received at the walk-through location instead of the location on the message, Merchandising will automatically adjust the receipt and process against the correct location.

Misdirected Container

Misdirected containers, or wrong store receiving, is when one or more containers on a receipt is identified as having been originally shipped to a different location (Location A) than the location that sent the receipt message (Location B). Whether or not misdirected container receiving is supported in Merchandising is controlled by a system option called Wrong Store Receipt Exception Handling. If this option is unchecked (N), then the receipt at the Location B will raise an error in this service. If set to checked (Y), then the shipment at Location A will be backed out, including in-transit updates, WAC adjustments, and stock ledger postings, and re-applied to Location B, prior to processing the receipt into Location B.

In order correctly manage this processing, Merchandising must receive the original carton number on the receipt. In some cases, such as when integrating with Store Inventory and Operations Cloud Service (SIOCS), the carton ID is reassigned by the receiving location. In that case, the reference carton field in the Receipt Detail node of the message must be populated to trigger this process. Otherwise, it will be treated as an overage at the actual receiving location and the original location will not have its quantities reversed until the transfer is reconciled.

Note: Wrong location receiving is supported only for item-level transfer/allocation receipts.

Unwanted Cartons

An unwanted carton is a situation where Merchandising never received notification of the original shipment, only the receipt. In this case, if receiving is done at the item level, Merchandising will process both the shipment and receipt together. If item level details are not included for the carton, an error will be raised, as Merchandising will not be able to determine the contents of the carton, having never received the initial shipment details.

Zero Receipts

This type of exception occurs when a location indicates to Merchandising that nothing was received for the item at the location by sending a receipt of zero and indicating that the carton is closed. Merchandising will reconcile the original ship to location based on system option settings to determine where to write off the lost items.

If a zero receipt occurs for an item that is part of a misdirected container, then some slightly different processing will occur. If the zero receipt is sent after a misdirected container reconciliation, then no further updates will be made, as the line would have been previously reconciled. If a zero receipt occurs before misdirected container processing, then the misdirected container processing at the actual receiving location will be treated as an overage, as the original location would have already been reconciled.

Other Key Notes

Externally generated warehouse-to-warehouse transfers are not supported in Merchandising, where the transfer is created in Merchandising at the physical warehouse level for both locations. For example, a warehouse-to-warehouse transfer created in WMS. This includes the receipt of such a transfer using this service.

Wrong store receiving is not supported for franchise transactions.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/receipts/manage

Input Payload Details

Manage - Object. See list of elements for detail

Items - Object. See list of elements for detail

Receipts - Object. See list of elements for detail

ReceiptDetails - Object. See list of elements for detail

Cartons - Object. See list of elements for detail

ReceiptOverages - Object. See list of elements for detail

Details - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Manage Purchase Order Receipts

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service is used to process receipts within Merchandising against purchase orders. The receipt gets processed against the matching shipment if it already exists within Merchandising whereas a new shipment record and receipt are created if it does not find a matching shipment. Purchase orders can be received only at an item level using this service.

When a PO is received at a location, Merchandising will update the appropriate tables, including the shipment and purchase order, stock on hand, as well as other inventory buckets (e.g., in-transit). A record of the receipt is also made in the transaction level stock ledger.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/purchaseOrder/receipts/manage

Input Payload Details

Table 5-301 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of receipt records.

Table 5-302 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentNo	No	Number (9)	Unique identifier for receiving a group of items.
receipts	Yes	Collection of Object	References a collection of purchase order receipts.

Table 5-302 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
receiptOverages	No	Collection of Object	Contains the Receipt Overage Information. This optional overage node is ignored by MFCS. This overage group is used by Retail Fiscal Management system to differentiate whether an item or item quantity was not received due to specific reason codes such as damaged for example.

Table 5-303 Receipts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	Indicates the location that has processed the receipt. For stores, this will be the store ID. For warehouses, this will be the physical warehouse ID.
orderNo	No	Number (12)	Specified the unique identifier of the purchase order, allocation or transfer.
referenceDocumentNo	No	Number (10)	Contains a reference number for a document associated to the shipment (e.g., Fiscal Document ID for Brazilian based transactions).
asnNo	No	String (30)	Used to relate the receipt message to the previous ASN message. This field is required for transfers and allocations.
receiptDetails	No	Collection of Object	References a collection of receipt details.

Table 5-304 ReceiptDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Specifies the item on the allocation, purchase order, or transfer that has been received.
receivedQuantity	Yes	Number (12,4)	Contains the quantity received for the allocation, purchase order, or transfer in the standard unit of measure.
receiptTransactionType	Yes	String (1)	Specifies whether the receipt detail line item is for a 'R'eceipt, 'T'rانشipment, or 'A'djustment. For purchase order receiving, Transshipments are not allowed. For stock order (ie. Allocation and Transfer) receiving, Transshipments are handled the same way as 'R'eceipt type.

Table 5-304 (Cont.) ReceiptDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
receiptDate	No	date	Identifies the date on which the transaction was received.
externalReferenceNo	Yes	String (17)	An externally generated identifier corresponding to the receipt of the item at the location.
distroLocation	No	Number (10)	Only used for purchase order receipt, when the purchase order has an allocation attached to it. This element specifies the location to which the allocation is being sent.
containerId	No	String (30)	Identifies the carton number for shipments originating from the Advance Shipment Notification process as carton shipments.
distroNo	No	Number (12)	Only used for purchase order receipts, when the purchase order has an allocation attached to it. This element contains the allocation id.
distroDocumentType	No	String (1)	Only used for purchase orders, when the purchase order has an allocation attached to it. When populated, this value should always be A to specify an allocation.
toDisposition	No	String (4)	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.
fromDisposition	No	String (4)	This value is used to determine inventory availability. Valid values are in the INV_STATUS_CODES table. The from disposition is used when the to disposition is not provided.
unitCost	No	Number (20,4)	Used for purchase order receipts when configured for Global Tax (GTS) where the location's country is localized. Otherwise, it is ignored except in scenarios where there is an unexpected item on the receipt, in which case it can be optionally included to indicate the receipt cost of the item. If not included for unexpected receipts, the default supplier cost will be used. Cost is converted to the order's currency before insert/update.
shippedQuantity	No	Number (12,4)	Only used for doc_type of 'P'. Updates the number of items expected to be received, originally set by the ASN process.

Table 5-304 (Cont.) ReceiptDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
weight	No	Number (12,4)	Contains the actual weight of the item received for the shipment. This will be included for some catch weight items.
weightUom	No	String (4)	Contains the unit of measure of the received weight (e.g., pounds, kilograms) where UOM class is of type MASS. Weight and Weight UOM must both be populated, or both must be NULL.
grossCost	No	Number (20,4)	Contains the Unit cost and Expenses incurred on an item in a particular transaction.
itemLineNo	No	Number (10)	This field indicates the item line number from customer orders.
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during receipt consume at item level. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on inventory transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Table 5-305 ReceiptOverages - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	No	String (12)	Specifies the unique identifier of the purchase order.
details	No	Collection of Object	References a collection of Overage Receipt Detail Information

Table 5-306 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Unique identifier for the item.
overageQuantity	No	Number (12,4)	Contains the quantity of the item received at the location as overage, damaged etc.
reasonCode	No	Number (4)	Contains the rejection reason code or overage reason code.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "appointmentNo": 1,
      "receipts": [
        {
          "location": 1211,
          "orderNo": 545005,
          "referenceDocumentNo": 531121,
          "asnNo": "PO_54005_2",
          "receiptDetails": [
            {
              "item": "106400029",
              "receivedQuantity": 3,
              "receiptTransactionType": "R",
              "receiptDate": "2001-12-31",
              "externalReferenceNo": "Receipt_531121",
              "distroLocation": null,
              "containerId": null,
              "distroNo": 100000055001,
              "distroDocumentType": null,
              "toDisposition": "ATS",
              "fromDisposition": "ATS",
              "unitCost": 10,
              "shippedQuantity": 5,
              "weight": 5,
              "weightUom": "KG",
              "grossCost": 10.5,
              "itemLineNo": null,
              "inventoryIdentifierType": "L",
              "inventoryId": "5346"
            }
          ]
        }
      ]
    }
  ],
  "receiptOverages": [
    {
      "orderNo": "545005",
      "details": [
        {
          "item": "106400029",
          "overageQuantity": 24,
          "reasonCode": 3
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Manage Stock Order Receipts

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service processes receipts that Merchandising receives against transfers, and allocations received at a store, warehouse or finisher. Transfers and allocations, collectively referred to as stock orders, may be received at the bill of lading (BOL) level, where the entire shipment is received without including details, carton level, or item level.

When a transfer or allocation is received at a location, Merchandising will update the appropriate tables, including the shipment, the transfer or allocation, stock on hand, as well as other inventory buckets (e.g., in-transit). A record of the receipt is also made in the Stock Ledger.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/stockOrder/receipts/
manage

Input Payload Details

Table 5-307 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of receipt records.

Table 5-308 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentNo	No	Number (9)	Unique identifier for receiving a group of items.
receipts	Yes	Collection of Object	References a collection of transfer receipts.

Table 5-309 Receipts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	Indicates the location that has processed the receipt. For stores, this will be the store ID. For warehouses, this will be the physical warehouse ID.
distroNo	No	Number (12)	Specified the unique identifier of the purchase order, allocation or transfer.
documentType	Yes	String (1)	Specified whether the receipt is for an allocation 'A', purchase order 'P', or transfer 'T'. Types 'D' and 'V' are also valid document types that are treated the same as transfers.
asnNo	No	String (30)	Used to relate the receipt message to the previous ASN message. This field is required for transfers and allocations.
receiptType	No	String (2)	This field is used in receiving transfers or allocations to determine if the receipt is at the BOL level (BL) or SKU level (SK). If not provided, the value will be defaulted to SK. It is not used for a PO receipt. This field is used in receiving transfers or allocations to determine if the receipt is at the BOL level (BL) or SKU level (SK). If not provided, the value will be defaulted to SK. It is not used for a PO receipt.
fromLocation	No	Number (10)	Contains the source location of the shipment.
fromLocationType	No	String (1)	Contains the from location type of the shipment. Valid values are W - Warehouse, E - External Finisher and S - Store.

Table 5-309 (Cont.) Receipts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	No	String (1)	Indicates the status of the ASN received. This field is used only for stock order receiving and a status of C indicates that the entire ASN is will be set to closed.
receiptDetails	No	Collection of Object	References a collection of receipt details.
cartons	No	Collection of Object	References a collection of container details.

Table 5-310 ReceiptDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Specifies the item on the allocation, purchase order, or transfer that has been received.
receivedQuantity	Yes	Number (12,4)	Contains the quantity received for the allocation, purchase order, or transfer in the standard unit of measure.
receiptTransactionType	Yes	String (1)	Specifies whether the receipt detail line item is for a 'R'eceipt, 'T'ransshipment, or 'A'djustment. For purchase order receiving, Transshipments are not allowed. For stock order (ie. Allocation and Transfer) receiving, Transshipments are handled the same way as 'R'eceipt type.
receiptDate	No	date	Identifies the date on which the transaction was received.
externalReferenceNo	Yes	String (17)	An externally generated identifier corresponding to the receipt of the item at the location.
containerId	No	String (30)	Identifies the carton number for shipments originating from the Advance Shipment Notification process as carton shipments.
refContainerId	No	String (30)	Identifies the original carton number the item was shipped under, if it was being received at the wrong destination. This is required by Merchandising to process the updates correctly, as SIOCS reassigns the container ID at the receiving location.
toDisposition	No	String (4)	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.

Table 5-310 (Cont.) ReceiptDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromDisposition	No	String (4)	This value is used to determine inventory availability. Valid values are in the INV_STATUS_CODES table. The from disposition is used when the to disposition is not provided.
weight	No	Number (12,4)	Contains the actual weight of the item received for the shipment. This will be included for some catch weight items.
weightUom	No	String (4)	Contains the unit of measure of the received weight (e.g., pounds, kilograms) where UOM class is of type MASS. Weight and Weight UOM must both be populated, or both must be NULL.
dummyCartonInd	No	String (1)	Indicates if this carton is a dummy carton. This field is only used for transfer receipts.
tamperedCartonInd	No	String (1)	Indicates if the carton was tampered. This field is only used for transfer receipts.
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during receipt consume at item level. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on inventory transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Table 5-311 Cartons - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cartonStatus	No	String (1)	Denotes the status of the carton received. It can be Actual (A), Dummy (D), Overage (O), or Closed (C). This field is used only for stock order receiving. A status of C indicates that the entire container is missing and the missing stock will be reconciled in RMS.
containerId	No	String (30)	Contains the carton ID for the shipment.

Table 5-311 (Cont.) Cartons - Object. See list of elements for detail

Element Name	Required	Data Type	Description
receiptTransactionType	Yes	String (1)	Specifies whether the receipt detail line item is for a 'R'eceipt, 'T'ransshipment, or 'A'djustment. For purchase order receiving, Transshipments are not allowed. For stock order (ie. Allocation and Transfer) receiving, Transshipments are handled the same way as 'R'eceipt type.
receiptDate	No	date	This field contains the date on which the carton is received.
externalReferenceNo	No	String (17)	This field holds the externally generated number when the shipment is received at the location.
toDisposition	No	String (4)	This value is used to determine if the inventory is available or unavailable, based on the code's INV_STATUS value on the INV_STATUS_CODES table.
weight	No	Number (12,4)	Contains the actual weight of the item received for the shipment. This will be included for some catch weight items.
weightUom	No	String (4)	Contains the unit of measure of the received weight (e.g., pounds, kilograms) where UOM class is of type MASS. Weight and Weight UOM must both be populated, or both must be NULL.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "appointmentNo": 2,
      "receipts": [
        {
          "location": 1111,
          "distroNo": 100000055001,
          "documentType": "T",
          "asnNo": "ASN_11223344_2",
          "receiptType": "BL",
          "fromLocation": 2,
          "fromLocationType": "W",
          "status": null,
          "receiptDetails": [
            {
              "item": "104100022",
              "receivedQuantity": 3,
```

```

        "receiptTransactionType": "R",
        "receiptDate": "2001-12-31",
        "externalReferenceNo": "ERN_10091",
        "containerId": "CTN_1_ASN_11223344_2",
        "refContainerId": null,
        "toDisposition": "ATS",
        "fromDisposition": "ATS",
        "weight": 3,
        "weightUom": "KG",
        "dummyCartonInd": "Y",
        "tampedCartonInd": "Y",
        "inventoryIdentifierType": "L",
        "inventoryId": "5346"
    }
],
"cartons": [
    {
        "cartonStatus": "A",
        "containerId": "CTN_1_ASN_11223344_2",
        "receiptTransactionType": "R",
        "receiptDate": "2001-12-31",
        "externalReferenceNo": "ERN_10091",
        "toDisposition": "ATS",
        "weight": 3,
        "weightUom": "KG"
    }
]
}
]
}
]
}
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
    }
  ]
}

```

```

        "inputValue": "X"
    }
  ]
}

```

Get Stock Order Shipments Created in Merchandising

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service publishes stock-order shipment data to downstream consuming systems. It publishes shipment transactions together with shipment lines, physical from and to fulfillment locations, customer-order and fulfillment-order references, and the published base cost. Functionally, the payload only includes shipment lines whose SHIPSKU.DISTRO_TYPE is A or T, derives fromLocation and toLocation as physical fulfillment locations, uses ORDCUST for customer-order and fulfillment-order references, and uses V_ITEM_SUPP_COUNTRY_LOC.BASE_COST for baseCost.

This service can be used to publish details of stock order (transfer or allocation) shipments that were created within the Merchandising solution, including two-legged transfers involving Finisher locations. If one or more filters of the available filters, (From Location, From Location Type, To Location, To Location Type, and/or Bill of Lading (BOL) Number), are present as input parameters, the stock order shipments published will be those matching the provided filter values.

In some retail implementations, the management of stores or warehouses are done by third party systems that are not designed to efficiently handle shipments related to vital processes such as transfers and allocations on a regular basis. For these store or warehouse locations, subscribing to this service will allow these locations to be updated with the latest inventory movement information.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: SHIPMENT, SHIPSKU

Additional business tables: ORDCUST, V_ITEM_SUPP_COUNTRY_LOC, STORE, WH, PARTNER, TSFHEAD, ALLOC_HEADER

JSON cache table: MERCHAPI_EXT_SHIPMENT

JSON generation view: not applicable; cache rows are populated directly by package logic

The JSON payload is built from SHIPMENT, SHIPSKU, ORDCUST, V_ITEM_SUPP_COUNTRY_LOC, and derived location mapping from STORE, WH, PARTNER, TSFHEAD, and ALLOC_HEADER.

The view only publishes shipment lines whose SHIPSKU.DISTRO_TYPE in ('A','T'), derives fromLocation and toLocation as physical fulfillment locations, uses ORDCUST for customer-order and fulfillment-order references, and uses V_ITEM_SUPP_COUNTRY_LOC.BASE_COST for the published baseCost.

This API is effectively delta-only.

Initial Seeding and post enablement rebuild/repair

This API has no initial load or rebuild; the refresh step only performs API-disable handling for the cache table.

ICL (Integration Change Log) table: ICL_RMS_SHIPMENT (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on SHIPMENT and SHIPSKU.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_SHIPMENT

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_SHIPMENT_ADHOC_PROCESS ->
MERCHAPI_EXT_SHIPMENT_ADHOC_JOB

Webhook configuration api name: inventory/shipmentsAndReceipts/stockOrder/shipment

Special handling

Additional GET query parameters: fromLoc, toLoc, fromLocType, toLocType, bolNo.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/stockOrder/shipment

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
fromLoc	No	Number	From Location
toLoc	No	Number	To Location
fromLocType	No	String	From Location Type
toLocType	No	String	To Location Type
bolNo	No	String	Bill of Lading (BOL)
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Response Code: 200 (Success)

Table 5-312 PageResultsItemsMerchApiShipment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-313 MerchApiShipment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Shipment details.
shipmentNo	No	Number(12,0)	This field specifies a number that uniquely identifies the shipment within the system.
toLocation	No	Number(10,0)	This field specifies the destination location of the shipment.

Table 5-313 (Cont.) MerchApiShipment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocationType	No	String (1)	This field specifies the location type of the destination location of the shipment.
fromLocation	No	Number(10,0)	This field specifies the source location of the shipment.
fromLocationType	No	String (1)	This field specifies the location type of the source location of the shipment.
bolNo	No	String (30)	This field specifies the Bill of Lading linked with the shipment.
shipDate	No	date	This field specifies the date on which the stock order was shipped.
cartonQuantity	No	Number(6,0)	This field specifies the number of cartons associated with the shipment.
carrierCode	No	String (250)	This field specifies the carrier associated with the shipment.
comments	No	String (2000)	This field specifies any comments associated with the shipment.
createDateTime	No	dateTime	This column holds the record creation date.
updateDateTime	No	dateTime	This column holds the date when the record was last updated.
details	No	Collection of Object	This List holds attributes for details of shipment record.
cacheTimestamp	No	dateTime	This field specifies date and time when the shipment record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies date and time when the record was first created for web service publication.

Table 5-314 ShipmentDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
distroNo	No	Number(12,0)	This field specifies the transfer/allocation number associated with the shipment.
distroDocumentType	No	String (1)	This field specifies the type of document associated with the shipment.
customerOrderNo	No	String (48)	This field specifies the linked customer order number generated by Order Management System, if applicable. This field will be NULL until Merchandising UI allows shipment of customer order transfers.
fulfillmentOrderNo	No	String (48)	This field specifies the fulfillment order number linked with the customer order number. This field will be NULL until Merchandising UI allows shipment of customer order transfers.
cartons	No	Collection of Object	This List holds attributes for details of cartons linked to the shipment.

Table 5-315 ShipmentDetailsCarton - Object. See list of elements for detail

Element Name	Required	Data Type	Description
carton	No	String (30)	This field specifies carton number associated with the shipment.
items	No	Collection of Object	This List holds attributes for items present in the specific carton.

Table 5-316 ShipmentDetailsCartonItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.
unitQuantity	No	Number(12,4)	This field specifies the number of items expected to be received for the item/shipment combination.
unitCost	No	Number(20,4)	This field specifies the unit cost associated with the item excluding any landed cost components.
baseCost	No	Number(20,4)	This field specifies the base cost (BC) from RFMCS.
weight	No	Number(12,4)	This field specifies the actual weight of the item.
weightUom	No	String (4)	This field specifies the unit of measure associated with the weight.
createDateTime	No	dateTime	This column holds the record creation date.
updateDateTime	No	dateTime	This column holds the date when the record was last updated.

Table 5-317 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
```

```

{
  "action": "INSERT",
  "shipmentNo": 1,
  "toLocation": 6000,
  "toLocationType": "S",
  "fromLocation": 10001,
  "fromLocationType": null,
  "bolNo": null,
  "shipDate": "2001-12-31",
  "cartonQuantity": 5,
  "carrierCode": "FDX",
  "comments": null,
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z",
  "details": [
    {
      "distroNo": 1,
      "distroDocumentType": "T",
      "customerOrderNo": null,
      "fulfillmentOrderNo": null,
      "cartons": [
        {
          "carton": null,
          "items": [
            {
              "item": null,
              "unitQuantity": 1.0,
              "unitCost": 1.0,
              "baseCost": 1.0,
              "weight": 1.0,
              "weightUom": null,
              "createDateTime": "2001-12-31T23:59:59.000Z",
              "updateDateTime": "2001-12-31T23:59:59.000Z"
            }
          ]
        }
      ]
    }
  ],
  "cacheTimestamp": "2001-12-31T23:59:59.000Z",
  "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
{
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	Yes	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_SHIPMENT	Yes	Yes	Yes	Yes
MERCHAPI_EXT_SHIPMENT	Yes	Yes	No	Yes
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
STORE	Yes	No	No	No
WH	Yes	No	No	No
PARTNER	Yes	No	No	No
V_ITEM_SUPP_COUNTRY_LOC	Yes	No	No	No
TSFHEAD	Yes	No	No	No
ALLOC_HEADER	Yes	No	No	No
ORDCUST	Yes	No	No	No
V_MERCHAPI_EXT_SHIPMENT_JSON	Yes	No	No	No

Update DSD Receipts

Functional Area

Inventory - Shipments and Receipts

Business Overview

This service allows to modify an existing direct store delivery (DSD) receipt within Merchandising. For more information about the DSD receipts, see the Create DSD Receipts service.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/shipmentsAndReceipts/dsd/update

Input Payload Details

Table 5-318 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of DSD Receipts.

Table 5-319 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	This field contains the unique identifying number for a supplier for the receipt.
store	Yes	Number (10)	This field contains the location where the items were delivered.
externalReceiptNo	No	String (17)	This field holds the external transaction sequence number for the receipt.
receiptDate	No	date	This field contains the date of the receipt.
details	No	Collection of Object	References a collection of DSD Receipt detail records.

Table 5-320 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains the item in the receipt. The item must be an approved item that is at transaction level or above.
quantityReceived	Yes	Number (12,4)	This field contains the number of items received for the item/shipment combination. The value must be greater than 0.
unitCost	No	Number (20,4)	This field contains the cost of the item from the supplier/origin country in the supplier's currency.
weight	No	Number (12,4)	This field contains the weight of the item in the receipt, if it is a catch weight item.
weightUom	No	String (4)	This field contains the unit of measure of the received weight.
inventoryIdentifierType	No	String (6)	This inventory identifier type field is passed during DSD receipt message. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	Holds the inventory identifier provided on inventory transaction messages. It is a free-form text ID field that will be interfaced and displayed as a text field.

Sample Input Message

```
{
  "items": [
    {
```



```

    "supplier": 2400,
    "store": 1531,
    "externalReceiptNo": "RECREF1531",
    "receiptDate": "2001-12-31",
    "details": [
      {
        "item": "100750001",
        "quantityReceived": 3,
        "unitCost": 9.65,
        "weight": 3,
        "weightUom": null,
        "inventoryIdentifierType": "L",
        "inventoryId": "5346"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Stock Counts

The following services are included in this functional area:

- [Stock Count Download Service](#)
 - [Get Stock Count Detail](#)
- [Stock Count Upload Service](#)

- [Create Stock Count Schedules](#)
- [Delete Stock Count Schedule Locations](#)
- [Delete Stock Count Schedules](#)
- [Update Stock Count Schedules](#)

Stock Count Download Service

The following services are included in this functional area:

- [Get Stock Count Detail](#)

Get Stock Count Detail

This section describes the Stock Count Detail service.

Business Overview

Stock Count Detail service allows user to retrieve open stock count details for a given item and/or store.

Service Type

Get

ReST URL

```
StockCount/stockCountDetail?
cycleCount={cycleCount}&locationType={locationType}&location={location}&item={
item}&stocktakeDate={stocktakeDate}&pageSize={pageSize}&pageNumber={pageNumber
}
```

Input Parameters

Parameter Name	Required	Description
Item	No	Item
Location	No	Location
Location Type	No	Location Type
Cycle Count	No	Cycle Count
Stocktake Date	No	Stocktake Date (always optional)

Output

RestStockCountRecRDO

Parameter Name	Data Type
cycleCount	BigDecimal
cycleCountDesc	String
stocktakeDate	Timestamp
stocktakeType	String
stakeSkuLoc	List<RestStakeSkuLocRecRDO>

RestStakeSkuLocRecRDO

Parameter Name	Data Type
item	String
location	BigDecimal
locType	String
snapshotOnHandQty	BigDecimal
snapshotInTransitQty	BigDecimal
snapshotUnitCost	BigDecimal
snapshotUnitRetail	BigDecimal
processed	String
physicalCountQty	BigDecimal
packCompQty	BigDecimal
inTransitAmt	BigDecimal
depositItemType	String
xformItemType	String
distributeQty	BigDecimal

JSON Structure

```
{
  "cycleCount":null,
  "cycleCountDesc":null,
  "stocktakeDate":null,
  "stocktakeType":null,
  "stakeSkuLoc": [
    {
      "item":null,
      "location":null,
      "locType":null,
      "snapshotOnHandQty":null,
      "snapshotInTransitQty":null,
      "snapshotUnitCost":null,
      "snapshotUnitRetail":null,
      "processed":null,
      "physicalCountQty":null,
      "packCompQty":null,
      "inTransitAmt":null,
      "depositItemType":null,
      "xformItemType":null,
      "distributeQty":null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

```

    }
  }

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
STAKE_HEAD	Yes	No	No	No
STAKE_SKU_LOC	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Stock Count Upload Service

The following services are included in this functional area:

- [Create Stock Count Schedules](#)
- [Delete Stock Count Schedule Locations](#)
- [Delete Stock Count Schedules](#)
- [Update Stock Count Schedules](#)

Create Stock Count Schedules

Functional Area

Inventory - Stock Counts

Business Overview

Stock count schedules are published by an inventory sub-system, such as Oracle Retail Store Inventory and Operations Cloud Service (SIOCS), to communicate unit and value stock count schedules to Merchandising. This stock count schedule data helps in synchronizing the inventory values of the integrated system and Merchandising. This integrated system then performs a physical inventory count and uploads the results, and Merchandising compares the discrepancies.

This webservice allows the external systems to create Unit and Value stock count requests within Merchandising. The count is assumed to be for the full location, unless any department, class or subclass details are included.

When a new stock count request is created, this webservice will validate all the required fields that are present in the message which includes a description, date, location type and location details. The stock take type will always be B (both unit and dollar). Optionally, the merchandise hierarchy information can also be included, but, if not included, it will be assumed the entire location will be counted. After the required field and business validations are successfully completed, the stock counts will be created in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/stockCount/schedules/create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-321 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of stock count schedule records.

Table 5-322 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cycleCount	Yes	Number (8)	Contains the number which uniquely identifies the stock or cycle count.
description	Yes	String (250)	This field contains a description of the cycle or stock count which, along with the cycle count number, identifies the cycle or stock count.
locationType	Yes	String (1)	This field contains an indicator which identifies whether the cycle count will be for Stores or Warehouses. Valid values are Store (S) and Warehouse (W).
stocktakeDate	Yes	date	Contains the date on which the stock or cycle count even will take place.
merchandiseHierarchy	No	Collection of Object	References a collection of merchandies hierarchies to be included in the stock count. If not present, all departments will be included in the count.
locations	No	Collection of Object	References a collection of locations to be included in the stock count.

Table 5-323 MerchandiseHierarchy - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	This field contains the department number where the cycle count will occur. If the value = -1, the stock count will apply to all departments. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.
class	No	Number (4)	This field contains the class number where the cycle count will occur. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.
subclass	No	Number (4)	This field contains the subclass number where the cycle count will occur. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.

Table 5-324 Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field contains the store or warehouse number on the cycle count. This must be a valid store or a stockholding warehouse in Merchandising.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "cycleCount": 1001,
      "description": "Cycle count 1001",
      "locationType": "S",
      "stocktakeDate": "2001-12-31",
      "merchandiseHierarchy": [
        {
          "dept": 3041,
          "class": 1,
          "subclass": 1
        }
      ],
      "locations": [
        {
          "location": 6000
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
}
```

```

    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}

```

Delete Stock Count Schedule Locations

Functional Area

Inventory - Stock Counts

Business Overview

This webservice allows external systems to delete locations from existing stock counts within Merchandising. This validates if all the required fields are present in the message and after the required field and business validations are completed successfully, the stock count locations will be removed. If all locations in the stock count are deleted, the entire stock count will be marked for deletion.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/stockCount/schedule/locations/delete

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-325 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cycleCount	Yes	Number (8)	Contains the number which uniquely identifies the stock or cycle count.
locations	No	Collection of Object	References a collection of locations to be deleted from the stock count.

Table 5-326 Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field contains the store or warehouse number on the cycle count. This must be a valid store or a stockholding warehouse in Merchandising.

Sample Input Message

```

{
  "cycleCount": 1001,

```

```
"locations": [  
  {  
    "location": 6000  
  }  
]
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Delete Stock Count Schedules

Functional Area

Inventory - Stock Counts

Business Overview

This webservice allows the external systems to delete stock counts from Merchandising. This webservice will validate if all the required fields are present in the message. After required field and business validation are successfully completed, the stock counts will be marked for deletion in Merchandising.

A separate webservice `Delete Stock Count Schedule Locations` supports deleting a location from the count.

For more details on Stock Count Schedules, see the `Create Stock Count Schedule` service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/stockCount/schedules/delete

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-327 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cycleCount	Yes	Number (8)	Contains the number which uniquely identifies the stock or cycle count.

Sample Input Message

```
{
  "cycleCount": 1001
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Stock Count Schedules

Functional Area

Inventory - Stock Counts

Business Overview

This webservice allows external systems to modify Unit and Value stock count requests that are already present within Merchandising. When an existing stock count request is updated, this webservice will validate all the required fields that are present in the message which includes a description, date, location type, and location details. After the required field and business validations are completed successfully, the stock counts will be updated in Merchandising.

For more details on Stock Count Schedules, see the Create Stock Count Schedule service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/stockCount/schedules/update

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-328 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cycleCount	Yes	Number (8)	Contains the number which uniquely identifies the stock or cycle count.
description	Yes	String (250)	This field contains a description of the cycle or stock count which, along with the cycle count number, identifies the cycle or stock count.
locationType	Yes	String (1)	This field contains an indicator which identifies whether the cycle count will be for Stores or Warehouses. Valid values are Store (S) and Warehouse (W).
stocktakeDate	Yes	date	Contains the date on which the stock or cycle count even will take place.
merchandiseHierarchy	No	Collection of Object	References a collection of merchandies hierarchies to be included in the stock count. If not present, all departments will be included in the count.
locations	No	Collection of Object	References a collection of locations to be included in the stock count.

Table 5-329 MerchandiseHierarchy - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	This field contains the department number where the cycle count will occur. If the value = -1, the stock count will apply to all departments. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.
class	No	Number (4)	This field contains the class number where the cycle count will occur. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.
subclass	No	Number (4)	This field contains the subclass number where the cycle count will occur. The dept/class/subclass hierarchy must be a valid hierarchy in Merchandising.

Table 5-330 Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	This field contains the store or warehouse number on the cycle count. This must be a valid store or a stockholding warehouse in Merchandising.

Sample Input Message

```
{
  "cycleCount": 1001,
  "description": "Cycle count 1001",
  "locationType": "S",
  "stocktakeDate": "2001-12-31",
  "merchandiseHierarchy": [
    {
      "dept": 3041,
      "class": 1,
      "subclass": 1
    }
  ],
  "locations": [
    {
      "location": 6000
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Transfers and Allocations

The following services are included in this functional area:

- [Allocation Download Service](#)
 - [Get Allocation Details for Provided Allocation](#)
 - [Get Details for an Allocation](#)
- [Allocation Number Download Service](#)
 - [Get Next Allocation Numbers Service](#)
- [Allocation Upload Service](#)
 - [Close Allocations](#)
 - [Create Allocation Details](#)
 - [Create Allocations](#)
 - [Delete Allocation Details](#)
 - [Update Allocation Details](#)
 - [Update Allocations](#)
- [Allocation Corporate Rules](#)
 - [Manage Corporate Rules](#)
- [Mobile Workflow Transfer Download Service](#)
 - [Create Transfer - Get From Locations for Transfer](#)

- [Create Transfer - Get Items for Transfer](#)
- [Create Transfer - Get Next Transfer Numbers](#)
- [Create Transfer - Get To Locations for Transfer](#)
- [Create Transfer - Refresh Transfer Items](#)
- [Create Transfer - Refresh Transfer Locations](#)
- [Mobile Workflow - Get Locations on Existing Transfers](#)
- [Mobile Workflow - Get Transfer Statuses](#)
- [Mobile Workflow - Get Transfer Summary](#)
- [Mobile Workflow - Get Transfer Types](#)
- [Mobile Workflow - Get Transfer Users](#)
- [Mobile Workflow - Get Transfers](#)
- [Mobile Workflow - Refresh Transfer Locations](#)
- [Mobile Workflow Transfer Upload Service](#)
 - [Create Transfer](#)
 - [Mobile Workflow - Update Transfer Status](#)
- [Stock Order Status Upload Service](#)
 - [Manage Stock Order Status](#)
- [Transfer Detail Service](#)
- [Transfer Download Service](#)
 - [Get Transfer](#)
 - [Get Transfer Details for Provided Transfer](#)
- [Transfer Upload Service](#)
 - [Create Book Transfer](#)
 - [Create Transfer Details](#)
 - [Create Transfers](#)
 - [Delete Transfer Details](#)
 - [Delete Transfers](#)
 - [Update Transfer Details](#)
 - [Update Transfers](#)
- [Work Order Status Upload Service](#)
 - [Update Work Orders Status](#)
- [Work Order Download Service](#)
 - [Purchase Order Work Order Publish Services](#)
 - [Transfer Work Order Publish Services](#)

Get Allocation Details for AIF Applications

Business Overview

The service is used to fetch all allocations from Merchandising for AIF application usage, data warehousing needs, or other downstream consuming systems. This service follows the AIF publish pattern and provides cache-backed JSON messages for consumers that need current allocation data.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and interacts with webhooks, refer to the common Publish API operational guidance.

Core business tables: ALLOC_HEADER, ALLOC_DETAIL

Additional business tables: ORDHEAD, ITEM_MASTER, WH, STORE

JSON cache table: MERCHAPI_AIF_ALLOC

JSON generation view: V_MERCHAPI_AIF_ALLOC_JSON

Builds the allocation header from ALLOC_HEADER and aggregates allocation detail rows from ALLOC_DETAIL, with order, item, warehouse, and store enrichment where applicable.

An allocation is published when it is present in the allocation JSON view and has a cacheable allocation message for AIF downstream consumption.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_AIF_ALLOC_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the allocation no longer exists in V_MERCHAPI_AIF_ALLOC_JSON.

ICL (Integration Change Log) table: ICL_RMS_ALLOC (ICL consumer name = MERCHAPI_4)

ICL entries are created using the delivered entity map and table map for inventory/aif/allocation. Qualifying inserts, updates, and deletes are written with change type U for cache refresh processing.

Batch Configuration. Background-engine execution is the delivered asynchronous option for this API.

Process configuration name: API_AIF_ALLOC

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM batch scheduler based delta processing is not supported for this API.

Webhook configuration api name: inventory/aif/allocation

Special handling:

Additional GET query parameters: allocNo.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime profiles.

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through `API_NO_OF_THREADS`, and commit sizing is controlled through `API_COMMIT_MAX_COUNTER`. Choose these settings together based on expected row volume, average message size, and the SQL cost of building each JSON message.

Dropping indexes and constraints during seeding

`DROP_INDEX_ON_REFRESH` applies only to initial seeding, including truncate-and-load execution. When enabled, indexes and constraints are dropped before seeding and rebuilt after the load completes. `GATHER_STATS_ON_REFRESH` should normally remain enabled unless implementation testing shows a specific reason to change it.

REST fetch considerations

Select REST page size based on average payload size so the average response remains below about 10 MB per call. For full downloads, omit the `since` query parameter. For delta downloads, use a short time window; wider outage or recovery windows should be handled through smaller polling windows, using `optimalBefore` where applicable.

Input Parameters

Response Code: 200 (Success)

`PageResultsItemsMerchApiAifAlloc` - Object. See list of elements for detail

`MerchApiAifAlloc` - Object. See list of elements for detail

`aifAllocDetails` - Object. See list of elements for detail

`LinkItems` - Object. See list of elements for detail

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "allocNo": 10010001,
      "orderNo": 464690012,
      "item": "100000001",
      "allocStatus": "A",
      "poType": "4000",
      "allocMethod": "M",
      "releaseDate": "2001-12-31",
      "warehouse": 1001,
      "allocDesc": "AIF allocation",
      "createDateTime": "2001-12-31T10:15:30Z",
      "updateDateTime": "2001-12-31T10:15:30Z",
    }
  ]
}
```

```

    "details": [
      {
        "toLocation": 101,
        "toLocationType": "S",
        "quantityAllocated": 24,
        "quantityTransferred": 0,
        "quantityCancelled": 0,
        "quantityReceived": 0,
        "inStoreDate": "2001-12-31"
      }
    ]
  },
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": []
}

```

Get Transfer Details for AIF Applications

Business Overview

The service is used to fetch all transfers from Merchandising for AIF application usage, data warehousing needs, or other downstream consuming systems. This service follows the AIF publish pattern and provides cache-backed JSON messages for consumers that need current transfer data.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and interacts with webhooks, refer to the common Publish API operational guidance.

Core business tables: TSFHEAD, TSFDETAIL

Additional business tables: ITEM_MASTER, STORE, WH, PARTNER

JSON cache table: MERCHAPI_AIF_TRANSFER

JSON generation view: V_MERCHAPI_AIF_TRANSFER_JSON

Builds the transfer header from TSFHEAD and aggregates detail rows from TSFDETAIL, with source and destination location enrichment for stores, warehouses, internal finishers, and external finishers.

A transfer is published when it is present in the transfer JSON view and has a cacheable transfer message for AIF downstream consumption.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_AIF_TRANSFER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the transfer no longer exists in V_MERCHAPI_AIF_TRANSFER_JSON.

ICL (Integration Change Log) table: ICL_RMS_TRANSFER (ICL consumer name = MERCHAPI_4)

ICL entries are created using the delivered entity map and table map for inventory/aif/transfer. Qualifying inserts, updates, and deletes are written with change type U for cache refresh processing.

Batch Configuration. Background-engine execution is the delivered asynchronous option for this API.

Process configuration name: API_AIF_TRANSFER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM batch scheduler based delta processing is not supported for this API.

Webhook configuration api name: inventory/aif/transfer

Special handling:

Additional GET query parameters: transferNo, fromLocation, toLocation.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime profiles.

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through API_NO_OF_THREADS, and commit sizing is controlled through API_COMMIT_MAX_COUNTER. Choose these settings together based on expected row volume, average message size, and the SQL cost of building each JSON message.

Dropping indexes and constraints during seeding

DROP_INDEX_ON_REFRESH applies only to initial seeding, including truncate-and-load execution. When enabled, indexes and constraints are dropped before seeding and rebuilt after the load completes. GATHER_STATS_ON_REFRESH should normally remain enabled unless implementation testing shows a specific reason to change it.

REST fetch considerations

Select REST page size based on average payload size so the average response remains below about 10 MB per call. For full downloads, omit the since query parameter. For delta downloads, use a short time window; wider outage or recovery windows should be handled through smaller polling windows, using optimalBefore where applicable.

Input Parameters

Response Code: 200 (Success)

PageResultsItemsMerchApiAifTransfer - Object. See list of elements for detail

MerchApiAifTransfer - Object. See list of elements for detail

aifTransferDetails - Object. See list of elements for detail

LinkItems - Object. See list of elements for detail

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "transferNo": 30030001,
      "fromLocationType": "W",
      "fromLocation": 1001,
      "toLocationType": "S",
      "toLocation": 101,
      "transferType": "MR",
      "deliveryDate": "2001-12-31",
      "transferStatus": "A",
      "orderNo": 464690012,
      "createDateTime": "2001-12-31T10:15:30Z",
      "updateDateTime": "2001-12-31T10:15:30Z",
      "details": [
        {
          "transferSeqNo": 1,
          "item": "100000001",
          "transferQuantity": 24,
          "shipQuantity": 0,
          "receivedQuantity": 0,
          "cancelledQuantity": 0,
          "transferPrice": 12.5,
          "transferCost": 10.0
        }
      ]
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": []
}
```

Update Transfer Price for Intercompany Allocations

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service is used to define and update the transfer price for intercompany allocations prior to the first shipment of an allocation. It enables users to override the default Weighted Average Cost (WAC) and explicitly specify transfer prices at either the allocation level or for specific destination locations within an allocation.

The available inputs for this service are allocation number, default transfer price, clear default transfer price flag, destination location, destination location type, and transfer price. The service processes requests to update transfer prices only for allocations that exist and have not yet undergone shipment processing.

The service provides the ability to accept a collection of allocation records and corresponding destination-level details from an external system or source to update transfer pricing for intercompany movements. Transfer prices can be applied globally across all eligible locations within an allocation or selectively for specific destination locations.

The service will attempt to process the request as a new update request. Success or failure of the operation will be returned as part of the response object.

When a default transfer price is provided at the header level, the service will update all intercompany destination locations within the allocation with the specified value.

When destination location details are provided, the service will update transfer prices only for the specified intercompany locations matching the allocation and location type.

When the clear default transfer price flag is set to 'Y', the service will remove any previously defined transfer prices for all intercompany locations within the allocation, provided the default transfer price is sent as NULL.

The service will only allow updates prior to the first shipment (i.e., when no quantity has been transferred).

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/allocation/transferPrice/update

Input Payload Details

Update - Object. See list of elements for detail

Items - Object. See list of elements for detail

Details - Object. See list of elements for detail

UpdateError - Object. See list of elements for detail

Sample Input Message

```
{
  "collectionSize": 1000,
  "items": [
    {
      "allocationNo": 10002,
      "defaultTransferPrice": 20,
      "clearTransferPrice": "N",
      "details": [
        {
          "toLocation": 9625519,
          "toLocationType": "S",
          "transferPrice": 10
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

UpdateResponse - Object. See list of elements for detail

Allocations - Object. See list of elements for detail

Sample Response Message

```
{
  "allocations": [
    {
      "allocationNo": 10002
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

UpdateError - Object. See list of elements for detail

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "allocationNo": 10002,
      "toLocation": 9625519,
      "toLocationType": "S",
      "errorMessage": [
        "String"
      ]
    }
  ]
}
```

Allocation Download Service

The following services are included in this functional area:

- [Get Allocation Details for Provided Allocation](#)
- [Get Details for an Allocation](#)

Get Allocation Details for Provided Allocation

Endpoint

MerchIntegrations/services/inventory/allocation

MerchIntegrations/services/inventory/allocation/{allocNo}

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service publishes allocation data to downstream consuming systems. It covers allocations that still exist in Merchandising and have at least one publishable destination row, where the destination is either a store or a warehouse on a different physical warehouse from the source; same-physical-warehouse warehouse destinations are excluded. The payload provides the allocation header together with qualifying destination details, related order context, pick window dates, ticket information, and supported pack-component detail.

Merchandising is responsible for communicating allocation information with external systems such as a store inventory system (SIOCS, for example) and a warehouse management system, like Oracle WMS Cloud.

There are several ways in which allocation information can be created in Merchandising:

Via integration with the Allocation Cloud Service

Via Merchandising replenishment, where cross dock orders generate allocations

Via the Allocation Subscription API, where a third-party system can create allocations and send to Merchandising for execution.

Allocations can be created from a virtual warehouse to any type of stockholding location in Merchandising, including other virtual warehouses, and to both company and franchise stores. Allocations include a store type and stockholding indicator at the detail level when allocating to stores, to allow the store and warehouse inventory management system to filter out the data irrelevant to their respective systems. When allocating to a franchise store, the linked franchise orders are not published; only the allocation itself is published. When allocating to another warehouse, the allocation quantities are summed up to the physical warehouse level and the physical warehouse is what is communicated in the integration.

Allocations in Approved, Closed or Deleted status can be retrieved by this service.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ALLOC_HEADER, ALLOC_DETAIL

Additional business tables: WH, ORDHEAD, ITEM_MASTER, ITEM_LOC, STORE, ITEM_TICKET, PACKITEM_BREAKOUT, V_CODE_DETAIL_TL

JSON cache table: MERCHAPI_EXT_ALLOC

JSON generation view: V_MERCHAPI_EXT_ALLOC_JSON

The current JSON view builds the header from ALLOC_HEADER, derives source-warehouse enrichment from WH, uses ORDHEAD and V_CODE_DETAIL_TL to calculate the pick-not-before and pick-not-after dates, derives ticket type from ITEM_TICKET, and builds detail rows from ALLOC_DETAIL, ITEM_LOC, STORE, WH, and PACKITEM_BREAKOUT.

Functionally, it publishes allocations that still exist in Merchandising and have at least one qualifying destination detail row, where the destination is either a store or a warehouse on a different physical warehouse from the source; same-physical-warehouse warehouse destinations are excluded.

The view itself does not apply a direct allocation-status filter, so eligibility is determined by those destination rules, while delta staging begins when an allocation first becomes approved and then continues for qualifying header and detail changes.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_ALLOC_JSON and therefore publish only allocations that still have at least one qualifying destination detail row, where the destination is either a store or a warehouse on a different physical warehouse from the source.

Mark cache rows deleted when the allocation no longer exists in ALLOC_HEADER.

ICL (Integration Change Log) table: ICL_RMS_ALLOC (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on ALLOC_HEADER and ALLOC_DETAIL.

ICL detail publication only when the related header status is A, the destination is publishable, and detail inserts satisfy QTY_TRANSFERRED <= 0.

ICL insert when the allocation is inserted in status A or changes to status A; ICL update when header publication logic fires for status A or C or a release-date change; ICL delete on header delete.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ALLOC

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ALLOCATION_ADHOC_PROCESS ->
MERCHAPI_EXT_ALLOCATION_ADHOC_JOB

Webhook configuration api name: inventory/allocation

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. Because this service builds allocation messages from multiple base and supporting tables, including destination-eligibility logic and pack detail, the available configuration should be validated against the customer production allocation profile, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through `API_NO_OF_THREADS`, commit sizing through `API_COMMIT_MAX_COUNTER`, and long-running initial seeding can also be segmented through `API_EXEC_SECONDS`.

Thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the message from multiple base and supporting tables. Common starting points for large-volume runs are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment.

`API_COMMIT_MAX_COUNTER` should be sized together with the thread setting. Typical values range from about 1,000 rows and can extend to 10,000. If the number of details line for allocation will be very high, start with smaller commit counter and then go up to see best throughput.

`API_EXEC_SECONDS` is mainly relevant for initial seeding. A value of about 30 minutes helps the batch pause and resume cleanly so tuning changes can be applied between runs if needed.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes. `GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a specific reason to change it.

REST fetch considerations

REST page size should be selected based on average payload size so the average response remains below about 10 MB per call. Common page-limit values are 500, 1,000, 5,000, or 10,000 rows and this should be selected based on average number of details lines in the allocation.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned `before` timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/allocation

MerchIntegrations/services/inventory/allocation/{allocNo}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-331 Input Parameter for "MerchIntegrations/services/inventory/allocation/{allocNo}"

Parameter Name	Required	Data Type	Description
allocNo	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-332 PageResultsItemsMerchApiAllocation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-333 MerchApiAllocation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for allocation details.
allocNo	No	Number(10,0)	This field specifies the number that uniquely identifies the allocation within the system.
docType	No	String (1)	This field specifies that this document is of type allocation. This field is defaulted to A.
physicalWarehouse	No	Number(10,0)	This field specifies the number that is used to identify the physical warehouse for a virtual warehouse.
warehouse	No	Number(10,0)	This field specifies the number that uniquely identifies the warehouse.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.

Table 5-333 (Cont.) MerchApiAllocation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pickNotBeforeDate	No	date	This field specifies the first date on which the allocation should be released from the warehouse for delivery to the store locations.
pickNotAfterDate	No	date	This field specifies the last date on which the allocation should be released from the warehouse for delivery to the store locations.
orderType	No	String (9)	This field specifies the type of the order. Allocations created against purchase orders will be marked as PREDIST order types. Allocations created against warehouse stock will be populated with the DEFAULT_ORDER_TYPE from the SYSTEM_OPTIONS table, which can be AUTOMATIC, MANUAL, or WAVE.
orderNo	No	Number(12,0)	This field specifies the order number to which the allocation applies.
orderDocType	No	String (1)	This field specifies the order type and is defaulted to P.
priority	No	Number(1,0)	This field is currently defaulted to 1.
ticketTypeId	No	String (4)	This field uniquely identifies the ticket type that is associated with the item.
contextType	No	String (6)	This field specifies the functional area code to which the allocation relates, for example, Promotions. Valid values are defined in the CNTX code type.

Table 5-333 (Cont.) MerchApiAllocation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contextValue	No	String (25)	This field specifies the value relating to the context type, for example, Promotion Number.
allocStatus	No	String (1)	This field specifies the status of the allocation.
poType	No	String (4)	This field specifies the value associated with the PO_TYPE for the order.
allocMethod	No	String (1)	This field specifies the preferred allocation method, which is used to distribute goods when the stock received at a warehouse cannot immediately fill all requested allocations to stores. Valid values are A - Allocation quantity based, P - Prorate method, and C - Custom.
releaseDate	No	date	This field specifies the date on which the allocation should be released from the warehouse for delivery to the store locations.
documentNo	No	String (30)	This field specifies the ASN or BOL number for an ASN- or BOL-sourced allocation. This will be populated for the product source of the tier-one allocation.
documentType	No	String (5)	This field contains the type of allocation product source. Valid values are ASN, TSF, BOL, or ALLOC. When this field is null, the source is assumed to be a purchase order or warehouse inventory if no purchase order number is provided.

Table 5-333 (Cont.) MerchApiAllocation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sourceParentAllocId	No	Number(15,0)	This field contains the parent allocation identifier from the source system that interfaced the allocation to merchandising after splitting at item level.
allocDesc	No	String (300)	This field specifies the description of the allocation.
commentDesc	No	String (2000)	This field specifies additional information concerning the allocation.
originInd	No	String (6)	This field identifies the origin point of the allocation. Valid values are AIP, EG, RMS, and ALC.
closeDate	No	date	This field specifies the date when the allocation is closed.
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).
details	No	Collection of Object	This list holds attributes for allocation detail records.
cacheTimestamp	No	dateTime	This field specifies the date and time when the allocation record was last picked for publication processing.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-334 AllocationDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
physicaltoLocation	No	Number(10,0)	This field specifies the physical warehouse of a virtual warehouse. The store ID is populated when the to location is a store.
toLocation	No	Number(10,0)	This field specifies the location to which the allocation is being sent.
locationType	No	String (1)	This field specifies the type of location in the location field. Valid values are Store and Warehouse.
storeType	No	String (6)	This field specifies whether the store is a company or franchise store and is not populated in the case of a warehouse.
stockholdingInd	No	String (1)	This field specifies whether the store can hold stock. In a non-multichannel environment, this field is defaulted to Y. It is not populated in the case of a warehouse.
quantityAllocated	No	Number(12,4)	This column contains the total number of items allocated.
price	No	Number(20,4)	This field specifies the unit retail price in the selling unit retail.
sellingUom	No	String (4)	This field specifies the selling unit of measure for an item.
priority	No	Number(1,0)	This field is defaulted to 1.
storeOrderMultiple	No	String (1)	This column contains the multiple in which the item needs to be shipped from a warehouse to the location.
inStoreDate	No	date	This field contains the date to be included in the publication for communication to the warehouse.

Table 5-334 (Cont.) AllocationDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rushFlag	No	String (1)	This field contains the rush flag indicator to be included in the publication for communication to the warehouse.
quantityTransferred	No	Number(12,4)	This column contains the number of items that have already been transferred under this allocation.
quantityPrescaled	No	Number(12,4)	This column contains the allocation quantity (system generated or manually generated) for the item/location before order scaling processing was performed.
quantityDistro	No	Number(12,4)	This column contains the final quantity that the external system plans on filling.
quantitySelected	No	Number(12,4)	This column contains the initial quantity that the external system plans on filling.
quantityCancelled	No	Number(12,4)	This column contains the quantity that was left to be allocated when the line item was cancelled.
quantityReceived	No	Number(12,4)	This column contains the quantity of the item that has been received at the to location on the allocation.
quantityReconciled	No	Number(12,4)	This column contains the quantity received at another location for this allocated item.
poReceivedQty	No	Number(12,4)	This column tracks the quantity received against a cross-docked order. It is only used to distribute unallocated quantities that are received in a physical warehouse.

Table 5-334 (Cont.) AllocationDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nonScaleInd	No	String (1)	This column determines whether the item/location allocation quantity should not be scaled during the order scaling process.
franchiseOrderNo	No	Number(10,0)	This field holds the franchise order number to which this allocation detail line is linked.
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).
detailsTicket	No	Collection of Object	This list holds attributes for allocation detail ticket records.

Table 5-335 AllocationDetailsTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentItem	No	String (25)	This field specifies the alphanumeric value that identifies the component item within the pack.
componentPrice	No	Number(20,4)	This field specifies the unit retail price in the selling unit retail of the component item within the pack.
componentSellingUom	No	String (4)	This field specifies the selling unit of measure for a component item within the pack.

Table 5-336 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-336 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "allocNo": 102881222,
      "docType": null,
      "physicalWarehouse": 1,
      "warehouse": 10001,
      "item": "2147483647",
      "pickNotBeforeDate": "2001-12-31",
      "pickNotAfterDate": "2001-12-31",
      "orderType": null,
      "orderNo": 585036,
      "orderDocType": null,
      "priority": 1,
      "ticketTypeId": "TKT1",
      "contextType": "PROM",
      "contextValue": "79890",
      "allocStatus": "A",
      "poType": null,
      "allocMethod": null,
      "releaseDate": "2001-12-31",
      "documentNo": null,
      "documentType": null,
      "sourceParentAllocId": 225414,
      "allocDesc": "Allocation to 1311.",
      "commentDesc": null,
      "originInd": "EG",
      "closeDate": "2001-12-31",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "details": [
        {
          "physicaltoLocation": 1,
          "toLocation": 6000,
          "locationType": "S",
          "storeType": "C",
          "stockholdingInd": "Y",
          "quantityAllocated": 3,

```



```

"price": 1.0,
"sellingUom": "EA",
"priority": 1,
"storeOrderMultiple": "E",
"inStoreDate": "2001-12-31",
"rushFlag": null,
"quantityTransferred": 1.0,
"quantityPrescaled": 1.0,
"quantityDistro": 1.0,
"quantitySelected": 1.0,
"quantityCancelled": 1.0,
"quantityReceived": 1.0,
"quantityReconciled": 1.0,
"poReceivedQty": 1.0,
"nonScaleInd": null,
"franchiseOrderNo": 1,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"detailsTicket": [
{
"componentItem": null,
"componentPrice": 1.0,
"componentSellingUom": null
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
{
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_ALLOC	Yes	No	Yes	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_EXT_ALLOC	Yes	Yes	Yes	No
V_MERCHAPI_EXT_ALLOC_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

Get Details for an Allocation

This section describes the Allocation Detail service.

Business Overview

Allocation Detail service allows user to retrieve Allocation information for a selected allocation number.

Service Type

Get

ReST URL

Alloc/allocDetail?allocNumber={allocationNumber}

Input Parameters

Parameter Name	Required	Description
allocNumber	Yes	Allocation Number

Output

RestAllocRecRDO

Parameter Name	Data Type
alloc_no	BigDecimal
order_no	BigDecimal
wh	BigDecimal
item	String
status	String
alloc_desc	String
po_type	String
alloc_method	String
release_date	Date
order_type	String
doc	String
doc_type	String
origin_ind	String
close_date	Date
alloc_detail	List<RestAllocDetailRecRDO>

RestAllocDetailRecRDO

Parameter Name	Data Type
to_loc	BigDecimal
to_loc_type	String
qty_transferred	BigDecimal
qty_allocated	BigDecimal
qty_prescaled	BigDecimal
qty_distro	BigDecimal
qty_selected	BigDecimal
qty_cancelled	BigDecimal
qty_received	BigDecimal
qty_reconciled	BigDecimal
po_rcvd_qty	BigDecimal
non_scale_ind	String
in_store_date	Date
wf_order_no	BigDecimal
rush_flag	String

JSON Structure

```
[
  {
    "docType": null,
    "allocDetail": [
      {
        "qtyTransferred": null,
        "rushFlag": null,
        "wfOrderNo": null,
        "inStoreDate": null,
        "qtyAllocated": null,
        "nonScaleInd": null,
        "toLoc": null,
        "qtyPrescaled": null,
        "toLocType": null,
        "qtyDistro": null,
        "qtySelected": null,
        "qtyReceived": null,
        "qtyCancelled": null,
        "qtyReconciled": null,
        "poRcvdQty": null,
        "links": [],
        "hyperMediaContent": {
          "linkRDO": []
        }
      }
    ],
    "doc": null,
    "originInd": null,
    "allocNo": null,
    "wh": null,
  }
]
```

```

        "allocMethod": null,
        "allocDesc": null,
        "poType": null,
        "item": null,
        "status": null,
        "orderNo": null,
        "orderType": null,
        "releaseDate": null,
        "closeDate": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    }
]

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	Yes	No	No	No
ALLOC_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Allocation Number Download Service

The following services are included in this functional area:

- [Get Next Allocation Numbers Service](#)

Get Next Allocation Numbers Service

Business Overview

This service retrieves next available allocation numbers from Merchandising. This service takes an optional query parameter of count to retrieve multiple (up to 999) allocation numbers. The count defaults to 1 if the query parameter is not provided.

Service Type

GET

ReST URL

/Alloc/allocNo

Input Parameters

Parameter Name	Required	Description
Count	No	Count (1-999) of allocation numbers to be fetched. Default value is 1.

Output

Parameter Name	Data Type	Details
allocNo	Number	Deprecated (Use allocNumbers) as this service has been enhanced to return multiple allocation numbers. This field will contain the first allocation number from the allocNumbers list.
allocNumbers	Array	Contains list of available allocation numbers.

JSON Structure:

```
{
  "allocNo": 123456,
  "allocNumbers": [123456,123457]
}
```

Table Impact

N/A

Allocation Upload Service

The following services are included in this functional area:

- [Close Allocations](#)
- [Create Allocation Details](#)
- [Create Allocations](#)
- [Delete Allocation Details](#)
- [Update Allocation Details](#)
- [Update Allocations](#)

Close Allocations

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service can be used to close allocations within Merchandising from an external system. After successfully validating the information in the message, the header level record is updated to the Closed (C) status.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/allocations/close

Input Payload Details

Table 5-337 Close - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	References a collection of allocations.

Table 5-338 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
allocationNo	Yes	Number (10)	Contains the unique identifier of the allocation. This should fall within the range of Merchandising IDs already designated for allocations.

Sample Input Message

```
{
  "items": [
    {
      "allocationNo": 102881222
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
}
```

```

    "businessError": [
      "Error message"
    ]
  }

```

Create Allocation Details

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service can be used to create detail level records for one or more allocations within Merchandising from an external application. This would add new destination locations to the items that are already present in existing allocations.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/allocation/details/create

Input Payload Details

Table 5-339 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of allocations.

Table 5-340 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
allocationNo	Yes	Number (10)	Contains the unique identifier of the allocation. This should fall within the range of Merchandising IDs already designated for allocations.
allocationDescription	Yes	String (300)	Contains the user defined description of the allocation.
orderNo	No	Number (12)	Contains the purchase order with which the allocation is associated. Only used if the source for the allocation is a purchase order.
item	Yes	String (25)	Contains the transaction level item that is being allocated.
fromLocation	Yes	Number (10)	Contains the location that is the source of the allocation. This must be a valid stockholding virtual warehouse.

Table 5-340 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
documentNo	No	String (30)	Contains identification number for a transfer, another allocation, bill of landing number (BOL), or advanced shipping notice (ASN) number for a purchase order. This field is populated according to documentation type and indicates where the inventory for the allocation should be sourced. This should be null if the source for the allocation is warehouse inventory or a purchase order.
documentType	No	String (5)	Contains the type of allocation product source. Valid values are ASN, Transfer (TSF), Bill of Lading (BOL), or Allocation (ALLOC). When this is passed in as null, the source is assumed to be a PO or warehouse inventory if not PO number provided.
details	No	Collection of Object	References a collection of allocation details.

Table 5-341 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	Yes	Number (10)	Contains the destination location of the allocation. This must be an active stockholding store or virtual warehouse.
toLocationType	Yes	String (1)	Contains the type of the destination location. Valid values are S (store) and W (warehouse).
quantityAllocated	Yes	Number (12,4)	Contains the allocated quantity of the item for the destination location. When the allocation is being created this value must be a positive integer. If this value is being modified, it will contain the quantity adjusted (positive or negative), rather than an override value.
inStoreDate	No	date	Contains the date the item is to be in store. This date will be included in the Merchandising publication for communication to the warehouse.

Sample Input Message

```
{
  "items": [
    {
      "allocationNo": 102881222,
```



```

    "allocationDescription": "Allocation to 1311.",
    "orderNo": 585036,
    "item": "2147483647",
    "fromLocation": 10001,
    "documentNo": null,
    "documentType": null,
    "details": [
      {
        "toLocation": 6000,
        "toLocationType": "S",
        "quantityAllocated": 3,
        "inStoreDate": "2001-12-31"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Create Allocations

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service can be used to create allocations within Merchandising from an external system. Allocations created or updated using this API include those that are based on warehouse inventory, as well as those using inventory from another transaction such as a purchase order, another allocation, a transfer, or a shipment (Vendor ASN or BOL). When allocating using something other than warehouse inventory, the ID of the transaction must also be included as part of the message.

Allocations can involve both stockholding and non-stockholding locations. This includes the ability to process allocations to both stockholding and non-stockholding company and franchise stores, as well as any stockholding warehouse locations, except internal finishers. If an allocation for a franchise store is received, Merchandising will also create a corresponding franchise order. This API supports multiple types of destination locations (warehouses as well as stores) as part of the detail section within the same message.

The release date for the allocation cannot be earlier than the release date of the document (ASN/BOL/ALLOC/TSF/PO) which it is based on. The requested total allocation quantity for an item cannot exceed the quantity available to allocate from the source location or the document it is based on.

The Oracle Retail Allocation Cloud Service does not use this API to interface allocations to Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/allocations/create

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Details - Object. See list of elements for detail

CreateError - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessErrors" will be present if the payload passes schema validation but exception is caught while processing business logic. The element businessErrorr will be present but deprecated.

CreateError - Object. See list of elements for detail

Delete Allocation Details

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service can be used to remove detail level data from one or more allocations within Merchandising from an external system. On successful validation of the information present in the message, the detail level records are deleted from the database.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/allocation/details/delete

Input Payload Details

Table 5-342 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	References a collection of allocations.

Table 5-343 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
allocationNo	Yes	Number (10)	Contains the unique identifier of the allocation. This should fall within the range of Merchandising IDs already designated for allocations.
details	No	Collection of Object	References a collection of allocation details

Table 5-344 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	Yes	Number (10)	Contains the destination location of the allocation. This must be an active stockholding store or virtual warehouse.
toLocationType	Yes	String (1)	Contains the type of the destination location. Valid values are S (store) and W (warehouse).

Sample Input Message

```
{
  "items": [
    {
      "allocationNo": 102881222,
      "details": [
        {
          "toLocation": 1311,
          "toLocationType": "S"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Update Allocation Details

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service allows an external application modify details on one or more allocations at a time within Merchandising. If modifying an existing location, Merchandising assumes the passed in quantity is an adjustment to the current quantity as opposed to an overwrite. The service verifies the allocation is not in-transit, received, nor in progress and that the quantity does not fall to zero or below.

For more information about allocation, see the Create Allocation Service.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/allocation/details/update

Input Payload Details

Table 5-345 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of allocations.

Table 5-346 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
allocationNo	Yes	Number (10)	Contains the unique identifier of the allocation. This should fall within the range of Merchandising IDs already designated for allocations.
orderNo	No	Number (12)	Contains the purchase order with which the allocation is associated. Only used if the source for the allocation is a purchase order.
item	Yes	String (25)	Contains the transaction level item that is being allocated.
fromLocation	Yes	Number (10)	Contains the location that is the source of the allocation. This must be a valid stockholding virtual warehouse.
documentNo	No	String (30)	Contains identification number for a transfer, another allocation, bill of landing number (BOL), or advanced shipping notice (ASN) number for a purchase order. This field is populated according to documentation type and indicates where the inventory for the allocation should be sourced. This should be null if the source for the allocation is warehouse inventory or a purchase order.
documentType	No	String (5)	Contains the type of allocation product source. Valid values are ASN, Transfer (TSF), Bill of Lading (BOL), or Allocation (ALLOC). When this is passed in as null, the source is assumed to be a PO or warehouse inventory if not PO number provided.
details	No	Collection of Object	References a collection of allocation details.

Table 5-347 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	Yes	Number (10)	Contains the destination location of the allocation. This must be an active stockholding store or virtual warehouse.

Table 5-347 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocationType	Yes	String (1)	Contains the type of the destination location. Valid values are S (store) and W (warehouse).
quantityAllocated	Yes	Number (12,4)	Contains the allocated quantity of the item for the destination location. When the allocation is being created this value must be a positive integer. If this value is being modified, it will contain the quantity adjusted (positive or negative), rather than an override value.
inStoreDate	No	date	Contains the date the item is to be in store. This date will be included in the Merchandising publication for communication to the warehouse.

Sample Input Message

```
{
  "items": [
    {
      "allocationNo": 102881222,
      "orderNo": 585036,
      "item": "2147483647",
      "fromLocation": 10001,
      "documentNo": null,
      "documentType": null,
      "details": [
        {
          "toLocation": 6000,
          "toLocationType": "S",
          "quantityAllocated": 3,
          "inStoreDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Update Allocations

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service can be used to update allocations within Merchandising from an external system. In order to perform an update, the existence of a matching record in the allocation header table is mandatory. Currently, only the allocation description, the release date, and status can be modified at the header level.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/allocations/update

Input Payload Details

Table 5-348 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of allocations.

Table 5-349 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
allocationNo	Yes	Number (10)	Contains the unique identifier of the allocation. This should fall within the range of Merchandising IDs already designated for allocations.
allocationDescription	Yes	String (300)	Contains the user defined description of the allocation.
releaseDate	No	date	Contains the earliest date on which the warehouse should ship the allocation.
contextType	No	String (6)	Contains the actual purpose behind the creation of the allocation in the overall business setup. Valid codes need to be defined in the code type = 'CNTX'. E.g. - Promotion, Repairing, etc.
contextValue	No	String (25)	Contains the value related to the specified context type, wherever applicable. E.g.: Promotion ID when the context type is set as 'Promotion'.
allocStatus	No	String (1)	If specified, needs to be either 'R' (Reserved) or 'A' (Approved).

Sample Input Message

```

{
  "items": [
    {
      "allocationNo": 102881222,
      "allocationDescription": "Allocation to 1311.",
      "releaseDate": "2001-12-31",
      "contextType": "REPAIR",
      "contextValue": "71256",
      "allocStatus": "A"
    }
  ]
}

```


Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Mobile Workflow Transfer Download Service

The following services are included in this functional area:

- [Create Transfer - Get From Locations for Transfer](#)
- [Create Transfer - Get Items for Transfer](#)
- [Create Transfer - Get Next Transfer Numbers](#)
- [Create Transfer - Get To Locations for Transfer](#)
- [Create Transfer - Refresh Transfer Items](#)
- [Create Transfer - Refresh Transfer Locations](#)
- [Mobile Workflow - Get Locations on Existing Transfers](#)
- [Mobile Workflow - Get Transfer Statuses](#)
- [Mobile Workflow - Get Transfer Summary](#)
- [Mobile Workflow - Get Transfer Types](#)
- [Mobile Workflow - Get Transfer Users](#)
- [Mobile Workflow - Get Transfers](#)
- [Mobile Workflow - Refresh Transfer Locations](#)

Create Transfer - Get From Locations for Transfer

This section describes the Search From Location service.

Business Overview

This service retrieves locations applicable for inventory transfer. Location can be searched by either 'S'tore or 'W'arehouse. Then enter a location number, a partial location number, a location description, or a partial location description in the search string.

The locations returned are constrained by the following criteria:

- When search type is warehouse only virtual warehouses are returned.
- Only stockholding location.
- When search type is store then only open stores are returned.
- When items are sent as input then only locations with available inventory are returned.
- When To Location is sent as input then:
 - It cannot be the same as the To Location.
 - When transfer type is Manual Requisition, then only locations with the same Transfer Entity/Set of Books as the To Location are returned in the search results.
 - When the transfer type is Intercompany, then only locations with a different Transfer Entity/Set of Books to the To Location are returned in the search results.
 - Only locations in the same transfer zone are returned in the search results.

Service Type

Get

ReST URL

/Transfer/fromLocation?
locationType={locationType}&searchString={searchString}&tsfType={tsfType}&toLocation={toLocation}&items={items}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description	Valid values
LocationType	Yes	Location type Store or warehouse	S, W
SearchString	Yes	search string for locations Id or Name	NA
tsfType	Yes	Transfer type	IC, MR
toLocation	No	Selected to location ID	NA
items	No	Comma Separated values for selected items	NA
PageSize	No	Maximum number of locations to retrieve per page	NA
PageNumber	No	Result page to retrieve	NA

Output

TsfLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locType	String
locName	String
locCurrencyCode	String
entity	BigDecimal
entityDesc	String
tsfLocitemSearchRes	List<TsfLocitemSearchResRDO>

TsfLocitemSearchResRDO

Parameter Name	Data Type
item	String
availQty	BigDecimal
averageCost	BigDecimal
unitRetail	BigDecimal
currencyCode	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "links": [],
  "totalRecordCount": 1,
  "results": [
    {
      "links": [],
      "location": 5991,
      "locType": "S",
      "locName": "DIT Company Stockholding Store",
      "locCurrencyCode": "USD",
      "entity": 1000,
      "entityDesc": "Regular Stores",
      "tsfLocitemSearchRes": [
        {
          "links": [],
          "item": "100054006",
          "availQty": 100,
          "averageCost": 0,
          "unitRetail": 181.82,
          "currencyCode": "USD",
          "hyperMediaContent": {
```

```

        "linkRDO": []
      }
    },
    {
      "links": [],
      "item": "100040051",
      "availQty": 998,
      "averageCost": 1,
      "unitRetail": 1.54,
      "currencyCode": "USD",
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
},
"hyperMediaContent": {
  "linkRDO": []
}
}
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	Yes	No	No	No
ALLOC_DETAIL	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
MV_LOC_SOB	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORG_UNIT	Yes	No	No	No
STORE	Yes	No	No	No
TRANSFER_LOC	Yes	No	No	No
TSF_ENTITY	Yes	No	No	No
V_STORE	Yes	No	No	No
V_TRANSFER_FROM_LOC	Yes	No	No	No
V_TRANSFER_TO_LOC	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Transfer - Get Items for Transfer

Business Overview

This service retrieves items applicable for inventory transfer. Item can be searched either by Item or VPN. To search the item, enter an item number, a partial item description, or a VPN in the search string.

- When search type is ITEM, the search string can be an item number, a partial item number, an item description, or partial item description. In this case, the query returns all items which match the item description or partial description, or which match the item number entered.
- When search type is VPN, the search string can be a VPN or partial VPN, the API should return all items with that VPN.

The items returned are constrained by the following criteria:

- Approved status.
- Transaction-level items.
- Inventory items.
- When From Location is sent as an input, then only the following items are returned:
 - With available inventory at the From Location.
 - Packs with Receive as Type as Each are filtered out when, from location is a virtual warehouse.
- If the System Option for DEPT_LEVEL_TRANSFERS is set as "Y" and a Department ID is sent as input, then only the input department items are returned.

Service Type

GET

ReST URL

/Transfer/item?
itemSearchType={itemSearchType}&searchString={searchString}&dept={dept}&fromLocation={fromLocation}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description	Valid values
itemSearchType	Yes	Search type item or VPN.	ITEM, VPN
searchString	Yes	Search string for items ID or Name.	NA
dept	No	Selected items' department ID.	NA
fromLocation	No	Selected from location ID.	NA
PageSize	No	Maximum number of items to retrieve per page.	NA
PageNumber	No	Result page to retrieve.	NA

Output

Table 5-350 TsflitemSearchRDO

Parameter Name	Data Type
item	String
itemDesc	String

Table 5-350 (Cont.) TsflitemSearchRDO

Parameter Name	Data Type
dept	BigDecimal
availQty	BigDecimal
averageCost	BigDecimal
unitRetail	BigDecimal
currencyCode	String
standardUnitOfMeasure	String
suppPackSize	BigDecimal
innerPackSize	BigDecimal
itemImageUrl	String

Table 5-351 PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "links": [
    {
      "href": "/Transfer/item?
itemSearchType=ITEM&searchString=Black&fromLocation=363640301&pageSize=1&pageN
umber=3",
      "rel": "next",
      "type": "GET",
      "methodType": null
    },
    {
      "href": "/Transfer/item?
itemSearchType=ITEM&searchString=Black&fromLocation=363640301&pageSize=1&pageN
umber=1",
      "rel": "prev",
      "type": "GET",
      "methodType": null
    }
  ],
  "totalRecordCount": 51,
  "results": [
    {
      "links": [],
      "item": "100001406",
      "itemDesc": "DIT Test 11:Black:Extra Small",
      "dept": 1102,
      "availQty": 100,
```

```

        "averageCost": 5,
        "unitRetail": 7.26,
        "currencyCode": "USD",
        "standardUnitOfMeasure": "CKG",
        "suppPackSize": 1,
        "innerPackSize": 1,
        "itemImageUrl": null,
        "hyperMediaContent": {
            "linkRDO": []
        }
    },
    ],
    "hyperMediaContent": {
        "linkRDO": [
            {
                "href": "/Transfer/item?
itemSearchType=ITEM&searchString=Black&fromLocation=363640301&pageSize=1&pageN
umber=3",
                "rel": "next",
                "type": "GET",
                "methodType": null
            },
            {
                "href": "/Transfer/item?
itemSearchType=ITEM&searchString=Black&fromLocation=363640301&pageSize=1&pageN
umber=1",
                "rel": "prev",
                "type": "GET",
                "methodType": null
            }
        ]
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	Yes	No	No	No
ALLOC_DETAIL	Yes	No	No	No
DAILY_PURGE	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
ITEM_IMAGE	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDHEAD	Yes	No	No	No
STORE	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Transfer - Get Next Transfer Numbers

Business Overview

This service retrieves the next available transfer numbers from Merchandising. This service takes an optional query parameter of count to retrieve multiple (up to 999) transfer numbers. The count defaults to 1 if the query parameter is not provided.

Service Type

GET

ReST URL

/Transfer/transferId

Input Parameters

Parameter Name	Required	Description
count	No	Count (1-999) of transfer numbers to be fetched. Default value is 1.

Output

Parameter Name	Data Type	Details
transfer_no	Number	Deprecated (Use transferNumbers) as this service has been enhanced to return multiple transfer numbers. This field will contain the first transfer number from transferNumbers list.
transferNumbers	Array	Contains list of available transfer numbers.

JSON Output

```
{
  "transfer_no": 123456,
  "transferNumbers": [123456, 123457]
}
```

Table Impact

N/A

Create Transfer - Get To Locations for Transfer

This section describes the Search To Location service.

Business Overview

This service retrieves locations applicable for inventory transfer. Location can be searched by either 'S'tore or 'W'arehouse. Then enter a location number, a partial location number, a location description, or a partial location description in the search string.

The locations returned are constrained by the following criteria:

- When search type is warehouse only virtual warehouses are returned.
- Internal finishers are filtered out.
- Only stockholding location.
- When search type is Store then only open stores are returned.
- When items are sent as input then only locations with available inventory are returned.
- When From Location is sent as input then:
 - To Location cannot be the same as the From Location.
 - When Transfer Type is set as a manual request, then only locations with the same Transfer Entity/Set of Books as the From Location are returned in the search results.
 - When the Transfer Type is Intercompany, then only locations with a different Transfer Entity/Set of Books to the From Location are returned in the search results.
 - Only locations in the same transfer zone are returned in the search results.

Service Type

Get

ReST URL

```
/Transfer/toLocation?
locationType={locationType}&searchString={searchString}&tsfType={tsfType}&from
Location={fromLocation}&pageSize={pageSize}&pageNumber={pageNumber}" )
```

Input Parameters

Parameter Name	Required	Description	Valid values
LocationType	Yes	Location type Store or warehouse	S, W
SearchString	Yes	search string for locations Id or Name	NA
tsfType	Yes	Transfer type	IC, MR
fromLocation	No	Selected from location ID	N/A
PageSize	No	Maximum number of locations to retrieve per page	N/A
PageNumber	No	Result page to retrieve	N/A

Output

TsfLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locType	String
locName	String
locCurrencyCode	String

Parameter Name	Data Type
entity	BigDecimal
entityDesc	String
tsfLocitemSearchRes	List<TsfLocitemSearchResRDO>

TsfLocitemSearchResRDO

Parameter Name	Data Type
item	String
availQty	BigDecimal
averageCost	BigDecimal
unitRetail	BigDecimal
currencyCode	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "links": [],
  "totalRecordCount": 1,
  "results": [
    {
      "links": [],
      "location": 5991,
      "locType": "S",
      "locName": "DIT Company Stockholding Store",
      "locCurrencyCode": "USD",
      "entity": 1000,
      "entityDesc": "Regular Stores",
      "tsfLocitemSearchRes": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	Yes	No	No	No
ALLOC_DETAIL	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
MV_LOC_SOB	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORG_UNIT	Yes	No	No	No
STORE	Yes	No	No	No
TRANSFER_LOC	Yes	No	No	No
TSF_ENTITY	Yes	No	No	No
V_STORE	Yes	No	No	No
V_TRANSFER_FROM_LOC	Yes	No	No	No
V_TRANSFER_TO_LOC	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Transfer - Refresh Transfer Items

This section describes the load items.

Business Overview

Load items service allows the user to refresh item records information for already selected items.

Service Type

Get

ReST URL

/Transfer/item/load?items={items}&fromLocation={fromLocation}

Input Parameters

Parameter Name	Required	Description
items	Yes	Comma Separated values for selected items' ID.
fromLocation	No	Selected from location ID.

Output

TsfltemSearchRDO

Parameter Name	Data Type
item	String

Parameter Name	Data Type
itemDesc	String
dept	BigDecimal
availQty	BigDecimal
averageCost	BigDecimal
unitRetail	BigDecimal
currencyCode	String
standardUnitOfMeasure	String
suppPackSize	BigDecimal
innerPackSize	BigDecimal
itemImageUrl	String

JSON Structure

```
[
  {
    "links": [],
    "item": "100001887",
    "itemDesc": "DIT Test 12:Black:Medium",
    "dept": 1102,
    "availQty": 100,
    "averageCost": 5,
    "unitRetail": 7.26,
    "currencyCode": "USD",
    "standardUnitOfMeasure": "CKG",
    "suppPackSize": 1,
    "innerPackSize": 1,
    "itemImageUrl": null,
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	Yes	No	No	No
ALLOC_DETAIL	Yes	No	No	No
DAILY_PURGE	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
ITEM_IMAGE	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ORDHEAD	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
STORE	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Transfer - Refresh Transfer Locations

This section describes the Load Locations service.

Business Overview

Load locations Web service allows user to refresh selected locations records.

Service Type

Get

ReST URL

/Transfer/loadLocations?fromLocation={fromLocation}&toLocation={toLocation}

Input Parameters

Parameter Name	Required	Description
FromLocation	No	Selected from location ID.
ToLocation	No	Selected to location ID.

Output

TsfLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locType	String
locName	String
locCurrencyCode	String
entity	BigDecimal
entityDesc	String
tsfLocitemSearchRes	List<TsfLocitemSearchResRDO>

TsfLocitemSearchResRDO

Parameter Name	Data Type
item	String
availQty	BigDecimal
averageCost	BigDecimal
unitRetail	BigDecimal

Parameter Name	Data Type
currencyCode	String

JSON Structure

```
[
  {
    "links": [],
    "location": 5991,
    "locType": "S",
    "locName": "DIT Company Stockholding Store",
    "locCurrencyCode": "USD",
    "entity": 1000,
    "entityDesc": "Regular Stores",
    "tsfLocitemSearchRes": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  },
  {
    "links": [],
    "location": 12310101,
    "locType": "W",
    "locName": "test",
    "locCurrencyCode": "USD",
    "entity": 1000,
    "entityDesc": "Regular Stores",
    "tsfLocitemSearchRes": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	Yes	No	No	No
ALLOC_DETAIL	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
MV_LOC_SOB	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORG_UNIT	Yes	No	No	No
STORE	Yes	No	No	No
TRANSFER_LOC	Yes	No	No	No
TSF_ENTITY	Yes	No	No	No
V_STORE	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
V_TRANSFER_FROM_LOC	Yes	No	No	No
V_TRANSFER_TO_LOC	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Get Locations on Existing Transfers

This section describes the transfer location search service.

Business Overview

The web service enables location search applicable for Transfers. Locations can be searched by either 'S'tore or 'W'arehouse, with the subsequent entry of a location number, a partial location number, a location description, or a partial location description in the search string.

The locations returned are constrained by the following criteria:

- When search type is warehouse then:
 - Internal finishers are filtered out
- When search type is store then:
 - Only company stores are returned
 - Only stockholding stores are returned

Service Type

Get

ReST URL

```
/Transfer/recent/transferLocSearch?
searchString={searchString}&locType={locType}&pageSize={pageSize}&pageNumber={
pageNumber}
```

Input Parameters

Parameter Name	Required	Description	Valid values
SearchString	No	Search string for locations ID or Name	N/A
LocType	No	Location type: Store or warehouse	S, W
PageSize	No	Maximum number of locations to retrieve per page	N/A
PageNumber	No	Result page to retrieve	N/A

Output

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal

Parameter Name	Data Type
results	List<RtsfLocSearchResultRDO>

RtsfLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locationType	String
locationName	String
currency	String

JSON Structure

```
{
  "totalRecordCount": 1,
  "results": [
    {
      "location": null,
      "locationType": null,
      "locationName": null,
      "currency": null,
      "links": [ ],
      "hyperMediaContent": {
        "linkRDO": [ ]
      }
    }
  ],
  "links": [ ],
  "hyperMediaContent": {
    "linkRDO": [ ]
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_STORE	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Get Transfer Statuses

This section describes the Transfer Status List service.

Business Overview

Retrieves all valid transfer statuses.

Service Type

Get

ReST URL

/Transfer/recent/transferStatusList

Input Parameters

No input.

Output

CodeDetailRDO

Parameter Name	Data Type
code	String
codeDescription	String
codeSequence	BigDecimal

JSON Structure

```
[
  {
    "code": null,
    "codeDescription": null,
    "codeSequence": null,
    "links": [ ],
    "hyperMediaContent": {
      "linkRDO": [ ]
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Get Transfer Summary

This section describes the Get Transfer Detail service.

Business Overview

Get Transfer Detail service allow user to retrieve Transfer information for a selected transfer number.

Service Type

Get

ReST URL

/Transfer/recent/transferDetail?
transferNumber={transferNumber}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description
TransferNumber	Yes	Transfer Number ID
PageSize	No	Maximum number of items to retrieve per page
PageNumber	No	Result page to retrieve

Output

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
results	List<RtsfTsfDtlRDO>

RtsfTsfDtlRDO

Parameter Name	Data Type
transferNumber	BigDecimal
status	String
fromLocation	BigDecimal
fromLocationName	String
finisher	BigDecimal
finisherName	String
toLocation	BigDecimal
toLocationName	String
transferType	String
totalCost	BigDecimal
totalRetail	BigDecimal
currency	String
deliveryDate	Long
createId	String
createDate	Long
transferItemsTable	List<RtsfTsfDtlItemRDO>

RtsfTsfDtlItemRDO

Parameter Name	Data Type
item	String
itemDescription	String
transferQuantity	BigDecimal

JSON Structure

```

{
  "totalRecordCount": null,
  "results": [
    {
      "transferNumber": null,
      "status": null,
      "fromLocation": null,
      "fromLocationName": null,
      "finisher": null,
      "finisherName": null,
      "toLocation": null,
      "toLocationName": null,
      "transferType": null,
      "totalCost": null,
      "totalRetail": null,
      "currency": null,
      "deliveryDate": null,
      "createId": null,
      "createDate": null,
      "transferItemsTable": [
        {
          "item": null,
          "itemDescription": null,
          "transferQuantity": null,
          "links": [ ],
          "hyperMediaContent": {
            "linkRDO": [ ]
          }
        }
      ],
      "links": [ ],
      "hyperMediaContent": {
        "linkRDO": [ ]
      }
    }
  ],
  "links": [ ],
  "hyperMediaContent": {
    "linkRDO": [ ]
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
TSF_ITEM_INV_FLOW	Yes	No	No	No
V_EXTERNAL_FINISHER	Yes	No	No	No
V_INTERNAL_FINISHER	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
V_ITEM_MASTER	Yes	No	No	No
V_LOCATION	Yes	No	No	No
V_STORE	Yes	No	No	No
V_TSFDETAIL	Yes	No	No	No
V_TSFHEAD	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Get Transfer Types

This section describes the Transfer Type List service.

Business Overview

Retrieves all valid transfer types.

Service Type

Get

ReST URL

/Transfer/recent/transferTypeList

Input Parameters

No input.

Output

CodeDetailRDO

Parameter Name	Data Type
code	String
codeDescription	String
codeSequence	BigDecimal

JSON Structure

```
[
  {
    "code": null,
    "codeDescription": null,
    "codeSequence": null,
    "links": [ ],
    "hyperMediaContent": {
      "linkRDO": [ ]
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Get Transfer Users

This section describes the Search Transfer User IDs.

Business Overview

The Search Transfer User IDs service retrieves for all User IDs that created transfers.

Service Type

Get

ReST URL

/Transfer/recent/searchUserIds?
searchString={searchString}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description
SearchString	Yes	search string for User Id
PageSize	No	Maximum number of transfer user IDs to retrieve per page
PageNumber	No	Result page to retrieve

Output

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
results	List<VarcharIdRDO>

VarcharIdRDO

Parameter Name	Data Type
id	String

JSON Structure

```
{
  "totalRecordCount": null,
  "results": [
    {
      "id": null,

```

```

        "links": [ ],
        "hyperMediaContent": {
            "linkRDO": [ ]
        }
    },
    "links": [ ],
    "hyperMediaContent": {
        "linkRDO": [ ]
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_TSFHEAD	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Get Transfers

This section describes the Transfer Search service.

Business Overview

The web services in this area enables search for applicable transfers. Transfers can be searched by their status, transfer types, transfer number, create date, delivery date, create ID, item department and/or locations. The transfers returned are constrained by the following criteria:

- Customer Orders and Book Transfers are filtered out.
- Only Transfers with transfer details are returned.

Service Type

Get

ReST URL

```

/Transfer/recent/transferSearch?
statuses={statuses}&transferTypes={transferTypes}&createIds={createIds}&startC
reateDate={startCreateDate}&endCreateDate={endCreateDate}&startDeliveryDate={s
tartDeliveryDate}&endDeliveryDate={endDeliveryDate}&transferNumber={transferNu
number}&locations={locations}&departments={departments}&pageSize={pageSize}&page
Number={pageNumber}

```

Input Parameters

Parameter Name	Required	Description	Valid values
Statuses	No	Comma Separated values for selected transfer statuses	
TransferTypes	No	Comma Separated values for selected transfer types	
CreateIds	No	Comma Separated values for selected transfer create ID	

Parameter Name	Required	Description	Valid values
StartCreateDate	No	Start of the range of transfer create dates	
EndCreateDate	No	End of the range of transfer create dates	
StartDeliveryDate	No	Start of the range of transfer create dates	
EndDeliveryDate	No	End of the range of transfer create dates	
TransferNumber	No	Transfer Number	
Locations	No	Comma Separated values for selected Location IDs	
Departments	No	Comma Separated values for selected Department IDs	
PageSize	No	Maximum number of locations to retrieve per page	
PageNumber	No	Result page to retrieve	

Output

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
results	List<RtsfSearchResRDO>

RtsfSearchResRDO

Parameter Name	Data Type
transferNumber	BigDecimal
tsfType	String
fromLocation	BigDecimal
fromLocationType	String
fromLocationName	String
toLocation	BigDecimal
toLocationType	String
toLocationName	String
status	String
totalCost	BigDecimal
currency	String
deliveryDate	Long

JSON Structure

```
{
  "totalRecordCount": null,
  "results": [
    {
      "transferNumber": null,
```

```

        "tsfType": null,
        "fromLocation": null,
        "fromLocationType": null,
        "fromLocationName": null,
        "toLocation": null,
        "toLocationType": null,
        "toLocationName": null,
        "status": null,
        "totalCost": null,
        "currency": null,
        "deliveryDate": null,
        "links": [ ],
        "hyperMediaContent": {
            "linkRDO": [ ]
        }
    },
    "links": [ ],
    "hyperMediaContent": {
        "linkRDO": [ ]
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_LOC_SOH	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
TSFDETAIL	Yes	No	No	No
TSFITEM_INV_FLOW	Yes	No	No	No
V_STORE	Yes	No	No	No
V_TSFDETAIL	Yes	No	No	No
V_TSFHEAD	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow - Refresh Transfer Locations

This section describes the Transfer Location Load service.

Business Overview

This web service allows the user to refresh already selected Transfer locations records.

Service Type

Get

ReST URL

/Transfer/recent/transferLocationLoad?locations={locations}

Input Parameters

Parameter Name	Required	Description
Locations	No	Comma-separated values for selected locations' ID

Output

RtsfLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locationType	String
locationName	String
currency	String

JSON Structure

```
[
  {
    "location": null,
    "locationType": null,
    "locationName": null,
    "currency": null,
    "links": [ ],
    "hyperMediaContent": {
      "linkRDO": [ ]
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_STORE	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow Transfer Upload Service

The following services are included in this functional area:

- [Create Transfer](#)
- [Mobile Workflow - Update Transfer Status](#)

Create Transfer

This section describes the Create Transfer service.

Business Overview

The Web service calls the existing Merchandising XTSF API directly with input parameters. For more information on Merchandising XTSF API, see Store Order Subscription API and Transfer Subscription API sections.

Service Type
Post

ReST URL

/Transfer

Input Parameters

Example JSON RDO Input

```
{
  "links" : [ ],
  "tsfdtlRDOs" : [
    {
      "links" : [ ],
      "item" : null,
      "tsfQty" : null,
      "suppPackSize" : null,
      "invStatus" : null,
      "unitCost" : null,
      "hyperMediaContent" : {
        "linkRDO" : [ ]
      }
    }
  ],
  "tsfNo" : null,
  "fromLocType" : null,
  "fromLoc" : null,
  "toLocType" : null,
  "toLoc" : null,
  "deliveryDate" : null,
  "dept" : null,
  "routingCode" : null,
  "freightCode" : null,
  "tsfType" : null,
  "status" : null,
  "userId" : null,
  "commentDesc" : null,
  "contextType" : null,
  "contextValue" : null,
  "hyperMediaContent" : {
    "linkRDO" : [ ]
  }
}
```

Output
N/A

Table Impact
For more information on the Merchandising XTSF API, see the Store Order Subscription API and Transfer Subscription API sections.

Mobile Workflow - Update Transfer Status

This section describes the Update Transfer Status service.

Business Overview

The web service approves or unapproves a transfer or a list of transfers.

Service Type

Post

ReST URL

```
/Transfer/recent/updateTransferStatus?
newStatus={newStatus}&transferNumbers={transferNumbers}
```

Input Parameters

Parameter Name	Required	Description	Valid values
NewStatus	Yes	New status of the transfer. May only be A- Approved or I-Input.	A, I
TransferNumbers	Yes	Comma Separated values for selected locations' ID	

Output

N/A

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
TSFHEAD	Yes	No	Yes	No
TSFDETAIL	Yes	Yes	Yes	Yes
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	Yes	No
ITEM_MASTER	Yes	No	No	No
PACKITEM_BREAKOUT	Yes	No	No	No
STORE	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
V_TSFHEAD	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Stock Order Status Upload Service

The following services are included in this functional area:

- [Manage Stock Order Status](#)

Manage Stock Order Status

Functional Area

Inventory - Transfers and Allocations

Business Overview

A stock order is an outbound merchandise request from a warehouse or store. In Merchandising, a stock order takes the form of either a transfer or allocation. Merchandising subscribes to stock order status messages published by an external application, such as a store system (SIOCS, for example) or a warehouse management system (Oracle WMS Cloud, for example) to communicate the status of a specific stock order.

Listed are the stock order statuses for both transfers and allocation document types and what occurs in Merchandising after receiving the respective status. Statuses other than listed below are ignored by Merchandising.

Statuses for Document Types T, D, and S

Document types of T, D, and S all refer to transfers and indicate if the transfer is initiated in Merchandising, a warehouse system, or a store system, respectively.

- SI (Stock Increased) - Insert or increase transfer quantity and increase item/location transfer reserve quantity for the source location and transfer expected quantity for the destination location.
- SD (Stock Decreased) - Delete or decrease the transfer quantity for the transfer/item combination. Transfer quantity for the transfer/item combination will be deleted if the transfer has been created but has not been shipped. Additionally, the item/location transfer reserved quantity for the source location and the transfer expected quantity for the destination location will be decreased.
- DS (Details Selected) - Increase the selected quantity for the transfer/item combination.
- DU (Details Un-selected) - Decrease selected and transfer quantity for the transfer/item by the quantity on the message and increase the cancelled quantity. Additionally, it will decrease the reserved quantity for the source location and decrease the expected quantity for the destination location by the lesser of the quantity on the message and transfer - shipped quantity. The transfer will also be added to the document close queue if transfer status is not closed. Document Close batch program will then determine if the transfer should be closed based on certain conditions. Transfers with outstanding appointments are not closed.
- PP (Distributed) - Decreases the selected quantity and increases the distro quantity for the transfer/item.
- PU (Un-Distribute) - Decreases the distro quantity for the transfer/item.
- RS (Return to Stock) - Decreases distro quantity and transfer quantity for the transfer/item; the cancelled quantity for the transfer/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the transfer - shipped quantity if the transfer status is not closed.
- EX (Expired) - Decreases transfer quantity for the transfer/item; the cancelled quantity for the transfer/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the transfer - shipped quantity if the transfer status is not closed. The transfer will also be added to the document close queue if transfer status is not closed. Document Close batch program will then determine if the transfer

should be closed based on certain conditions. Transfers with outstanding appointments are not closed.

- SR (Store Reassign) - Updates the distro quantity for the transfer/item. This can either increase or decrease the value, depending on whether a positive or negative value is sent.

Statuses for Document Type A

Document type A is always used for Allocations.

- SI (Stock Increased) - Insert or increase allocated quantity and increase item/location transfer reserve quantity for the source location and transfer expected quantity for the destination location.
- SD (Stock Decreased) - Decrease the allocated quantity for the allocation/item combination. Additionally, the item/location transfer reserved quantity for the source location and the transfer expected quantity for the destination location will be decreased.
- DS (Details Selected) - Increase the selected quantity for the allocation/item combination.
- DU (Details Un-Selected) - Decrease the selected quantity for the allocation/item combination.
- NI (WMS Line Cancellation) - Decrease selected and allocation quantity for the allocation/item by the quantity on the message and increase the cancelled quantity. Additionally, it will decrease the reserved quantity for the source location and decrease the expected quantity for the destination location by the lesser of the quantity on the message and allocation - shipped quantity if the allocation is not closed. The allocation will also be added to the document close queue if allocation status is not closed. Document Close batch program will then determine if the allocation should be closed based on certain conditions. Allocations with outstanding appointments are not closed.
- PP (Distributed) - Decreases the selected quantity and increases the distro quantity for the allocation/item.
- PU (Un-Distribute) - Decreases the distro quantity for the allocation/item.
- RS (Return to Stock) - Decreases distro quantity and allocation quantity for the allocation/item; the cancelled quantity for the allocation/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the allocation - shipped quantity if the allocation status is not closed.
- EX (Expired) - Decreases allocation quantity for the allocation/item; the cancelled quantity for the allocation/item is increased. Additionally, transfer reserved is decreased for the item/source location and transfer expected is decreased for the item/destination location for the lesser of the quantity in the message and the allocation - shipped quantity if the allocation status is not closed. The allocation will also be added to the document close queue if allocation status is not closed. Document Close batch program will then determine if the allocation should be closed based on certain conditions. Allocations with outstanding appointments are not closed.
- SR (Store Reassign) - Updates the distro quantity for the allocation/item. This can either increase or decrease the value, depending on whether a positive or negative value is sent.

For customer orders, Merchandising assumes it will get updates from an OMS for customer order related stock orders. Therefore, to avoid duplicate processing, Merchandising will ignore No Inventory, Expired, Stock Decreased, and Stock Increased statuses received for a customer order transfer.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/stockOrders/status/manage

Input Payload Details

Table 5-352 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of stock order status records.

Table 5-353 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
location	Yes	Number (10)	Contains the location number of the stock order source location.
distroNo	No	Number (12)	This field contains the stock order number. This is either the transfer or allocation number in Merchandising.
distroType	Yes	String (1)	This field specifies whether the stock order status pertains to an allocation (A) or transfer (T - transfers created in Merchandising, D - transfers created in a warehouse management system or S - transfers created in a store inventory system) that is already existing in Merchandising. Customer Order (C), and Virtual Distro (V) are also valid document types but will be ignored by Merchandising.

Table 5-353 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contextType	No	String (6)	This field holds the reason code related to which a transfer is made. Valid values are found in the code type CNTX. Examples are Promotion (PROM), Customer Transfer (WED), Store Requisition (STORE) and Repairing (REPAIR).
contextValue	No	String (25)	This field holds the value relating to the context type like promotion number.
inventoryType	No	String (6)	This field indicates if a transfer is made from the available (A) or unavailable (U) inventory.
customerOrderNo	No	String (48)	This field holds the master customer order number for a stock order associated with a customer order.
fulfillOrderNo	No	String (48)	This field holds the number related to the fulfillment details for a stock order associated with a customer order. One or more fulfillment orders could relate back to a single customer order.
details	Yes	Collection of Object	References a collection of stock order details.

Table 5-354 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	Yes	Number (10)	Contains the location number of the stock order receiving location.
item	Yes	String (25)	This contains the unique identifier for the item.

Table 5-354 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitQuantity	No	Number (12,4)	This field contains the difference between the number of item units shipped versus the receiving count for the given item. This is subtracted from the document-line-item-unit-count to yield an over/under variance between what a supplier said was shipped and what was counted and received at by the store's staff.
status	Yes	String (2)	This field contains the status of the stock order. Valid values are: Accepted (SI), Rejected (SD), Distributed (PP), Un-Distributed (PU), Details Selected (DS), Details Un-selected (DU), WMS Line Cancellation (NI), Return To Stock (RS), Expired (EX), and Store Reassign (SR). Statuses other than listed are ignored by Merchandising.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "location": 2,
      "distroNo": 100000190000,
      "distroType": "T",
      "contextType": "PROM",
      "contextValue": "2255",
      "inventoryType": "A",
      "customerOrderNo": null,
      "fulfillOrderNo": null,
      "details": [
        {
          "toLocation": 1211,
          "item": "102900034",
          "unitQuantity": 10,
          "status": "SI"
        }
      ]
    }
  ]
}
```



```
}  
]  
}  
]  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Transfer Detail Service

This section describes the Transfer Detail service.

Business Overview

Transfer Detail service allows user to retrieve details for a given transfer.

Service Type

Get

ReST URL

Transfer/transferDetail?transferNumber={transferNumber}

Input Parameters

Parameter Name	Required	Description
Transfer Number	Yes	Transfer number

Output

RestTsfheadRecRDO

Parameter Name	Data Type
tsfNo	BigDecimal
tsfParentNo	BigDecimal
fromLocType	String
fromLoc	BigDecimal
toLocType	String
toLoc	BigDecimal
expDcDate	Timestamp
dept	BigDecimal
inventoryType	String
tsfType	String
status	String
deliveryDate	Timestamp
closeDate	Timestamp
notAfterDate	Timestamp
contextType	String
contextValue	String
wfOrderNo	BigDecimal
tsfdetail	List<RestTsfdetailRecRDO>

RestTsfdetailRecRDO

Parameter Name	Data Type
tsfSeqNo	BigDecimal
item	String
invStatus	BigDecimal
tsfPrice	BigDecimal
tsfQty	BigDecimal
fillQty	BigDecimal
shipQty	BigDecimal
receivedQty	BigDecimal
reconciledQty	BigDecimal
distroQty	BigDecimal
selectedQty	BigDecimal

Parameter Name	Data Type
cancelledQty	BigDecimal
suppPackSize	BigDecimal
tsfCost	BigDecimal
publishInd	String

JSON Structure

```

{
  "tsfNo": null,
  "tsfParentNo": null,
  "fromLocType": null,
  "fromLoc": null,
  "toLocType": null,
  "toLoc": null,
  "expDcDate": null,
  "dept": null,
  "inventoryType": null,
  "tsfType": null,
  "status": null,
  "deliveryDate": null,
  "closeDate": null,
  "notAfterDate": null,
  "contextType": null,
  "contextValue": null,
  "wfOrderNo": null,
  "tsfdetail": [
    {
      "tsfSeqNo": null,
      "item": null,
      "invStatus": null,
      "tsfPrice": null,
      "tsfQty": null,
      "fillQty": null,
      "shipQty": null,
      "receivedQty": null,
      "reconciledQty": null,
      "distroQty": null,
      "selectedQty": null,
      "cancelledQty": null,
      "suppPackSize": null,
      "tsfCost": null,
      "publishInd": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

```

    }
  }

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
TSFHEAD	Yes	No	No	No
TSFDETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Transfer Download Service

The following services are included in this functional area:

- [Get Transfer](#)
- [Get Transfer Details for Provided Transfer](#)

Get Transfer

Endpoint

MerchIntegrations/services/inventory/transfer

MerchIntegrations/services/inventory/transfer/{tsfNo}

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service publishes transfer transactions to downstream consuming systems. It covers transfer header and line-detail information for stock movement between stores, warehouses, internal finishers, and external finishers, and, for customer-order transfers, it also publishes delivery, billing, and consumer-fulfillment enrichment. Only transfers in Approved (A), Shipped (S), Closed (C), Picked (P), or Selected (L) status are published. Transfers still in Input (I), Submitted (B), externally closed (X), or Deleted (D) status do not appear in this API payload.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: TSFHEAD, TSFDETAIL, ORDCUST

Additional business tables: ITEM_LOC, ITEM_TICKET, PACKITEM_BREAKOUT, ITEM_MASTER, ORDCUST_DETAIL, ORDCUST_CUSTOMER_DETAIL, STORE, WH, PARTNER, DELIVERY_SLOT, V_CODE_DETAIL_TL, SYSTEM_OPTIONS, TSFHEAD_CFA_EXT

JSON cache table: MERCHAPI_EXT_TRANSFER

JSON generation view: V_MERCHAPI_EXT_TSF_JSON

Pulls transfer header data from TSFHEAD, line details from TSFDETAIL, item-location selling attributes from ITEM_LOC, ticket and component detail from ITEM_TICKET, PACKITEM_BREAKOUT, and ITEM_MASTER, customer-order enrichment from ORDCUST, ORDCUST_DETAIL, and ORDCUST_CUSTOMER_DETAIL, delivery-slot text from DELIVERY_SLOT, location enrichment from STORE, WH, and PARTNER, and custom flex attributes from MERCHAPI_EXT_TRANSFER_CFA.

The view only publishes transfers whose TSFHEAD.STATUS is one of A, S, C, P, or L, so input, submitted, externally closed, and already-deleted transfer headers are not published through this API.

The view also derives additional business fields needed by downstream consumers.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_TSF_JSON and apply TSFHEAD.STATUS IN ('A','S','C','P','L').

Mark cache rows deleted when the transfer no longer exists in TSFHEAD.

ICL (Integration Change Log) table: ICL_RMS_TRANSFER (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on TSFHEAD, TSFDETAIL, TSFHEAD_CFA_EXT, and qualifying ORDCUST updates.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_TRANSFER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_TRANSFER_ADHOC_PROCESS ->
MERCHAPI_EXT_TRANSFER_ADHOC_JOB

Webhook configuration api name: inventory/transfer

Special handling

Additional GET query parameters: fromLoc, toLoc, fromLocType, toLocType.

These filters are not backed by dedicated cache-table indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. Because this service builds transfer messages from multiple base and supporting tables, including customer-order enrichment and line-detail expansion, the available configuration should be validated against the customer production transfer profile, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through `API_NO_OF_THREADS`, commit sizing through `API_COMMIT_MAX_COUNTER`, and long-running initial seeding can also be segmented through `API_EXEC_SECONDS`.

Thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the message from multiple base and supporting tables. Common starting points for large-volume runs are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment.

`API_COMMIT_MAX_COUNTER` should be sized together with the thread setting. Typical values range from about 1,000 rows and can extend to 10,000.

`API_EXEC_SECONDS` is mainly relevant for initial seeding. A value of about 30 minutes helps the batch pause and resume cleanly so tuning changes can be applied between runs if needed.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes. `GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a specific reason to change it.

REST fetch considerations

Transfer payloads can be comparatively large, so REST page size should be selected based on average payload size so the average response remains below about 10 MB per call. Common page-limit values are 500, 1,000, 5,000, or 10,000 rows, with the lower end often proving better for broader transfer messages.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned `before` timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/transfer

MerchIntegrations/services/inventory/transfer/{tsfNo}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
fromLoc	No	Number	From Location
toLoc	No	Number	To Location
fromLocType	No	String	From Location Type
toLocType	No	String	To Location Type
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-355 Input Parameter for "MerchIntegrations/services/inventory/transfer/{tsfNo}"

Parameter Name	Required	Data Type	Description
tsfNo	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Table 5-355 (Cont.) Input Parameter for "MerchIntegrations/services/inventory/transfer/{tsfNo}"

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-356 PageResultsItemsMerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-357 MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for transfer details.
transferNo	No	Number(12,0)	This field specifies a number that uniquely identifies the transfer within the system.
docType	No	String (1)	This field is defaulted to T for transfers.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
physicalFromLocation	No	Number(10,0)	This field specifies the number that is used to identify the physical warehouse for a virtual warehouse when the from location is a warehouse. Store ID is populated when the from location is a store.
fromLocationType	No	String (1)	This field specifies the location type of the transfer from location.
fromStoreType	No	String (6)	This field specifies whether a particular store is a franchise or company store if the from location is a store. This field is not populated if the from location is a warehouse.
fromStockholdingInd	No	String (1)	This field specifies whether the store can hold stock. In a non-multichannel environment this will always be Y if the from location is a store. This field is not populated if the from location is a warehouse.
fromLocation	No	Number(10,0)	This field specifies the location number of the transfer from location.
physicalToLocation	No	Number(10,0)	This field specifies the number that is used to identify the physical warehouse for a virtual warehouse when the to location is a warehouse. Store ID is populated when the to location is a store.
toLocationType	No	String (1)	This field specifies the location type of the transfer to location.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toStoreType	No	String (6)	This field specifies whether a particular store is a franchise or company store if the to location is a store. This field is not populated if the to location is a warehouse.
toStockholdingInd	No	String (1)	This field specifies whether the store can hold stock. In a non-multichannel environment this will always be Y if the to location is a store. This field is not populated if the to location is a warehouse.
toLocation	No	Number(10,0)	This field specifies the location number of the transfer to location.
transferType	No	String (6)	This field identifies the type or reason for the transfer.
pickNotBeforeDate	No	date	This field specifies the date the transfer was approved.
pickNotAfterDate	No	date	This field is calculated as the transfer approval date plus the days from code_detail where code_type = 'DEFT' and code = 'DATE'.
orderType	No	String (9)	This field specifies the order type associated with transfers. RWMS uses this value to determine how to distribute stock orders.
breakByDistro	No	String (1)	This field is set to 'Y' if the delivery_type for the transfer's customer is 'S'. Otherwise, it is set to 'N'.
deliveryDate	No	date	This field specifies the earliest date that the transfer can be delivered to the store. This will be populated if the transfer type is PO-Linked.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deliverAdd1	No	String (240)	For customer order transfers, this field contains the first line of the customer's delivery address. For other types of transfer this field is not populated.
deliverAdd2	No	String (240)	For customer order transfers, this field contains the second line of the customer's delivery address. For other types of transfer this field is not populated.
deliverCity	No	String (120)	For customer order transfers, this field contains the city portion of the customer's delivery address. For other types of transfer this field is not populated.
deliverState	No	String (3)	For customer order transfers, this field contains the state portion of the customer's delivery address. For other types of transfer this field is not populated.
deliverPost	No	String (30)	For customer order transfers, this field contains the postal code portion of the customer's delivery address. For other types of transfer this field is not populated.
deliverCountryId	No	String (3)	For customer order transfers, this field contains the country portion of the customer's delivery address. For other types of transfer this field is not populated.
comments	No	String (2000)	This field specifies any comments associated with the transfer.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferParentNo	No	Number(12,0)	This field identifies the transfer at the level above the transfer.
expectedDcDate	No	date	This field specifies the date the inventory is expected to arrive at the DC.
expectedDcEowDate	No	date	This column holds the eow_date for the exp_dc_date column. It is used for OTB extracts for intercompany transfers.
approvalId	No	String (30)	This field specifies the ID of the user who approved the transfer.
approvalDate	No	date	This field specifies the date the transfer was approved.
fromLocationTransfer Entity	No	Number(10,0)	This field specifies the transfer entity of the from location.
toLocationTransfer Entity	No	Number(10,0)	This field specifies the transfer entity of the to location.
invType	No	String (6)	This field specifies that a transfer can only contain available or unavailable merchandise, never a combination of both. This field indicates whether the transfer is for available inventory or unavailable inventory.
transferStatus	No	String (1)	This field specifies the status of the transfer.
notAfterDate	No	date	This field specifies the last day delivery of the transfer will be accepted.
contextType	No	String (6)	This field can be used to indicate the transfer reason or category, for example 'promotion'.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contextValue	No	String (25)	This field can be used to further indicate the reason or category for the transfer. For instance, if context_type was 'promotion', this could be the promotion number.
deliverySlotId	No	String (15)	This field specifies when the transfer quantity is needed at the location.
deliverySlotDescription	No	String (240)	This field specifies the delivery slot description.
customerOrderNo	No	String (48)	For customer order transfers, this field contains the master customer order number from the Order Management System.
fulfillmentOrderNumber	No	String (48)	This field contains the fulfillment order number for the transfer as provided by OMS. One or more fulfillment orders could relate back to a single customer order in OMS.
carrierCode	No	String (4)	This field indicates the carrier the order is to be shipped with, if specified on the order.
carrierServiceCode	No	String (6)	This field indicates the method that was selected for shipping by the customer placing the order (for example, Standard Shipping or Overnight).
consumerDeliveryDate	No	date	This field specifies the desired date the delivery is required by the customer.
consumerDeliveryTime	No	dateTime	This field specifies the desired time the delivery is required by the customer.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deliverFirstName	No	String (120)	For customer order transfers, this field contains the first name for the delivery address on the order. For other types of transfer this field is not populated.
deliverPhoneticFirst	No	String (120)	For customer order transfers, this field contains the phonetic first name for the delivery address on the order. For other types of transfer this field is not populated.
deliverLastName	No	String (120)	For customer order transfers, this field contains the last name for the delivery address on the order. For other types of transfer this field is not populated.
deliverPhoneticLast	No	String (120)	For customer order transfers, this field contains the phonetic last name for the delivery address on the order. For other types of transfer this field is not populated.
deliverPreferredName	No	String (120)	For customer order transfers, this field contains the preferred name for the delivery address on the order. For other types of transfer this field is not populated.
deliverCompanyName	No	String (120)	For customer order transfers, this field contains the company name for the delivery address on the order. For other types of transfer this field is not populated.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deliverAdd3	No	String (240)	For customer order transfers, this field contains the third line of the customer's delivery address. For other types of transfer this field is not populated.
deliverCounty	No	String (250)	For customer order transfers, this field contains the county portion of the customer's delivery address. For other types of transfer this field is not populated.
deliverPhone	No	String (20)	For customer order transfers, this field contains the delivery phone number. For other types of transfer this field is not populated.
billFirstName	No	String (120)	For customer order transfers, this field contains the first name for the billing address on the order. For other types of transfer this field is not populated.
billPhoneticFirst	No	String (120)	For customer order transfers, this field contains the phonetic first name for the billing address on the order. For other types of transfer this field is not populated.
billLastName	No	String (120)	For customer order transfers, this field contains the last name for the billing address on the order. For other types of transfer this field is not populated.
billPhoneticLast	No	String (120)	For customer order transfers, this field contains the phonetic last name for the billing address on the order. For other types of transfer this field is not populated.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
billPreferredName	No	String (120)	For customer order transfers, this field contains the preferred name for the billing address on the order. For other types of transfer this field is not populated.
billCompanyName	No	String (120)	For customer order transfers, this field contains the company name for the billing address on the order. For other types of transfer this field is not populated.
billAdd1	No	String (240)	For customer order transfers, this field contains the first line of the customer's billing address. For other types of transfer this field is not populated.
billAdd2	No	String (240)	For customer order transfers, this field contains the second line of the customer's billing address. For other types of transfer this field is not populated.
billAdd3	No	String (240)	For customer order transfers, this field contains the third line of the customer's billing address. For other types of transfer this field is not populated.
billCounty	No	String (250)	For customer order transfers, this field contains the county portion of the customer's billing address. For other types of transfer this field is not populated.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
billCity	No	String (120)	For customer order transfers, this field contains the city portion of the customer's billing address. For other types of transfer this field is not populated.
billCountry	No	String (3)	For customer order transfers, this field contains the country portion of the customer's billing address. For other types of transfer this field is not populated.
billPost	No	String (30)	For customer order transfers, this field contains the postal code portion of the customer's billing address. For other types of transfer this field is not populated.
billState	No	String (3)	For customer order transfers, this field contains the state portion of the customer's billing address. For other types of transfer this field is not populated.
billPhone	No	String (20)	For customer order transfers, this field contains the billing phone number. For other types of transfer this field is not populated.
partialDeliveryInd	No	String (1)	For customer order transfers, this field indicates if the order can be picked and shipped partially or if it should be shipped only when complete.
consumerDirectInd	No	String (1)	This field indicates the order as being consumer direct (Y) or not (N). An order is consumer direct if it is a customer order.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	No	Number(12,0)	This field contains the import order number for which the intercompany book transfer was created for POs created using an importer.
freightCode	No	String (1)	This field is used to determine the priority for this transfer. Valid values for this field are: N - Normal, E - Expedite, H - Hold.
routingCode	No	String (1)	This field is used to indicate the type of freight to use on the transfer. Valid values are 1 - 1 day expedite, 2 - 2 day expedite, 3 - 3 day expedite.
externalReferenceNo	No	String (14)	This field contains the audit trail reference to an external system when an external transaction initiates master record creation in the Oracle Retail system.
massReturnTransferNo	No	Number(10,0)	This field holds the mass return transfer number with which this transfer is associated. This is the primary key for the table MRT.
restockingPercentage	No	Number(12,4)	This field holds the percentage of cost charged by the receiving location for re-stocking.
franchiseOrder	No	Number(10,0)	This field holds the franchise order number this transfer is linked to.
franchiseReturnAuthorizationNo	No	Number(10,0)	This field holds the franchise return number this transfer is linked to.

Table 5-357 (Cont.) MerchApiTransfer - Object. See list of elements for detail

Element Name	Required	Data Type	Description
franchiseNeedDate	No	date	This field contains the need date of the franchise order. This is populated only for franchise order transfers.
closeDate	No	date	This field contains the date the transfer was closed.
createId	No	String (254)	This field contains the user ID of the user that created the transfer.
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).
details	No	Collection of Object	This list holds attributes for transfer record details.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the transfer.
cacheTimestamp	No	dateTime	This field specifies the date and time when the transfer record was first picked for processing from the ICL table.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-358 TransferDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferSeqNo	No	Number(8,0)	This field contains a sequence number that, when combined with the transfer number, makes this detail row unique.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.

Table 5-358 (Cont.) TransferDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferQuantity	No	Number(12,4)	This field specifies the total quantity of the item reserved at the from location for this transfer.
price	No	Number(20,4)	This field specifies the unit retail price in the selling unit of measure for the item/location combination.
sellingUom	No	String (4)	This field specifies the selling unit of measure of the item.
expediteFlag	No	String (1)	This field is determined from the transfer's freight code field.
storeOrderMultiple	No	String (1)	This field specifies the multiple in which the item needs to be shipped from a warehouse to the location.
transferOrderLinkNo	No	Number(12,0)	This field specifies the reference number to link the item on the transfer to any purchase orders that have been created to allow the from location (that is, warehouse) on the transfer to fulfill the transfer quantity to the to location (that is, store) on the transfer.
ticketTypeId	No	String (4)	This field specifies the ticket type associated with the item.
ticketDetails	No	Collection of Object	This list holds attributes for transfer detail ticket records.
invStatus	No	Number(2,0)	This field specifies inventory status, if one exists, for the transfer detail record. The inventory status field corresponds to the inventory status type found on the INV_STATUS_TYPES table.

Table 5-358 (Cont.) TransferDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transactionUom	No	String (4)	This field specifies the original transaction unit of measure to be passed on to the store system. It is only used for customer direct orders coming from the Order Management System. All quantities in the response payload are expressed in the item's standard unit of measure.
itemLineNo	No	Number(10,0)	This field specifies the detail item line number for customer order transfers. For other types of transfer this field is not populated.
comments	No	String (2000)	This field specifies any comments associated with the transfer line item.
transferPrice	No	Number(20,4)	This field holds the unit cost of the item that will be used for this specific transfer if being transferred at something other than current WAC. This field should only have a value for intercompany transfers.
transferCost	No	Number(20,4)	This field holds the unit cost of the item that will be used for this specific transfer if the current shipping location's WAC is not used.
supplierPackSize	No	Number(12,4)	This field contains the supplier pack size for this transfer/item.

Table 5-358 (Cont.) TransferDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
defaultCharges2ndLegInd	No	String (1)	This field is an indicator to identify whether item up charges, if they exist, should be defaulted to the 2nd leg of the transfer. The valid values are Y, N, and NULL. Its value is set on the transfer detail form when applying item up charges to the 1st leg.
restockingPercentage	No	Number(12,4)	This field contains the percentage of cost charged by the supplier for restocking. This could be different than the restocking percentage held in TSFHEAD if the user decides to change the percentage at item level.
finisherAverageRetail	No	Number(20,4)	This field is applicable to two-legged transfers and contains the weighted average retail of all the units of the transfer item currently at or in transit to the finisher.
finisherUnits	No	Number(12,4)	This field is applicable to two-legged transfers and contains the number of units of the transfer item currently at or in transit to the finisher.
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).

Table 5-359 TransferDetailsTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentItem	No	String (25)	This field specifies the alphanumeric value that identifies the component item within the pack.
componentPrice	No	Number(20,4)	This field specifies the unit retail price in the selling unit retail of the component item within the pack.
componentSellingUom	No	String (4)	This field specifies the selling unit of measure for a component item within the pack.

Table 5-360 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-361 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-361 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "transferNo": 12300056,
      "docType": null,
      "physicalFromLocation": 1,
      "fromLocationType": "S",
      "fromStoreType": "C",
      "fromStockholdingInd": "Y",
      "fromLocation": 1521,
      "physicalToLocation": 1,
      "toLocationType": "S",
      "toStoreType": "C",
      "toStockholdingInd": "Y",
      "toLocation": 1511,
      "transferType": "AD",
      "pickNotBeforeDate": "2001-12-31",
      "pickNotAfterDate": "2001-12-31",
      "orderType": null,
      "breakByDistro": null,
      "deliveryDate": "2001-12-31",
      "deliverAdd1": null,
      "deliverAdd2": null,
      "deliverCity": null,
      "deliverState": null,
      "deliverPost": null,
      "deliverCountryId": null,
      "comments": "transfer to loc 1511",
      "transferParentNo": 1,
      "expectedDcDate": "2001-12-31",
      "expectedDcEowDate": "2001-12-31",
      "approvalId": null,
      "approvalDate": "2001-12-31",
      "fromLocationTransferEntity": 1,
      "toLocationTransferEntity": 1,
      "invType": null,
      "transferStatus": null,
      "notAfterDate": "2001-12-31",
    }
  ]
}
```



```

"contextType": null,
"contextValue": null,
"deliverySlotId": null,
"deliverySlotDescription": null,
"customerOrderNo": "12342132156",
"fulfillmentOrderNumber": "123123213",
"carrierCode": "30",
"carrierServiceCode": "D",
"consumerDeliveryDate": "2001-12-31",
"consumerDeliveryTime": "2001-12-31T23:59:59.000Z",
"deliverFirstName": null,
"deliverPhoneticFirst": null,
"deliverLastName": null,
"deliverPhoneticLast": null,
"deliverPreferredName": null,
"deliverCompanyName": null,
"deliverAdd3": null,
"deliverCounty": null,
"deliverPhone": null,
"billFirstName": null,
"billPhoneticFirst": null,
"billLastName": null,
"billPhoneticLast": null,
"billPreferredName": null,
"billCompanyName": null,
"billAdd1": null,
"billAdd2": null,
"billAdd3": null,
"billCounty": null,
"billCity": null,
"billCountry": null,
"billPost": null,
"billState": null,
"billPhone": null,
"partialDeliveryInd": "N",
"consumerDirectInd": null,
"orderNo": 464690012,
"freightCode": "N",
"routingCode": null,
"externalReferenceNo": null,
"massReturnTransferNo": 1,
"restockingPercentage": 1.0,
"franchiseOrder": 1,
"franchiseReturnAuthorizationNo": 1,
"franchiseNeedDate": "2001-12-31",
"closeDate": "2001-12-31",
"createId": "RMS01APP",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"details": [
{
"transferSeqNo": 1,
"item": "104100022",
"transferQuantity": 3,
"price": 1.0,
"sellingUom": "EA",

```

```

"expediteFlag": "Y",
"storeOrderMultiple": null,
"transferOrderLinkNo": 1,
"ticketTypeId": null,
"ticketDetails": [
{
"componentItem": null,
"componentPrice": 1.0,
"componentSellingUom": "EA"
}
],
"invStatus": 1,
"transactionUom": "EA",
"itemLineNo": 1,
"comments": null,
"transferPrice": 1.0,
"transferCost": 1.0,
"supplierPackSize": 3,
"defaultCharges2ndLegInd": null,
"restockingPercentage": 1.0,
"finisherAverageRetail": 1.0,
"finisherUnits": 1.0,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	Yes	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_TRANSFER	Yes	No	Yes	Yes
MERCHAPI_EXT_TRANSFER	Yes	Yes	Yes	Yes
MERCHAPI_EXT_TRANSFER_CFA	No	Yes	No	Yes
TSFHEAD	Yes	No	No	No
TSFHEAD_CFA_EXT	Yes	No	No	No
V_MERCHAPI_EXT_TSF_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

Get Transfer Details for Provided Transfer

Endpoint

```
MerchIntegrations/services/inventory/transfer
MerchIntegrations/services/inventory/transfer/{tsfNo}
```

Functional Area

Inventory - Transfers

Business Overview

This service can be used by external applications to get all or selected transfers and its details from Merchandising.

Service Type

GET

ReST URL

```
/MerchIntegrations/services/inventory/transfer?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
```

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
fromLoc	No	Number	Source Location
toLoc	No	Number	Destination location
fromLocType	No	String	Source location type
toLocType	No	String	Destination location type

Parameter Name	Required	Data Type	Description
offsetkey	No	String	Offset Key. Valid Value is transfer ID.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.

Output

Name	Data Type	Description
action	String	Action for Item details. Valid Values are INSERT, UPDATE and DELETE.
transferNo	Number	This field specifies a number that uniquely identifies the transfer within the system.
docType	String	This field Specifies the type of document. This field is defaulted to T for Transfers.
physicalFromLocation	String	This field specifies number that is used to identify physical warehouse for virtual warehouse in case from location is warehouse. Store+ id is populated in case of store.
fromLocationType	String	This field specifies the location type of from location of the transfer
fromStoreType	String	This field specifies whether a particular store is a franchise or company store if from location is Store. Field not populated if from location is Warehouse.
fromStockholdingInd	Number	This field specifies whether the store can hold stock. In a non-multichannel environment this will always be Y, if from location is Store. Field not populated if from location is Warehouse.
fromLocation	Number	This field specifies location number of the transfer from location.
physicalToLocation	String	This field specifies number that is used to identify physical warehouse for virtual warehouse in case to location is warehouse. Store id is populated in case of store.
toLocationType	String	This field specifies the location type of to location of the transfer.
toStoreType	String	This field specifies whether a particular store is a franchise or company store if to location is Store. Field not populated if to location is Warehouse.
toStockholdingInd	Number	This field specifies whether the store can hold stock. In a non-multichannel environment this will always be Y, if to location is Store. Field not populated if to location is Warehouse.
toLocation	Number	This field specifies location number of the transfer to location.
transferType	Number	This field identifies the type or reason for the transfer.
pickNotBeforeDate	String	This field Contains the date on which the transfer should be released from the from location for delivery to the to location.

Name	Data Type	Description
pickNotAfterDate	Number	This field is calculated as transfer approval date + days from code_detail where code_type = 'DEFT' and code = 'DATE'.
orderType	String	This field specifies order type associated with transfers.
breakByDistro	String	This field is set to 'Y' if the deliver_type for the transfer's customer is 'S'. Otherwise, set to N
deliveryDate	String	This field specifies earliest date that the transfer can be delivered to the store. This will be populated if the transfer type is PO-Linked.
deliverAdd1	String	For customer order transfers, this field contains the first line of the delivery address of the customer. For other type's of transfer this field would not be populated.
deliverAdd2	String	For customer order transfers, this field contains the second line of the delivery address of the customer. For other types of transfer this field would not be populated.
deliverCity	String	For customer order transfers, this field contains the city portion of the customer. For other types of transfer this field would not be populated.
deliverState	String	For customer order transfers, this field contains the state portion of the customer. For other types of transfer this field would not be populated.
deliverPost	String	For customer order transfers, this field contains the postal code portion of the customer. For other types of transfer this field would not be populated.
deliverCountryId	String	For customer order transfers, this field contains the country portion of the customer. For other types of transfer this field would not be populated.
comments	String	This field specifies any comments associated with the transfer.
details		This list holds detail records for the transfers
transferParentNo	Number	This field identifies the transfer at the level above the transfer.
expectedDcDate	String	This field specifies date the inventory is expected to arrive at the DC.
approvalId	String	This field specifies ID of the user who approved the transfer.
approvalDate	String	This field specifies date the transfer was approved.
fromLocationTransferEntity	Number	This field specifies transfer entity of the from location.
toLocationTransferEntity	Number	This field specifies transfer entity of the to location.

Name	Data Type	Description
invType	String	This field specifies transfer can only contain available or unavailable merchandise, never a combination of both. This field will indicate whether the transfer is for Available inventory or Unavailable inventory.
transferStatus	String	This field specifies Status of the transfer.
notAfterDate	String	This field specifies last day delivery of the transfer will be accepted.
contextType	String	This field can be used to indicate the transfer reason or category, e.g., 'promotion'.
contextValue	String	This field can be used to further info to indicate the reason or category for the transfer. For instance, if context type was 'promotion', this could be the promotion number.
deliverySlotId	String	This field Specifies when the transfer quantity is needed at the location.
deliverySlotDescription	String	This field Specifies delivery slot description.
customerOrderNo	String	or customer order transfers, this field contains the master customer order number from the Order Management System.
fulfillmentOrderNumber	String	Contains the fulfillment order number for the transfer as provided by OMS. One or more fulfillment orders could relate back to a single customer order in OMS.
carrierCode	String	This field indicates the carrier the order is to be shipped with, if specified on the order.
carrierServiceCode	String	This field indicates method that was selected for shipping by the customer placing the order (e.g., Standard Shipping, Overnight, etc.).
consumerDeliveryDate	String	This field specifies desired date the delivery is required by the customer.
consumerDeliveryTime	String	This field specifies desired time the delivery is required by the customer.
deliverFirstName	String	For customer order transfers, this field contains the first name for the delivery address on the order. For other types of transfer this field would not be populated.
deliverPhoneticFirst	String	For customer order transfers, this field contains the phonetic first name for the delivery address on the order. For other types of transfer this field would not be populated.
deliverLastName	String	For customer order transfers, this field contains the phonetic first name for the delivery address on the order. For other types of transfer this field would not be populated.
deliverPhoneticLast	String	For customer order transfers, this field contains the phonetic last name for the delivery address on the order. For other types of transfer this field would not be populated.

Name	Data Type	Description
deliverPreferredName	String	For customer order transfers, this field contains the preferred name for the delivery address on the order. For other types of transfer this field would not be populated.
deliverCompanyName	String	For customer order transfers, this field contains the company name for the delivery address on the order. For other types of transfer this field would not be populated.
deliverAdd3	String	For customer order transfers, this field contains the third line of the delivery address of the customer. For other types of transfer this field would not be populated.
deliverCounty	String	For customer order transfers, this field contains country of the delivery address of the customer. For other types of transfer this field would not be populated.
deliverPhone	String	For customer order transfers, this field contains delivery phone number. For other types of transfer this field would not be populated.
billFirstName	String	For customer order transfers, this field contains the first name for the billing address on the order. For other types of transfer this field would not be populated.
billPhoneticFirst	String	For customer order transfers, this field contains the phonetic first name for the billing address on the order. For other types of transfer this field would not be populated.
billLastName	String	For customer order transfers, this field contains the last name for the billing address on the order. For other types of transfer this field would not be populated.
billPhoneticLast	String	For customer order transfers, this field contains the phonetic last name for the billing address on the order. For other types of transfer this field would not be populated.
billPreferredName	String	For customer order transfers, this field contains the preferred name for the billing address on the order. For other types of transfer this field would not be populated.
billCompanyName	String	For customer order transfers, this field contains the preferred name for the billing address on the order. For other types of transfer this field would not be populated.
billAdd1	String	For customer order transfers, this field contains the first line of the billing address of the customer. For other types of transfer this field would not be populated.
billAdd2	String	For customer order transfers, this field contains the second line of the billing address of the customer. For other types of transfer this field would not be populated.

Name	Data Type	Description
billAdd3	String	For customer order transfers, this field contains the third line of the billing address of the customer. For other types of transfer this field would not be populated.
billCounty	String	For customer order transfers, this field contains the county portion of the billing address of the customer. For other types of transfer this field would not be populated.
billCity	String	For customer order transfers, this field contains the city portion of the billing address of the customer. For other types of transfer this field would not be populated.
billCountry	String	For customer order transfers, this field contains the city portion of the billing address of the customer. For other types of transfer this field would not be populated.
billPost	String	For customer order transfers, this field contains the postal code portion of the billing address of the customer. For other types of transfer this field would not be populated.
billState	String	For customer order transfers, this field contains the state portion of the billing address of the customer. For other types of transfer this field would not be populated.
billPhone	String	For customer order transfers, this field contains the billing phone number. For other types of transfer this field would not be populated.
partialDeliveryInd	String	For customer order transfers, this field indicates if the order can be picked and shipped partially or if it should be shipped only when complete.
consumerDirectInd	String	This field indicates the order as being consumer direct (Y) or not (N). An order is consumer direct if it is a customer order.
customFlexAttribute		This list holds custom flex attributes for the transfers.
createDateTime	String	This column holds the record creation date.
updateDateTime	String	his column holds the date when the record was last updated.

Table 5-362 details

Name	Data Type	Description
item	String	This field specifies unique alphanumeric value that identifies the item.
transferQuantity	Number	This field specifies total quantity of the item reserved at the from location for this transfer.
price	Number	This field specifies unit retail price in the selling unit of measure for the item/location combination.

Table 5-362 (Cont.) details

Name	Data Type	Description
sellingUom	String	This field specifies selling unit of measure of the item.
expediteFlag	String	This field is determined from transfer's freight code field.
storeOrderMultiple	String	This field specifies multiple in which the item needs to be shipped from a warehouse to the location.
transferOrderLinkNo	Number	This field specifies reference number to link the item on the transfer to any purchase orders that have been created to allow the from location (i.e. warehouse) on the transfer to fulfill the transfer quantity to the to location (i.e. store) on the transfer.
ticketTypeId	String	This field specifies reference number to link the item on the transfer to any purchase orders that have been created to allow the from location (i.e., warehouse) on the transfer to fulfill the transfer quantity to the to location (i.e. store) on the transfer.
ticketDetails		This List holds attributes for details of transfer details ticket record.
invStatus	String	This field specifies inventory status, if one exists, for the transfer detail record. The inventory status field corresponds to the inventory status type found on the INV_STATUS_TYPES table
transactionUom	String	This field specifies original transaction unit of measure to be passed on to the store system. It is only used for customer direct orders coming from the Order Management System. All quantities in the response payload are expressed in item's standard unit of measure.
itemLineNo	String	This field specifies detail item line number for customer order transfers. For other types of transfer this field is not populated.
comments	String	This field specifies any comments associated with the transfer line item.
createDateTime	String	This column holds the record creation date.
updateDateTime	String	This column holds the date when the record was last updated.

Table 5-363 ticketDetails

Name	Data Type	Description
componentItem	String	This field specifies Alphanumeric value that identifies the component item within the pack.
componentPrice	Number	This field specifies unit retail price in the selling unit retail of the component item within the pack.

Table 5-363 (Cont.) ticketDetails

Name	Data Type	Description
componentSellingUom	String	This field specifies selling unit of measure for a component item within the pack.

Table 5-364 customFlexAttribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Table 5-365 Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
" {
  "items": [
    {
      "action": "INSERT",
      "transferNo": 100000005000,
      "docType": "T",
      "physicalFromLocation": 1515,
      "fromLocationType": "S",
```

```

"fromStoreType": "C",
"fromStockholdingInd": "Y",
"fromLocation": 1515,
"physicalToLocation": 8811,
"toLocationType": "S",
"toStoreType": "C",
"toStockholdingInd": "Y",
"toLocation": 8811,
"transferType": "MR",
"pickNotBeforeDate": "2021-12-08",
"pickNotAfterDate": "2022-01-07",
"orderType": "AUTOMATIC",
"breakByDistro": "N",
"deliveryDate": "2021-12-08",
"deliverAdd1": null,
"deliverAdd2": null,
"deliverCity": null,
"deliverState": null,
"deliverPost": null,
"deliverCountryId": null,
"comments": null,
"details": [
  {
    "item": "100500002",
    "transferQuantity": 1,
    "price": 12.22,
    "sellingUom": "EA",
    "expediteFlag": "N",
    "storeOrderMultiple": "E",
    "transferOrderLinkNo": null,
    "ticketTypeId": null,
    "ticketDetails": null,
    "invStatus": null,
    "transactionUom": null,
    "itemLineNo": null,
    "comments": null,
    "createDateTime": "2021-12-08T18:16:04.000Z",
    "updateDateTime": "2021-12-08T18:16:10.000Z"
  }
],
"transferParentNo": null,
"expectedDcDate": "2021-12-08",
"approvalId": "RMS_ADMIN",
"approvalDate": "2021-12-08",
"fromLocationTransferEntity": 1111,
"toLocationTransferEntity": 1111,
"invType": "A",
"transferStatus": "A",
"notAfterDate": "2021-12-10",
"contextType": null,
"contextValue": null,
"deliverySlotId": null,
"deliverySlotDescription": null,
"customerOrderNo": null,
"fulfillmentOrderNumber": null,
"carrierCode": null,

```

```

"carrierServiceCode": null,
"consumerDeliveryDate": null,
"consumerDeliveryTime": null,
"deliverFirstName": null,
"deliverPhoneticFirst": null,
"deliverLastName": null,
"deliverPhoneticLast": null,
"deliverPreferredName": null,
"deliverCompanyName": null,
"deliverAdd3": null,
"deliverCounty": null,
"deliverPhone": null,
"billFirstName": null,
"billPhoneticFirst": null,
"billLastName": null,
"billPhoneticLast": null,
"billPreferredName": null,
"billCompanyName": null,
"billAdd1": null,
"billAdd2": null,
"billAdd3": null,
"billCounty": null,
"billCity": null,
"billCountry": null,
"billPost": null,
"billState": null,
"billPhone": null,
"partialDeliveryInd": null,
"consumerDirectInd": "N",
"customFlexAttribute": null,
"createDateTime": "2021-12-08T00:00:00.000Z",
"updateDateTime": "2021-12-08T18:16:24.000Z",
"cacheTimestamp": "2022-04-06T10:00:47.220Z"
},
{
  "action": "INSERT",
  "transferNo": 100000015002,
  "docType": "T",
  "physicalFromLocation": 440,
  "fromLocationType": "W",
  "fromStoreType": null,
  "fromStockholdingInd": null,
  "fromLocation": 441,
  "physicalToLocation": 2345,
  "toLocationType": "E",
  "toStoreType": null,
  "toStockholdingInd": null,
  "toLocation": 2345,
  "transferType": "MR",
  "pickNotBeforeDate": "2021-12-20",
  "pickNotAfterDate": "2022-01-19",
  "orderType": "AUTOMATIC",
  "breakByDistro": "N",
  "deliveryDate": "2021-12-21",
  "deliverAdd1": null,
  "deliverAdd2": null,

```

```

"deliverCity": null,
"deliverState": null,
"deliverPost": null,
"deliverCountryId": null,
"comments": null,
"details": [
  {
    "item": "1005000002",
    "transferQuantity": 3,
    "price": 12.22,
    "sellingUom": "EA",
    "expediteFlag": "N",
    "storeOrderMultiple": "E",
    "transferOrderLinkNo": null,
    "ticketTypeId": null,
    "ticketDetails": null,
    "invStatus": null,
    "transactionUom": null,
    "itemLineNo": null,
    "comments": null,
    "createDateTime": "2021-12-21T14:07:46.000Z",
    "updateDateTime": "2021-12-21T14:07:50.000Z"
  }
],
"transferParentNo": null,
"expectedDcDate": "2021-12-21",
"approvalId": "RMS_ADMIN",
"approvalDate": "2021-12-20",
"fromLocationTransferEntity": 1111,
"toLocationTransferEntity": 1111,
"invType": "A",
"transferStatus": "A",
"notAfterDate": "2021-12-22",
"contextType": null,
"contextValue": null,
"deliverySlotId": null,
"deliverySlotDescription": null,
"customerOrderNo": null,
"fulfillmentOrderNumber": null,
"carrierCode": null,
"carrierServiceCode": null,
"consumerDeliveryDate": null,
"consumerDeliveryTime": null,
"deliverFirstName": null,
"deliverPhoneticFirst": null,
"deliverLastName": null,
"deliverPhoneticLast": null,
"deliverPreferredName": null,
"deliverCompanyName": null,
"deliverAdd3": null,
"deliverCounty": null,
"deliverPhone": null,
"billFirstName": null,
"billPhoneticFirst": null,
"billLastName": null,
"billPhoneticLast": null,

```

```

        "billPreferredName": null,
        "billCompanyName": null,
        "billAdd1": null,
        "billAdd2": null,
        "billAdd3": null,
        "billCounty": null,
        "billCity": null,
        "billCountry": null,
        "billPost": null,
        "billState": null,
        "billPhone": null,
        "partialDeliveryInd": null,
        "consumerDirectInd": "N",
        "customFlexAttribute": null,
        "createDateTime": "2021-12-20T00:00:00.000Z",
        "updateDateTime": "2021-12-21T14:08:42.000Z",
        "cacheTimestamp": "2022-04-06T10:00:48.823Z"
    }
},
"hasMore": true,
"limit": 2,
"count": 2,
"links": [
    {
        "href": "https://rex.retail.us-phoenix-1.ocs.oc-test.com/rgbu-rex-rgbu-
stg83-mfcs/MerchIntegrations/services/inventory/transfer?
limit=2&since=1970-01-01T00:00:00.001Z&before=2022-04-10T15:26:38.684995+00:00
",
        "rel": "self"
    },
    {
        "href": "https://rex.retail.us-phoenix-1.ocs.oc-test.com/rgbu-rex-rgbu-
stg83-mfcs/MerchIntegrations/services/inventory/transfer?
offsetkey=100000015002&limit=2&since=1970-01-01T00:00:00.001Z&before=2022-04-1
0T15:26:38.684995+00:00",
        "rel": "next"
    }
]
}"

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	Yes	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_TRANSFER	Yes	No	Yes	Yes
MERCHAPI_EXT_TRANSFER	Yes	Yes	Yes	Yes
MERCHAPI_EXT_TRANSFER_CFA	No	Yes	No	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
TSFHEAD	Yes	No	No	No
TSFHEAD_CFA_EXT	Yes	No	No	No
V_MERCHAPI_EXT_TSF_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

Transfer Upload Service

The following services are included in this functional area:

- [Create Book Transfer](#)
- [Create Transfer Details](#)
- [Create Transfers](#)
- [Delete Transfer Details](#)
- [Delete Transfers](#)
- [Update Transfer Details](#)
- [Update Transfers](#)

Create Book Transfer

This section describes the Book Transfer ReSTful Web Service

Functional Area

Transfer and Customer Order

Business Overview

This web service will be built to virtually move inventory from one location to the other for the purposes of attributing the sale to a location different from the location that is fulfilling the order physically. For example, if the order is being fulfilled via shipment from a physical store, some retailers will want to actually have the sale processed against the e-commerce store. This service also accept a customer order number and fulfillment order number to be associated with the transfer when it is created as a cross reference.

Service Type

Post

ReST URL

/Transfer/customerOrderBookTransfer

Input Parameters

The Book Transfer web service has the following parameters:

Parameter Name	Required	Data Type	Description
FromLocation	Yes	BigDecimal	Transfer source location
ToLocation	Yes	BigDecimal	Transfer destination
CustomerOrderNumber	No	String	Customer order identification
FulfillOrderNumber	No	String	Fulfillment order identification

Parameter Name	Required	Data Type	Description
UpdateCustomerReservedQty	No	String	Indicates if any of the customer reserved quantity should be update either the source or destination location or both. Valid values: • B update both the source and destination location • S update only the source location • R update only the destination location • N or NULL no update
ItemsDetail	Yes		Collection of itemsDetail RDO
ApprovalDate	No	Date	Approval date of the transfer (Format: 'DD-MON-YYYY')

ItemDetail RDO

The Book Transfer web service has the following parameters:

Parameter Name	Required	Data Type	Description
Item	Yes	String	Item identification
Quantity	Yes	BigDecimal	Item quantity to be transferred

Example JSON Input

```
[
  {
    "toLocation":null,
    "fromLocation":null,
    "customerOrderNumber":null,
    "fulfillOrderNumber":null,
    "updateCustomerReservedQty":null,
    "itemsDetail":[
      {
        "item":null,
        "quantity":null
      },
      {
        "item":null,
        "quantity":null
      },
      {
        "item":null,
        "quantity":null
      }
    ],
    "approvalDate":null
  },
  {
    "fromLocation":null,
    "toLocation":null,
    "customerOrderNumber":null,
    "fulfillOrderNumber":null,
```



```

    "updateCustomerReservedQty":null,
    "itemsDetail":[
      {
        "item":null,
        "quantity":null
      }
    ],
    "approvalDate":null
  }
]

```

Output

RestCobtsfStatuRDO

Parameter Name	Data Type
successCobtsfCount	Big Decimal
successCobtsfTbl	List< successCobtsfRDO>
failCobtsfCount	BigDecimal
failCobtsfTable	List<failCobtsfRDO>

SuccessCobtsfRDO

Parameter Name	Data Type
FromLocation	Big Decimal
ToLocation	Big Decimal
TransferNumber	BigDecimal

FailCobtsfRDO

Parameter Name	Data Type
FromLocation	Big Decimal
ToLocation	Big Decimal
errorMessage	BigDecimal

JSON Structure

```

{
  "successCobtsfCount": 2,
  "successCobtsfTable": [
    {
      "fromLocation ": 123,
      "toLocation ": 987,
      "transferNumber ": 123456789
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    },
    {
      "fromLocation ": 456,

```

```

        "toLocation ": 654,
        "transferNumber ": 987654321
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    },
    ],
    "failCobtsfCount": 1,
    "failCobtsfTable": [
        {
            "orderNumber": 123,
            "errorMessage": "Invalid Item.",
            "links": [],
            "hyperMediaContent": {
                "linkRDO": []
            }
        }
    ],
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}

```

Table Impact

The following tables are affected:

TABLE	SELECT	INSERT	UPDATE	DELETE
CURRENCIES	Yes	No	No	No
DEPS	Yes	No	No	No
ITEM_LOC	Yes	Yes	No	No
ITEM_LOC_SOH	Yes	Yes	Yes	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
MONTH_DATA	Yes	No	No	No
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No
ORDCUST	Yes	Yes	No	No
STORE	Yes	No	No	No
TRAN_DATA	No	Yes	No	No
TSFDETAIL	No	Yes	No	No
TSFHEAD	No	No	No	No
UOM_CLASS	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
WEEK_DATA	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Transfer Details

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service is used to create additional line items for one or more existing transfers at a time in Merchandising. For more information about transfer, see the Create Transfer service.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/transfer/details/create

Input Payload Details

Table 5-366 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of transfers.

Table 5-367 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	Number that uniquely identifies the transfer.
fromLocationType	Yes	String (1)	The location type of the from location. Valid values are 'S' and 'W'.
fromLocation	Yes	String (10)	The location number of the from location.
toLocationType	Yes	String (1)	The location type of the to location. Valid values are 'S', 'W' and 'E'
toLocation	Yes	String (10)	The location number of the to location.
details	No	Collection of Object	References collection of transfer details

Table 5-368 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the item being transferred.
transferQuantity	Yes	Number (12,4)	The total quantity of the item reserved for this transfer at the from location.
supplierPackSize	No	Number (12,4)	The supplier pack size for this item/transfer.

Table 5-368 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
invStatus	No	Number (2)	A code indicating the inventory status for this transfer detail. Valid values are found on the inv_status_types table.
adjustmentType	No	String (6)	This field, along with the adjustment value, is used to calculate the transfer price for intercompany transfers. It will be ignored for all other transfers. If the adjustment value is provided, then the type must also be specified. Valid values are: 'IA', 'IP', 'DA', 'DP', 'S'. IA and IP can only be used if you have your system options set to allow the transfer price to exceed weighted average cost.
adjustmentValue	No	Number (20,4)	If the adjustment type is provided, then the value must also be specified. This must always be a positive amount.

Sample Input Message

```
{
  "items": [
    {
      "transferNo": 12300056,
      "fromLocationType": "S",
      "fromLocation": "1521",
      "toLocationType": "S",
      "toLocation": "1511",
      "details": [
        {
          "item": "100100076",
          "transferQuantity": 3,
          "supplierPackSize": 3,
          "invStatus": 1,
          "adjustmentType": "IA",
          "adjustmentValue": 3.55
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
```

```
"message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Create Transfers**Functional Area**

Inventory - Transfers and Allocations

Business Overview

This service is used to by external systems to create one or more transfers at a time in Merchandising. Within Oracle Retail solutions, this is also leveraged by Advanced Inventory Planning (AIP) to create standalone transfers generated out of its replenishment processing.

When a new transfer is created, this service will first validate that all required fields are present in the message. Certain fields are required regardless of transfer type and system configuration, while others are dependent on other Merchandising configurations. Additionally, when creating a new transfer at least one detail line must also be included in the message. After that, business level validation on the input information will be performed.

Location Validation

The from and to locations passed into the message must be valid stores or warehouses in Merchandising; but they cannot be the same. If both locations are stores, then they must both exist in the same transfer zone. Additionally, if the to location is a store, then it must be open. This is determined based on whether there is a close date defined for the store and the stop order days.

If either location is a warehouse, then it can be either a physical warehouse or a virtual warehouse, depending on transfer type. A physical warehouse is only allowed as the from location type for an EG type of transfer. Additionally, only Book type transfers are allowed between two warehouses in the same physical warehouse.

If either the from or to location is a franchise store, then the other location cannot be a finisher. If the franchise store is a non-stockholding location, then the other location on the transfer must be a warehouse.

Validation is also done at the item level based on the locations on the transfer. Each item on the transfer must be in active, inactive, or discontinued status at the from location. It also must have been ranged to the from location in Merchandising, when that location is a warehouse. However, if the from location is a store, there is an exception where the transfer can still be created even though it is not yet ranged, which also bypasses inventory validation. This is to support a specific function in Oracle Retail Store Inventory Management (SIM). If the item is not already ranged to the to location, then ranging will occur when the transfer is created, regardless of status. The ranging that occurs will flag the item/location as unintentionally ranged for all transfer types except service. If the to location is an external finisher, see the section below on transfers with finishing.

Inventory Validation

Another part of the validation that is applicable for all transfers created is that inventory is available for transfer if the status passed through the integration is approve (A), with a few exceptions. First, EG type transfers do not have inventory validated as it is assumed that this type of transfer is generated in the store or warehouse and the inventory availability check has been done in that solution as part of the shipping of the inventory. Additionally, if the system option titled Validate External Warehouse Availability is set to No (unchecked), then warehouse inventory will not be validated for any transfers initiated in this service regardless of type. Store inventory availability is never validated by this service because of support for the process where the item does not need to be ranged to the shipping store.

Status Validation

Transfers can be created in a status of Input (I) or Approved (A) using this service. Transfers in input status are not subject to inventory validation, but all other validations are applicable. Book type transfers can only be created in Input status using this service, as there isn't really a concept of an 'approved' book transfer. As soon as it is approved it is executed. Additionally, transfers of type Reallocation (RAC) and Return to Vendor (RV) can also only be created in Input status. Conversely, transfers of type AIP, SIM, and EG must always be created in Approved status. If any validation fails when processing the new transfer that results in it not being able to be approved, the transfer will be created but will remain in input status. The exception to this is for transfers of type AIP, SIM, and EG, as they must always be created in approved status. If they are not able to be approved, the transfer is not created or updated.

Transfer Type Specific Validation

Most of the validation defined above is relevant regardless of transfer type, except where noted. However, there are also some other validations done as part of this service processing that are specific to a type of transfer.

Administrative (AD)

See Manual Requisition

AIP Generated (AIP)

This type of transfer is expected only to be sent from AIP as an output of the replenishment process. As such, Merchandising assumes certain validations have been done by AIP in advance of receiving the transfer and slightly different validation is enforced. The following special validations apply for this transfer type using this service:

Must be created in Approved status

Can only be to stockholding locations

Supports transferring packs from stores

Allows the department number to be passed even when the system option is N

Item/location ranging to the to location will result in the Ranged flag being set to Yes as it is assumed this an intentional ranging.

Can be an intercompany transfer

Book (BT)

Book transfers processed through this service can be created for two virtual warehouses in the same physical warehouse only. This is usually used for inventory rebalancing between virtual locations. The following special validations apply for this transfer type using this service:

Can only be created in Input status

Can only be created for virtual warehouses in the same physical warehouse

Warehouses must be in the same legal entity

Confirmation (CF)

See Manual Requisition

Externally Generated (EG)

Externally Generated transfers are assumed to be created in the store or warehouse. Further, it is assumed that once they get to Merchandising, the transfer is already in process at that location. As such, there are certain validations that are managed differently for this transfer type in this service:

Must be created in Approved status

Supports transferring packs from stores

Allows the department number to be passed even when the system option is N

Can be an intercompany transfer

Uses the physical warehouse number, not a virtual warehouse number, if warehouses are involved

Intercompany (IC)

An intercompany transfer is a type of business to business transaction that sells product from one legal entity and purchases it into another. Legal entities in Merchandising are determined based on the setting of the Intercompany Basis system option, which indicates whether the transfer entity or the set of books of a location should be used. This transfer type is used when either it is explicitly passed into the service or if the transfer type is NULL in the inbound message and the locations are in different legal entities. Other transfer types may also be intercompany, as well, but the below rules apply for those flagged as intercompany type explicitly:

The legal entity of the from and to locations must be different.

If an adjustment type or value is passed into the message, that will be used to calculate the 'selling' price between entities. Otherwise, the from location's weighted average cost is used

Manual Requisition (MR)

This is the most basic type of transfer in Merchandising, so it is used as a default transfer type when either it is explicitly passed into the service or if the transfer type is NULL in the inbound message and the locations are in the same legal entity. The behavior for this transfer type is

the same as that for AD and CF types of transfers, those could be used as different reasons for a transfer. For this transfer type the following validation rules are enforced:

Locations must be in the same legal entity

Reallocation (RAC)

A reallocation transfer is assumed to be used to pull back inventory from stores or warehouses to a single warehouse for re-allocation to other stores or other warehouses. This is the type of transfer that is created when a mass-return transfer is created, for example. Because it has unique rules tied to it related to MRTs, some additional validations are followed:

Can only be created in Input status in this service

Locations must be in the same legal entity

Return to Vendor (RV)

A return to vendor type of transfer is similar to a reallocation type, in that it is assumed to be pulling inventory back to a warehouse from stores or other warehouses, but in this case, for the purpose of returning the merchandise to the supplier. This is the type of transfer that is created when a mass-return transfer is created, for example. Because it has some unique rules tied to it related to MRTs, some additional validations are followed:

Can only be created in Input status in this service

Locations must be in the same legal entity

SIM Generated (SIM)

SIM generated transfers are created only by the store orders process in SIM. This functionality is not available in SIOCS. Because of this, they have special rules applied, including the ability to create the transfer even though no item/store relationship exists for the originating location in Merchandising. The rules that apply for this type of transfer include:

Must be created in Approved status

Supports transferring packs from stores

Allows the department number to be passed even when the Merchandising system option is No

Can be an intercompany transfer

All Transfer Types

For all of the above transfer types, if all validation described above passes, then the transfer will be created. If the transfer is created in Approved status, then in addition to the transfer itself, other details may also be created based on the items and locations involved.

Inventory will be updated to reflect the reserved quantity at the from location and expected quantity at the to location.

Upcharges will be applied, if configured, for transfers that do not include a physical warehouse location. For transfers with a physical warehouse, the records for upcharges are added when the transfer is shipped.

An associated franchise order or return will be created if the transfer involves a franchise location.

Transfers with Finishing

Transfers with finishing are sometimes referred to as a two-legged transfer, as they generate two transfers in Merchandising. One from the originating store or warehouse to the finisher and one from the finisher back to a store or warehouse. This service supports the creation of a transfer with finishing only through an external finisher, a type of partner, and back to the originating location. Transfers to an internal finisher are not supported via this integration. To do this, when sending the transfer details in the message, you will indicate the external finisher as the to location. Then when the transfer is created, it will automatically generate the second leg.

When creating transfers in this way, it does not generate any work order activities to send to the finisher with the transfer. These will either need to be added manually in the Merchandising screens, or sent separately to the finisher.

Flex Attributes

If you have defined any custom flex attributes (CFAS) for transfers, then they can be integrated as part of this service. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute. Flex attributes can only be added or updated to a transfer, they cannot be deleted. Additionally, for transfers with finishing, flex attributes can only be added to the first leg of the transfer.

Location Validation

The from and to locations passed into the message must be valid stores or warehouses in Merchandising; but they cannot be the same. If both locations are stores, then they must both exist in the same transfer zone. Additionally, if the to location is a store, then it must be open. This is determined based on whether there is a close date defined for the store and the stop order days.

If either location is a warehouse, then it can be either a physical warehouse or a virtual warehouse, depending on transfer type. A physical warehouse is only allowed as the from location type for an EG type of transfer. Additionally, only Book type transfers are allowed between two warehouses in the same physical warehouse.

If either the from or to location is a franchise store, then the other location cannot be a finisher. If the franchise store is a non-stockholding location, then the other location on the transfer must be a warehouse.

Validation is also done at the item level based on the locations on the transfer. Each item on the transfer must be in active, inactive, or discontinued status at the from location. It also must have been ranged to the from location in Merchandising, when that location is a warehouse. However, if the from location is a store, there is an exception where the transfer can still be created even though it is not yet ranged, which also bypasses inventory validation. This is to support a specific function in Oracle Retail Store Inventory Management (SIM). If the item is not already ranged to the to location, then ranging will occur when the transfer is created, regardless of status. The ranging that occurs will flag the item/location as unintentionally ranged for all transfer types except service. If the to location is an external finisher, see the section below on transfers with finishing.

Inventory Validation

Another part of the validation that is applicable for all transfers created is that inventory is available for transfer if the status passed through the integration is approve (A), with a few exceptions. First, EG type transfers do not have inventory validated as it is assumed that this type of transfer is generated in the store or warehouse and the inventory availability check has been done in that solution as part of the shipping of the inventory. Additionally, if the system option titled Validate External Warehouse Availability is set to No (unchecked), then warehouse inventory will not be validated for any transfers initiated in this service regardless of type. Store inventory availability is never validated by this service because of support for the process where the item does not need to be ranged to the shipping store.

Status Validation

Transfers can be created in a status of Input (I) or Approved (A) using this service. Transfers in input status are not subject to inventory validation, but all other validations are applicable. Book type transfers can only be created in Input status using this service, as there isn't really a concept of an 'approved' book transfer. As soon as it is approved it is executed. Additionally, transfers of type Reallocation (RAC) and Return to Vendor (RV) can also only be created in Input status. Conversely, transfers of type AIP, SIM, and EG must always be created in Approved status. If any validation fails when processing the new transfer that results in it not being able to be approved, the transfer will be created but will remain in input status. The exception to this is for transfers of type AIP, SIM, and EG, as they must always be created in approved status. If they are not able to be approved, the transfer is not created or updated.

Transfer Type Specific Validation

Most of the validation defined above is relevant regardless of transfer type, except where noted. However, there are also some other validations done as part of this service processing that are specific to a type of transfer.

Administrative (AD)

- See [Manual Requisition \(MR\)](#)

Service Type

This type of transfer is expected only to be sent from AIP as an output of the replenishment process. As such, Merchandising assumes certain validations have been done by AIP in advance of receiving the transfer and slightly different validation is enforced. The following special validations apply for this transfer type using this service:

- Must be created in Approved status
- Can only be to stockholding locations
- Supports transferring packs from stores
- Allows the department number to be passed even when the system option is N
- Item/location ranging to the to location will result in the Ranged flag being set to Yes as it is assumed this an intentional ranging.
- Can be an intercompany transfer

Book (BT)

Book transfers processed through this service can be created for two virtual warehouses in the same physical warehouse only. This is usually used for inventory rebalancing between virtual locations. The following special validations apply for this transfer type using this service:

- Can only be created in Input status
- Can only be created for virtual warehouses in the same physical warehouse
- Warehouses must be in the same legal entity

Confirmation (CF)

- See [Manual Requisition \(MR\)](#)

Externally Generated (EG)

Externally Generated transfers are assumed to be created in the store or warehouse. Further, it is assumed that once they get to Merchandising, the transfer is already in process at that location. As such, there are certain validations that are managed differently for this transfer type in this service:

- Must be created in Approved status
- Supports transferring packs from stores

- Allows the department number to be passed even when the system option is N
- Can be an intercompany transfer
- Uses the physical warehouse number, not a virtual warehouse number, if warehouses are involved

Intercompany (IC)

An intercompany transfer is a type of business to business transaction that sells product from one legal entity and purchases it into another. Legal entities in Merchandising are determined based on the setting of the Intercompany Basis system option, which indicates whether the transfer entity or the set of books of a location should be used. This transfer type is used when either it is explicitly passed into the service or if the transfer type is NULL in the inbound message and the locations are in different legal entities. Other transfer types may also be intercompany, as well, but the below rules apply for those flagged as intercompany type explicitly:

- The legal entity of the from and to locations must be different.
- If an adjustment type or value is passed into the message, that will be used to calculate the 'selling' price between entities. Otherwise, the from location's weighted average cost is used

Manual Requisition (MR)

This is the most basic type of transfer in Merchandising, so it is used as a default transfer type when either it is explicitly passed into the service or if the transfer type is NULL in the inbound message and the locations are in the same legal entity. The behavior for this transfer type is the same as that for AD and CF types of transfers, those could be used as different reasons for a transfer. For this transfer type the following validation rules are enforced:

- Locations must be in the same legal entity

Reallocation (RAC)

A reallocation transfer is assumed to be used to pull back inventory from stores or warehouses to a single warehouse for re-allocation to other stores or other warehouses. This is the type of transfer that is created when a mass-return transfer is created, for example. Because it has unique rules tied to it related to MRTs, some additional validations are followed:

- Can only be created in Input status in this service
- Locations must be in the same legal entity

Return to Vendor (RV)

A return to vendor type of transfer is similar to a reallocation type, in that it is assumed to be pulling inventory back to a warehouse from stores or other warehouses, but in this case, for the purpose of returning the merchandise to the supplier. This is the type of transfer that is created when a mass-return transfer is created, for example. Because it has some unique rules tied to it related to MRTs, some additional validations are followed:

- Can only be created in Input status in this service
- Locations must be in the same legal entity

SIM Generated (SIM)

SIM generated transfers are created only by the store orders process in SIM. This functionality is not available in SIOCS. Because of this, they have special rules applied, including the ability to create the transfer even though no item/store relationship exists for the originating location in Merchandising. The rules that apply for this type of transfer include:

- Must be created in Approved status
- Supports transferring packs from stores

- Allows the department number to be passed even when the Merchandising system option is No
- Can be an intercompany transfer

All Transfer Types

For all of the above transfer types, if all validation described above passes, then the transfer will be created. If the transfer is created in Approved status, then in addition to the transfer itself, other details may also be created based on the items and locations involved.

- Inventory will be updated to reflect the reserved quantity at the from location and expected quantity at the to location.
- Upcharges will be applied, if configured, for transfers that do not include a physical warehouse location. For transfers with a physical warehouse, the records for upcharges are added when the transfer is shipped.
- An associated franchise order or return will be created if the transfer involves a franchise location.

Transfers with Finishing

Transfers with finishing are sometimes referred to as a two-legged transfer, as they generate two transfers in Merchandising. One from the originating store or warehouse to the finisher and one from the finisher back to a store or warehouse. This service supports the creation of a transfer with finishing only through an external finisher, a type of partner, and back to the originating location. Transfers to an internal finisher are not supported via this integration. To do this, when sending the transfer details in the message, you will indicate the external finisher as the to location. Then when the transfer is created, it will automatically generate the second leg.

When creating transfers in this way, it does not generate any work order activities to send to the finisher with the transfer. These will either need to be added manually in the Merchandising screens, or sent separately to the finisher.

Flex Attributes

If you have defined any custom flex attributes (CFAS) for transfers, then they can be integrated as part of this service. The node of the integration that supports this will accept the name of the attribute as it is defined in the group set level view and the value for the attribute. Flex attributes can only be added or updated to a transfer, they cannot be deleted. Additionally, for transfers with finishing, flex attributes can only be added to the first leg of the transfer.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/transfers/create

Input Payload Details

Table 5-369 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of transfers.

Table 5-370 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	Number that uniquely identifies the transfer.
fromLocationType	Yes	String (1)	The location type of the from location. Valid values are 'S' and 'W'.
fromLocation	Yes	Number (10)	The location number of the from location.
toLocationType	Yes	String (1)	The location type of the to location. Valid values are 'S', 'W' and 'E'
toLocation	Yes	Number (10)	The location number of the to location.
deliveryDate	No	date	The earliest date the transfer can be delivered. When Inventory Planning is part of your implementation, this is required for all transfer types, except RAC, EG, and Store Inventory Operations transfers. If included in the message, this must be today or a future date.
expectedDcDate	No	date	The date that the transfer is expected to be shipped at the warehouse. If included in the message, this must be today or a future date.

Table 5-370 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
department	No	Number (4)	The department number associated with the transfer. A system option determines whether or not the department is required for transfers. If the system option is set to require a department, then this must be included in the message. If the system option is set to not require the department, then the department must be null in this message unless the transfer type is SIM (Store Inventory Operations), AIP (Inventory Planning), or EG (Externally Generated).
routingCode	No	String (1)	If the freight code is Expedite (E), then this must have a value. Otherwise, it must be null. The descriptions for these three options are held in the Codes table under code TRRC and can be configured as needed for your business. Examples are Federal Express (1), UPS (2) and mail (3).
freightCode	No	String (1)	A code indicating the freight status of the transfer (e. g. normal, expedite, etc.).

Table 5-370 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferType	No	String (6)	A code indicating the type of transfer (e. g. store requisition, book transfer, etc.). The following types of transfers can be created in this service: Administrative (AD), Inventory Planning Generated (AIP), Book (BT), Confirmation (CF), Externally Generated (EG), Intercompany (IC), Manual Requisition (MR), Reallocation (RAC), Return to Vendor ,(RV) and Store Inventory Operations Generated (SIM).
details	No	Collection of Object	References a collection of transfer details.
status	No	String (1)	A code indicating the status of the transfer. Valid values are 'I' and 'A'.
createdBy	No	String (30)	The userid of the user who created the transfer.
comments	No	String (2000)	Comments associated with the transfer.
contextType	No	String (6)	This field holds the reason code related to which a transfer is made. Valid values are found in the code type CNTX. Examples are Promotion (PROM), Customer Transfer (WED), Store Requisition (STORE) and Repairing (REPAIR).
contextValue	No	String (25)	This field holds the value relating to the context type like Promotion number.

Table 5-370 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Table 5-371 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the item being transferred.
transferQuantity	Yes	Number (12,4)	The total quantity of the item reserved for this transfer at the from location.
supplierPackSize	No	Number (12,4)	The supplier pack size for this item/transfer.
invStatus	No	Number (2)	A code indicating the inventory status for this transfer detail. Valid values are found on the inv_status_types table.

Table 5-371 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
adjustmentType	No	String (6)	This field, along with the adjustment value, is used to calculate the transfer price for intercompany transfers. It will be ignored for all other transfers. If the adjustment value is provided, then the type must also be specified. Valid values are: 'IA', 'IP', 'DA', 'DP', 'S'. IA and IP can only be used if you have your system options set to allow the transfer price to exceed weighted average cost.
adjustmentValue	No	Number (20,4)	If the adjustment type is provided, then the value must also be specified. This must always be a positive amount.

Table 5-372 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "items": [
    {
      "transferNo": 12300056,
      "fromLocationType": "S",
      "fromLocation": 1521,
      "toLocationType": "S",
      "toLocation": 1511,
      "deliveryDate": "2001-12-31",
      "expectedDcDate": "2001-12-31",
      "department": null,

```

```

"routingCode": null,
"freightCode": "N",
"transferType": "AD",
"details": [
{
"item": "104100022",
"transferQuantity": 3,
"supplierPackSize": 3,
"invStatus": 1,
"adjustmentType": "IA",
"adjustmentValue": 3.6
}
],
"status": "I",
"createdBy": "RMS01APP",
"comments": "transfer to loc 1511",
"contextType": "PROM",
"contextValue": "2255",
"customFlexAttribute": [
{
"name": "REVIEW_DATE",
"value": null,
"valueDate": "2001-12-31"
}
]
}
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

```
    ],  
    "businessError": [  
      "Error message"  
    ]  
  }  
}
```

Delete Transfer Details

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service is used to delete line items from one or more existing transfers at a time in Merchandising.

If you are deleting a detail on the transfer, this service will validate that the transfer number is valid and that the transfer or transfer line was not already shipped or received, at least partially, or is not in process at the shipping warehouse or store. Validation will be done to ensure that the item exists on the transfer after transfer line level deletes have been performed. Deleting the last line on the transfer will result in the transfer being flagged for delete.

If the transfer is in a status other than input, deleting a line will also update inventory to release the reserved inventory at the from location and decrease expected quantity at the to location. As well, if the transfer involves any franchise stores, then any franchise order or return created with the transfer will also be cancelled.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/transfer/details/delete

Input Payload Details

Table 5-373 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of transfers.

Table 5-374 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	The transfer number where info is being deleted from.
details	No	Collection of Object	References collection of transfer details

Table 5-375 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item for which transfer info is being deleted.

Sample Input Message

```
{
  "items": [
    {
      "transferNo": 12300056,
      "details": [
        {
          "item": "100100076"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Delete Transfers

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service is used to delete one or more transfers from Merchandising.

When deleting a transfer, this API will validate that the transfer number is valid and that the transfer or transfer line has not already shipped or received already or is not undergoing any processing at the shipping warehouse or store. Transfers are not actually deleted via this API, rather they are updated to a deleted status and a secondary process does the actual removal. If you want to delete the entire transfer, then no detail level records should be included in the message.

Transfers can be deleted in any status, other than ones that are already in a closed or deleted status. If the transfer involves an external finisher, then both legs of the transfer will be marked for deletion. If the transfer is in a status other than Input, then moving it to a deleted status will also update the related inventory data to release the reserved inventory at the source location and decrease the expected quantity at the destination location. If the transfer involves any franchise stores, then the franchise order or return that was created with the transfer will also be cancelled.

Service Type

DELETE

ReST URL

MerchIntegrations/services/inventory/transfers/delete

Input Payload Details

Table 5-376 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of transfers.

Table 5-377 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	This is the transfer being deleted.

Sample Input Message

```
{
  "items": [
    {
      "transferNo": 12300001
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Update Transfer Details

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service is used to modify line items for one or more existing transfers at a time in Merchandising.

To add or update a line item - only the transfer number is required in the header record, the other details are ignored. If not included, then the message will be rejected. Adding a new item to the transfer will use similar validations when creating the transfer.

If modifying an existing transfer line item, the full transfer quantity should be sent with the update, not the difference from the original quantity. This will be compared to the previous transfer quantity to determine how to update the transfer. For example, if the transfer is in approved or submitted status, a reduction in quantity would update the cancelled quantity on the transfer. It will also be validated to ensure that the quantity change doesn't result in the total transfer quantity being lower than what has already been shipped or what is expected to be picked based on updates to the selected or distro quantities on the transfer. For increases in transfer quantity, if the transfer is in submitted or approved status, then inventory will be validated based on the changed quantity (depending on system option settings) to validate that the additional units are available. The inventory status for the item cannot be modified.

For more information about transfer, see the Create Transfer service.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/transfer/details/update

Input Payload Details

Table 5-378 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of transfers.

Table 5-379 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	Number that uniquely identifies the transfer.
fromLocationType	Yes	String (1)	The location type of the from location. Valid values are 'S' and 'W'.
fromLocation	Yes	Number (10)	The location number of the from location.
toLocationType	Yes	String (1)	The location type of the to location. Valid values are 'S', 'W' and 'E'
toLocation	Yes	Number (10)	The location number of the to location.
details	No	Collection of Object	References collection of transfer details

Table 5-380 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the item being transferred.
transferQuantity	Yes	Number (12,4)	The total quantity of the item reserved for this transfer at the from location.
supplierPackSize	No	Number (12,4)	The supplier pack size for this item/transfer.
invStatus	No	Number (2)	A code indicating the inventory status for this transfer detail. Valid values are found on the inv_status_types table.

Table 5-380 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
adjustmentType	No	String (6)	This field, along with the adjustment value, is used to calculate the transfer price for intercompany transfers. It will be ignored for all other transfers. If the adjustment value is provided, then the type must also be specified. Valid values are: 'IA', 'IP', 'DA', 'DP', 'S'. IA and IP can only be used if you have your system options set to allow the transfer price to exceed weighted average cost.
adjustmentValue	No	Number (20,4)	If the adjustment type is provided, then the value must also be specified. This must always be a positive amount.

Sample Input Message

```
{
  "items": [
    {
      "transferNo": 12300056,
      "fromLocationType": "S",
      "fromLocation": 1521,
      "toLocationType": "S",
      "toLocation": 1511,
      "details": [
        {
          "item": "104100022",
          "transferQuantity": 3,
          "supplierPackSize": 3,
          "invStatus": 1,
          "adjustmentType": "IA",
          "adjustmentValue": 3.55
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Update Transfers

Functional Area

Inventory - Transfers and Allocations

Business Overview

This service is used to update transfers in Merchandising at the header level. The transfer number included in the message must already exist in Merchandising. All the required header level information needs to be included in the update, similar to that for creating a new transfer. Fields that can be updated at the header level using this service include:

- Delivery Date - Must always be a date today or later.
- Routing Code - If the freight code is updated to expedite (E), then this must also have a value. If freight code is updated to something other than expedite, then this should be null.
- Freight Code
- Status - Used to move from Input or Submitted to Approved only. Transfers cannot be moved back to Input status using this API.
- Comments
- Context Type
- Context Value

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/transfers/update

Input Payload Details

Table 5-381 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of transfers.

Table 5-382 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	Number that uniquely identifies the transfer.
fromLocationType	Yes	String (1)	The location type of the from location. Valid values are 'S' and 'W'.
fromLocation	Yes	String (10)	The location number of the from location.
toLocationType	Yes	String (1)	The location type of the to location. Valid values are 'S', 'W' and 'E'
toLocation	Yes	String (10)	The location number of the to location.
deliveryDate	No	date	The earliest date the transfer can be delivered. When Inventory Planning is part of your implementation, this is required for all transfer types, except RAC, EG, and SIM transfers. If included in the message, this must be today or a future date.
expectedDcDate	No	date	The date that the transfer is expected to be shipped at the warehouse. If included in the message, this must be today or a future date.

Table 5-382 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
routingCode	No	String (1)	If the freight code is Expedite (E), then this must have a value. Otherwise, it must be null. The descriptions for these three options are held in the Codes table under code TRRC and can be configured as needed for your business. Examples are Federal Express (1), UPS (2) and mail (3).
transferType	No	String (6)	A code indicating the type of transfer (e. g. store requisition, book transfer, etc.). The following types of transfers can be created in this service: Administrative (AD), Inventory Planning Generated (AIP), Book (BT), Confirmation (CF), Externally Generated (EG), Intercompany (IC), Manual Requisition (MR), Reallocation (RAC), Return to Vendor ,(RV) and Store Inventory Operations Generated (SIM).
status	No	String (1)	A code indicating the status of the transfer. Valid values are 'T' and 'A'.
createdBy	No	String (30)	The userid of the user who created the transfer.
comments	No	String (2000)	Comments associated with the transfer.

Table 5-382 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contextType	No	String (6)	This field holds the reason code related to which a transfer is made. Valid values are found in the code type CNTX. Examples are Promotion (PROM), Customer Transfer (WED), Store Requisition (STORE) and Repairing (REPAIR).
contextValue	No	String (25)	This field holds the value relating to the context type like Promotion number.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Table 5-383 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "items": [
    {
      "transferNo": 12300056,
      "fromLocationType": "S",
      "fromLocation": "1521",
      "toLocationType": "S",
```

```

    "toLocation": "1511",
    "deliveryDate": "2001-12-31",
    "expectedDcDate": "2001-12-31",
    "routingCode": null,
    "transferType": "AD",
    "status": "A",
    "createdBy": "RMS01APP",
    "comments": "transfer to loc 1511",
    "contextType": "PROM",
    "contextValue": "2255",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ]
  }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Work Order Status Upload Service

The following services are included in this functional area:

- [Update Work Orders Status](#)

Update Work Orders Status

Functional Area

Inventory - Transfers and Allocations

Business Overview

For transfers with finishing, Merchandising subscribes to Work Order Status messages sent from internal finishers indicating that the work order activities are complete. This message is used for internal finishers located in the same physical warehouse as the final destination for the transfer, as there is no physical shipment of goods. Other finishing scenarios exist in which the finisher is not a virtual warehouse that shares a physical warehouse with the transfer's final receiving location. In these instances, Work Order Status messages are not necessary, and Merchandising will disregard Work Order Status messages sent in these scenarios.

Work order status messages contain the items for which the activities have been completed along with the quantity that was completed. All items on transfers that pass through an internal finisher must have at least one work order activity associated with them. When Work Order status messages are received for a particular item/quantity, it is assumed that all activities on the work order associated with the item/quantity have been completed. If work order activities involve item transformation or repacking, the work order status messages are always created in terms of the resultant item or pack.

On processing the work order status update, a book transfer is executed between the internal finisher (which is held as a virtual warehouse) and the final receiving location (also a virtual warehouse). If the internal finisher belongs to the sending location's transfer entity, intercompany out and intercompany in transactions are recorded. Quantities on hand, reserved quantities, and weighted average costs are adjusted to accurately reflect the status of the stock.

It is possible to receive multiple Work Order Status messages for a particular item/transfer. Work order completion of partial quantities addresses the following scenarios:

1. Work order activities could not be performed for the entire quantity of a particular item at one time.
2. A given quantity of the particular item was damaged while work order activities were performed.

Work Order Example

Assume that a quantity of 20 of item 100 (White XL T-shirt) are sent to an internal finisher at the receiving physical warehouse, where they will be dyed black, thereby transforming them into item 101 (Black XL T-shirt). If all finishing activities were successfully completed in this example, Merchandising could expect to receive a Work Order Status message containing item 101 with a quantity of 20.

Work Order Update Process

While consuming the Work Order Status message, Merchandising validates that the finisher and the transfer's final receiving location are in the same physical warehouse. If not, processing is halted. If the message contains an item, work order complete processing will be called for that item. Otherwise, said processing will be called for all items on the transfer. If the

entire transfer is processed, the child transfer (that is, the second leg) will be set to Shipped status. Note that work orders are always associated with the second leg of multi-leg transfers. Whether processing is performed at the item or transfer level, transfer closing logic will be used to determine if the entire multi-leg transfer can be closed.

Service Type

PUT

ReST URL

MerchIntegrations/services/inventory/transfer/workOrdersStatus/update

Input Payload Details

Table 5-384 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of work orders.

Table 5-385 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferNo	Yes	Number (12)	This field contains the identifier of the transfer during which the work order activities will be performed. For 2-legged transfers, this is the transfer number of the 1st leg transfer.
parentTransferNo	No	Number (12)	The higher level transfer that this transfer is part of.
item	No	String (25)	The item on which the work order has been completed. If an item transformation occurred, this should be the resultant item. This should be populated if completedQty is provided.
location	Yes	Number (10)	This is the finisher location.
completeDate	Yes	date	This is the workorder completion date. If this is not provided, the value will be defaulted to the current date.
completedQuantity	No	Number (12,4)	The quantity of the item on which work has been completed. This should be provided if item is provided.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
```

```

        "transferNo": 100000135001,
        "parentTransferNo": 100000135000,
        "item": "101300002",
        "location": 15000,
        "completeDate": "2001-12-31",
        "completedQuantity": 1
    }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Work Order Download Service

The following services are included in this functional area:

- [Purchase Order Work Order Publish Services](#)
- [Transfer Work Order Publish Services](#)

Purchase Order Work Order Publish Services

Endpoints

MerchIntegrations/services/inventory/woin

MerchIntegrations/services/inventory/woin/{workOrderId}

Functional Area

Inventory – Work Orders

Business Overview

This service publishes work-order-in data to downstream consuming systems. The payload is keyed by the work order ID and publishes one header row together with its detail lines. Functionally, the JSON view builds the payload directly from WO_HEAD and WO_DETAIL.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: WO_HEAD, WO_DETAIL

Additional business tables: none

JSON cache table: MERCHAPI_EXT_WOIN

JSON generation view: V_MERCHAPI_EXT_WOIN_JSON

Builds the payload directly from WO_HEAD and WO_DETAIL.

Functionally, the payload is keyed by WO_HEAD.WO_ID and publishes one header row with an array of detail lines.

Refresh loads rows from V_MERCHAPI_EXT_WOIN_JSON into the cache, rebuild compares the current payload against the live cache by MD5 and republishes changed rows, and rows that no longer exist in WO_HEAD are marked deleted.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_WOIN_JSON. No additional filter criteria were identified beyond the cache-build logic.

Delete handling can mark a cache row deleted when the source WO_HEAD row no longer exists.

ICL (Integration Change Log) table: ICL_RMS_WOIN (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on WO_HEAD, WO_DETAIL.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_WOIN

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_WO_IN_ADHOC_PROCESS ->
MERCHAPI_EXT_WO_IN_ADHOC_JOB

Webhook configuration api name: inventory/woin

Special handling

Additional GET query parameters: orderNo.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/woin

MerchIntegrations/services/inventory/woin/{workOrderId}

Input Parameters for MerchIntegrations/services/inventory/woin

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
orderNo	No	Number	OrderNo
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-386 Input Parameter for "MerchIntegrations/services/inventory/woin/{workOrderId}"

Parameter Name	Required	Data Type	Description
workOrderId	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/inventory/woin/{workOrderId}

Parameter Name	Required	Data Type	Description
workOrderId	Yes	Number	This is the unique identifier for work in process associated with an order.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output “ WO In Head

Table 5-387 PageResultsItemsMerchApiWorkOrderIn - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-388 MerchApiWorkOrderIn - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Work Order In details.
workOrderId	No	Number(10,0)	This field contains the unique identifier for work in process associated with an order or transfer.
orderNo	No	Number(12,0)	This field contains the order number associated with the work order.

Table 5-388 (Cont.) MerchApiWorkOrderIn - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This column holds the record creation date.
details	No	Collection of Object	This List holds attributes for details of Work Order In record.
cacheTimestamp	No	dateTime	This field specifies date and time when the Work Order In record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies date and time when the record was first created for web service publication.

Table 5-389 WorkOrderInDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
warehouse	No	Number(10,0)	This field holds the physical warehouse where the work order will be done.
item	No	String (25)	This field holds the item on which the work order will be done.
locationType	No	String (1)	This field holds the location type of the final destination.
location	No	Number(10,0)	This field holds the final destination for the item on the order. If the item is pre-distributed this location will be the store or warehouse on the allocation. If not, it will be the same warehouse as above.
sequenceNo	No	Number(4,0)	This field holds the sequence number that is needed to ensure each record is unique, since the same work-in-progress code can be listed more than once on the same work order.

Table 5-389 (Cont.) WorkOrderInDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
activityCode	No	String (6)	This field holds the WIP code for the work to be done on the item. Valid values are in the codes table under the WWIP code type.
createDateTime	No	dateTime	This column holds the record creation date.

Table 5-390 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "workOrderId": 1,
      "orderNo": 545005,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "details": [
        {
          "warehouse": 1,
          "item": "101300002",
          "locationType": "S",
          "location": 15000,
          "sequenceNo": 1,
          "activityCode": "12122",
          "createDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
}
```

```

"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output – WO In Detail

Name	Data Type	Description
warehouse	Number	This field holds physical warehouse where the work order will be done.
item	String	This field holds item on which the work order will be done.
locationType	String	This field holds location type of the final destination.
location	Number	This field holds final destination for the item on the order. If the item is pre-distributed this location will be the store or warehouse on the allocation. If not, it will be the same warehouse as above.
sequenceNo	Number	This field holds sequence number that is needed to ensure each record is unique, since the same work-in-progress code can be listed more than once on the same work order.
activityCode	String	This field holds WIP code for the work to be done on the item. Valid values are in the codes table under the WWIP code type.
createDateTime	String	This column holds the record creation date.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```

"items": [
  {
    "action": "INSERT",
    "workOrderId": 10007,
    "orderNo": 10001,
    "createDateTime": "2021-05-13T19:47:33.000Z",
    "details": [
      {
        "warehouse": 5,
        "item": "101550001",
        "locationType": "W",
        "location": 5,
        "sequenceNo": 1,
        "activityCode": "QC",
        "createDateTime": "2021-05-13T19:47:49.000Z"
      }
    ],
    "cacheTimestamp": "2021-06-21T05:42:18.934Z"
  },
  {
    "action": "UPDATE",
    "workOrderId": 15001,
    "orderNo": 655005,
    "createDateTime": "2021-08-05T03:17:27.000Z",
    "details": [
      {
        "warehouse": 2,
        "item": "100950020",
        "locationType": "W",
        "location": 2,
        "sequenceNo": 2,
        "activityCode": "QC",
        "createDateTime": "2021-08-05T03:20:54.000Z"
      },
      {
        "warehouse": 2,
        "item": "100950062",
        "locationType": "W",
        "location": 2,
        "sequenceNo": 4,
        "activityCode": "KIT",
        "createDateTime": "2021-08-05T12:08:03.000Z"
      },
      {
        "warehouse": 2,
        "item": "100950062",
        "locationType": "W",
        "location": 2,
        "sequenceNo": 3,
        "activityCode": "TICKET",
        "createDateTime": "2021-08-05T03:32:18.000Z"
      }
    ],
  }
]

```

```

        "warehouse": 2,
        "item": "100950062",
        "locationType": "W",
        "location": 2,
        "sequenceNo": 2,
        "activityCode ": "QC",
        "createDateTime": "2021-08-05T03:20:54.000Z"
    },
    {
        "warehouse": 2,
        "item": "100950020",
        "locationType": "W",
        "location": 2,
        "sequenceNo": 4,
        "activityCode ": "KIT",
        "createDateTime": "2021-08-05T12:08:03.000Z"
    },
    {
        "warehouse": 2,
        "item": "100950020",
        "locationType": "W",
        "location": 2,
        "sequenceNo": 3,
        "activityCode ": "TICKET",
        "createDateTime": "2021-08-05T03:32:18.000Z"
    }
],
"cacheTimestamp": "2021-08-05T12:08:56.163Z"
},
],
"hasMore": false,
"limit": 10000,
"count": 2,
"links": [
    {
        "href": "https://<hostname>:<port>/rgbu-rex-rgbu-dev1-mfcs19-1/
MerchIntegrations/services/inventory/woin?
limit=10000&since=1970-01-01T00:00:00.001Z&before=2022-03-26T02:28:02.971902+0
0:00",
        "rel": "self"
    }
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_WOIN	Yes	No	Yes	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_EXT_WOIN	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_WOIN_JSON	Yes	No	No	No

Transfer Work Order Publish Services

Endpoints

MerchIntegrations/services/inventory/woout

MerchIntegrations/services/inventory/woout/{workOrderId}

Functional Area

Inventory – Work Orders

Business Overview

This service publishes work-order-out data to downstream consuming systems. It publishes transfer work orders together with the related transfer and work-order detail built for outbound fulfillment. Functionally, the view only publishes transfer work orders whose source transfer location resolves to either a warehouse or an external finisher partner, and refresh and rebuild only load transfers whose TSFHEAD.APPROVAL_DATE is not null.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: TSF_WO_DETAIL, TSF_WO_HEAD, TSFDETAIL, TSF_XFORM_DETAIL, TSF_PACKING_DETAIL, TSFHEAD

Additional business tables: TSF_XFORM, TSF_PACKING, WH, PARTNER

JSON cache table: MERCHAPI_EXT_WOOUT

JSON generation view: V_MERCHAPI_EXT_WOOUT_JSON

The payload is assembled from transfer work-order data in TSF_WO_DETAIL, TSF_WO_HEAD, TSFDETAIL, TSF_XFORM_DETAIL, TSF_XFORM, TSF_PACKING_DETAIL, TSF_PACKING, TSFHEAD, WH, and PARTNER.

The view only publishes transfer work orders whose source transfer location resolves to either a warehouse or an external finisher partner, and the refresh/rebuild logic only loads transfers whose TSFHEAD.APPROVAL_DATE is not null.

Insert and update events republish the JSON payload as a merge, while direct delete events only mark the cache row deleted when a live cache row still exists.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_WOOUT_JSON and apply refresh and rebuild only load rows where TSFHEAD.APPROVAL_DATE is not null.

ICL (Integration Change Log) table: ICL_RMS_WOOUT (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on TSF_WO_DETAIL, TSF_WO_HEAD, TSFDETAIL, TSF_XFORM_DETAIL, TSF_PACKING_DETAIL, TSFHEAD.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_WOOUT

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_WO_OUT_ADHOC_PROCESS ->
MERCHAPI_EXT_WO_OUT_ADHOC_JOB

Webhook configuration api name: inventory/woout

Special handling

Additional GET query parameters: tsfNo.

Service Type

GET

ReST URL

MerchIntegrations/services/inventory/woout

MerchIntegrations/services/inventory/woout/{workOrderId}

Input Parameters for MerchIntegrations/services/inventory/woout

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
tsfNo	No	Number	TransferNo
offsetkey	No	String	Offset key
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-391 Input Parameter for "MerchIntegrations/services/inventory/woout/{workOrderId}"

Parameter Name	Required	Data Type	Description
workOrderId	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/inventory/woout/{workOrderId}

Parameter Name	Required	Data Type	Description
workOrderId	Yes	Number	This is the unique identifier for work in process associated with an order or transfer.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output "WO Out Head"

Table 5-392 PageResultsItemsMerchApiWorkOrderOut - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object

Table 5-392 (Cont.) PageResultsItemsMerchApiWorkOrderOut - Object. See list of elements for detail

Element Name	Required	Data Type	Description
links	No	Collection of Object	List of Self and/or Next URL

Table 5-393 MerchApiWorkOrderOut - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Work Order Out details.
workOrderId	No	Number(15,0)	This field contains the unique identifier for work in process associated with an order or transfer.
distroNo	No	Number(12,0)	This field contains the first leg transfer number of the 2-legged transfer.
parentDistroNo	No	Number(12,0)	This field contains the second leg transfer number of the 2-legged transfer.
finisherLocationId	No	Number(10,0)	This field contains the destination location for the transfer.
inventoryType	No	String (6)	This field contains the code to indicate whether the inventory on the transfer is available.
createDateTime	No	dateTime	This column holds the record creation date.
updateDateTime	No	dateTime	This column holds the record updated date.
activityDetails	No	Collection of Object	This List holds attributes for details of Work Order Out record.
transformationDetails	No	Collection of Object	This List holds attributes for details of Work Order transformation details record.
packingDetails	No	Collection of Object	This List holds attributes for details of Work Order packing details record.

Table 5-393 (Cont.) MerchApiWorkOrderOut - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cacheTimestamp	No	dateTime	This field specifies date and time when the Work Order Out record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies date and time when the record was first created for web service publication.

Table 5-394 WorkOrderOutDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	This field contains the item associated with the finishing activity.
activityId	No	Number(10,0)	This field contains a code number indicating the activity that will be performed on the item.
activityUnitCost	No	Number(20,4)	This field contains the per unit cost of the finishing activity.
comments	No	String (2000)	This field contains any comments regarding this activity.
invStatus	No	Number(2,0)	This field contains the inventory status of the transfer detail.
createDateTime	No	dateTime	This column holds the record creation date.
updateDateTime	No	dateTime	This column holds the record updated date.

Table 5-395 TransformationDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromItem	No	String (25)	This field contains the original item on the transfer.
toItem	No	String (25)	This field contains the item which results from the finishing activity.
createDateTime	No	dateTime	This column holds the record creation date.

Table 5-395 (Cont.) TransformationDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This column holds the record updated date.

Table 5-396 PackingDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	This field contains the item resulting from the packing process.
packingSetNo	No	Number(10,0)	This field contains packing set number from the packing process.
recordType	No	String (1)	This field determines if the item in the ITEM field is being used to create a pack (F - From) an item or a result of the pack creation process (R - Result).
quantity	No	Number(12,4)	This field contains the quantity of the item used to create the pack or as a result of the packing process.
createDateTime	No	dateTime	This column holds the record creation date.
updateDateTime	No	dateTime	This column holds the record updated date.

Table 5-397 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
```

```

{
  "action": "INSERT",
  "workOrderId": 1,
  "distroNo": 100000135001,
  "parentDistroNo": 100000135000,
  "finisherLocationId": 1,
  "inventoryType": "A",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z",
  "activityDetails": [
    {
      "item": "101300002",
      "activityId": 12122,
      "activityUnitCost": 10,
      "comments": null,
      "invStatus": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ],
  "transformationDetails": [
    {
      "fromItem": null,
      "toItem": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ],
  "packingDetails": [
    {
      "item": null,
      "packingSetNo": 1,
      "recordType": null,
      "quantity": 2,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ],
  "cacheTimestamp": "2001-12-31T23:59:59.000Z",
  "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
{
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}

```

Output – WO Activity Details

Name	Data Type	Description
item	String	Unique identifier for the item.
activityId	Number	This field contains a code number indicating the activity that will be performed on the item.
activityUnitCost	Number	This field contains the per unit cost of the finishing activity.
comments	String	This field contains any comments regarding this activity
invStatus	String	This field contains the inventory status of the transfer detail.
createDateTime	String	This column holds the record creation date.
updateDateTime	String	This column holds the record updated date.

Output – WO Transformation Details

Name	Data Type	Description
fromItem	String	This field contains the original item on the transfer.
toItem	String	This field contains the item which results from the finishing activity.
createDateTime	String	This column holds the record creation date.
updateDateTime	String	This column holds the record updated date.

Output – WO Packing Details

Name	Data Type	Description
item	String	This field contains the item resulting from the packing process
packingSetNo	Number	This field contains packing set number from the packing process.
recordType	String	This field determines if the item in the ITEM field is being used to create a pack (F - From) an item or a result of the pack creation process (R - Result).
quantity	Number	his field contains the quantity of the item used to create the pack or as a result of the packing process.
createDateTime	String	This column holds the record creation date.
updateDateTime	String	This column holds the record updated date.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.

Elements	Description
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```

"items": [
  {
    "workOrderId": 1,
    "distroNo": 100000135001,
    "parentDistroNo": 100000135000,
    "finisherLocationId": 15000,
    "inventoryType": "A",
    "createDateTime": "2020-10-31T16:00:00.000Z",
    "updateDatetime": "2021-06-13T09:44:00.000Z",
    "activityDetails": [{
      "item": "101200001",
      "activityId": 2,
      "activityUnitCost": 0.3,
      "comments": "2 pc 1 kit",
      "invStatus": null,
      "createDatetime": "2021-06-11T03:32:34.000Z",
      "updateDateTime": "2021-06-11T03:32:34.000Z"
    }, {
      "item": "101300002",
      "activityId": 3,
      "activityUnitCost": 0.3,
      "comments": null,
      "invStatus": null,
      "createDatetime": "2021-06-11T03:57:54.000Z",
      "updateDateTime": "2021-06-11T03:57:54.000Z"
    }, {
      "item": "101800040",
      "activityId": 2,
      "activityUnitCost": 0.3,
      "comments": null,
      "invStatus": null,
      "createDatetime": "2021-06-11T03:35:58.000Z",
      "updateDateTime": "2021-06-11T03:35:58.000Z"
    }
  ],
  "transformationDetails": [{
    "fromItem": "OMNITESTITEM",
    "toItem": "100000323",
    "createDateTime": "2021-06-11T03:13:17.000Z",
    "updateDateTime": "2021-06-11T03:13:17.000Z"
  }
],

```

```

    "packingDetails": [{
      "item": "101800040",
      "packingSetNo": 1,
      "recordType": "F",
      "quantity": 10,
      "createDateTime": "2021-06-11T03:10:13.000Z",
      "updateDateTime": "2021-06-11T03:10:13.000Z"
    }, {
      "item": "101800040",
      "packingSetNo": 1,
      "recordType": "R",
      "quantity": 10,
      "createDateTime": "2021-06-11T03:10:13.000Z",
      "updateDateTime": "2021-06-11T03:10:13.000Z"
    }]
  }, {
    "hasMore": false,
    "limit": 10000,
    "count": 1,
    "links": [
      {
        "href": "https://rex.retail.us-phoenix-1.ocs.oc-test.com/rgbu-rex-rgbu-dev1-mfcs19-1/MerchIntegrations/services/inventory/woin?limit=10000&since=1970-01-01T00:00:00.001Z&before=2022-03-26T02:28:02.971902+00:00",
        "rel": "self"
      }
    ]
  }
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_WOOUT	Yes	No	Yes	Yes
MERCHAPI_EXT_WOOUT	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_WOOUT_JSON	Yes	No	No	No

Items

The following services are included in this functional area:

- [Item Definition](#)
 - [Item AI Services](#)
 - * [Item Attribute Extraction Services](#)
 - * [Get Item Attributes Base](#)

- * [Get Item Attributes Customized](#)
 - * [Update Item Attributes Customized](#)
- [Item Download Service](#)
 - * [Get Item Details](#)
 - * [Get Item Details for All or Provided Items](#)
 - * [Get Item Details for Provided Item](#)
 - * [Get Item Location Details](#)
 - * [Get Item VAT Details](#)
 - * [Get Item VAT Details by Item Number](#)
- [Item ELC Upload Service](#)
 - * [Create Item Conditional Tariff Treatments](#)
 - * [Create Item Expenses](#)
 - * [Create Item HTS](#)
 - * [Create Item HTS Assessments](#)
 - * [Delete Item Conditional Tariff Treatments](#)
 - * [Delete Item Expenses](#)
 - * [Delete Item HTS](#)
 - * [Delete Item HTS Assessments](#)
 - * [Update Item Expenses](#)
 - * [Update Item HTS](#)
 - * [Update Item HTS Assessments](#)
- [Item Location Upload Service](#)
 - * [Create Item Locations](#)
 - * [Manage Item Location Status Update Requests](#)
 - * [Update Item Locations](#)
- [Item Number Reservation Upload Service](#)
 - * [Generate and Reserve Item Numbers](#)
- [Item Placeholder Upload Service](#)
 - * [Create Placeholder Items](#)
- [Item Sourcing Upload Service](#)
 - * [Create Item Supplier Countries](#)
 - * [Create Item Supplier Countries of Manufacture](#)
 - * [Create Item Supplier Country Dimensions](#)
 - * [Create Item Supplier Country Locations](#)
 - * [Create Item Supplier Translations](#)
 - * [Create Item Suppliers](#)
 - * [Delete Item Supplier Countries](#)
 - * [Delete Item Supplier Countries of Manufacture](#)

- * [Delete Item Supplier Country Dimensions](#)
- * [Delete Item Supplier Country Locations](#)
- * [Delete Item Supplier Translations](#)
- * [Delete Item Suppliers](#)
- * [Update Item Supplier Countries](#)
- * [Update Item Supplier Countries of Manufacture](#)
- * [Update Item Supplier Country Dimensions](#)
- * [Update Item Supplier Country Locations](#)
- * [Update Item Supplier Translations](#)
- * [Update Item Suppliers](#)
- [Item Taxes Upload Service](#)
 - * [Create Item Taxes](#)
 - * [Delete Item Taxes](#)
- [Item Up Charge Download Service](#)
 - * [Get Item Up Charges for Future Date](#)
- [Item Upload Service](#)
 - * [Create Item Description Translations](#)
 - * [Create Item Image Translations](#)
 - * [Create Item Images](#)
 - * [Create Item Seasons](#)
 - * [Create Item Tickets](#)
 - * [Create Item UDA Free Form Translations](#)
 - * [Create Item UDAs](#)
 - * [Create Item Up Charges](#)
 - * [Create Items](#)
 - * [Create Required Documents](#)
 - * [Delete Item Description Translations](#)
 - * [Delete Item Image Translations](#)
 - * [Delete Item Images](#)
 - * [Delete Item Seasons](#)
 - * [Delete Item Tickets](#)
 - * [Delete Item UDA Free Form Translations](#)
 - * [Delete Item UDAs](#)
 - * [Delete Item Up Charges](#)
 - * [Delete Items](#)
 - * [Delete Required Documents](#)
 - * [Update Item Description Translations](#)
 - * [Update Item Image Translations](#)

- * [Update Item Images](#)
- * [Update Item Tickets](#)
- * [Update Item UDA Free Form Translations](#)
- * [Update Item UDAs](#)
- * [Update Item Up Charge Details](#)
- * [Update Items](#)
- [Related Items Upload Service](#)
 - * [Create Related Items](#)
 - * [Create Related Item Translations](#)
 - * [Delete Related Items](#)
 - * [Delete Related Item Translations](#)
 - * [Update Related Items](#)
 - * [Update Related Item Translations](#)
- [Item Foundation](#)
 - [Brand Download Service](#)
 - * [Get Brands](#)
 - [Brand Upload Service](#)
 - * [Manage Brands](#)
 - [Differentiator Download Service](#)
 - * [Get Differentiator Details](#)
 - * [Get Differentiator Details for Provided Differentiator](#)
 - * [Get Differentiator Details for Single Differentiator](#)
 - [Differentiator Group Download Service](#)
 - * [Get Differentiator Groups](#)
 - * [Get Differentiators for Provided Group](#)
 - [Differentiator Group Upload Service](#)
 - * [Create Differentiator Group Details](#)
 - * [Create Differentiator Groups](#)
 - * [Delete Differentiator Group Details](#)
 - * [Delete Differentiator Groups](#)
 - * [Update Differentiator Group Details](#)
 - * [Update Differentiator Groups](#)
 - [Differentiator Type Download Service](#)
 - * [Get Differentiator Type Details for Provided Type](#)
 - * [Get Differentiator Types](#)
 - [Differentiator Upload Service](#)
 - * [Create Differentiators](#)
 - * [Delete Differentiators](#)

- * [Update Differentiators](#)
- [Seasons and Phases Download Service](#)
- * [Get Seasons and Phases Service](#)
- [User Defined Attribute \(UDA\) Download Service](#)
- * [REST Endpoint for UDA details](#)
- * [REST Endpoint to Fetch UDA by UDA ID](#)

Item AI Services

The following services are included in this functional area:

- [Item Attribute Extraction Services](#)

Item Attribute Extraction Services

The following services are included in this functional area:

- [Get Item Attributes Base](#)
- [Get Item Attributes Customized](#)
- [Update Item Attributes Customized](#)

Get Item Attributes Base

Endpoints

/RmsReSTServices/services/private/ai/item/attributes/base

Functional Area

Items Attributes Extraction

Business Overview

Fetches the default (out-of-the-box) set of item attributes that ODA can extract. Use this to see all pre-configured attributes.

Service Type

GET

ReST URL

/RmsReSTServices/services/private/ai/item/attributes/base

Output

Name	Data Type	Description
<CategoryName>	Array of String	Each key in the JSON response is a product category name (e.g., APPAREL, ELECTRONICS). The value is an array of extractable attribute names (Strings) for that category.

Name	Data Type	Description
attribute	String	Each element within the array is the name of an attribute extractable for the given category (e.g., color, brand, price)

Elements in JSON Output

Elements	Description
CATEGORY_NAME	Product category name (e.g., "APPAREL", "ELECTRONICS")
Attribute	Extractable attribute for the category (e.g., "color", "price")

JSON Schema Overview

```
{
  "CATEGORY_NAME": [
    "attribute1",
    "attribute2",
    "attribute3"
    // ...more attributes
  ],
  // ...more categories
}
```

Example JSON output

```
{
  "APPAREL": [
    "productType",
    "material",
    "color",
    "size",
    "fit",
    "pattern",
    "sleeveLength",
    "neckStyle",
    "gender",
    "occasion",
    "price"
  ],
  "BAGS": [
    "productType",
    "material",
    "color",
    "capacity",
    "closureType",
    "style",
    "price"
  ],
}
```

```
"COSMETICS": [  
  "productType",  
  "brand",  
  "shade",  
  "form",  
  "skinType",  
  "scent",  
  "size",  
  "price"  
],  
"ELECTRONICS": [  
  "productType",  
  "brand",  
  "model",  
  "color",  
  "memory",  
  "features",  
  "price"  
],  
"FOOTWEAR": [  
  "productType",  
  "material",  
  "color",  
  "size",  
  "closureType",  
  "soleMaterial",  
  "heelType",  
  "usage",  
  "price"  
],  
"FRESH PRODUCE": [  
  "productType",  
  "brand",  
  "weight",  
  "units",  
  "packaging",  
  "expiryDate",  
  "price"  
],  
"FURNITURE": [  
  "productType",  
  "material",  
  "color",  
  "dimensions",  
  "style",  
  "roomType",  
  "price"  
],  
"GROCERY": [  
  "productType",  
  "brand",  
  "flavor",  
  "weight",  
  "packaging",  
  "expiryDate",  
  "price"
```



```
],  
"JEWELRY": [  
  "productType",  
  "material",  
  "stoneType",  
  "cutShape",  
  "color",  
  "sizeDimensions",  
  "price"  
],  
"PET SUPPLIES": [  
  "productType",  
  "brand",  
  "packaging",  
  "weight",  
  "flavor",  
  "petCategory",  
  "expiryDate",  
  "price"  
],  
"PHARMA": [  
  "productType",  
  "brand",  
  "flavor",  
  "packaging",  
  "expiryDate",  
  "price"  
]  
}
```

Get Item Attributes Customized

Endpoints

/RmsReSTServices/services/private/ai/item/attributes

Functional Area

Items Attributes Extraction

Business Overview

Retrieves only those item attributes that have been customized for extraction. If no customizations have been made, the response will be empty. Use this endpoint to check which product categories and attributes have customized extraction setups in your system.

Service Type

GET

ReST URL

/RmsReSTServices/services/private/ai/item/attributes

Output

Name	Data Type	Description
<CategoryName>	Array of String	Each key in the JSON response is a product category name (e.g., APPAREL, ELECTRONICS). The value is an array of extractable attribute names (Strings) for that category.
attribute	String	Each element within the array is the name of an attribute extractable for the given category (e.g., color, brand, price)

Elements in JSON Output

Elements	Description
CATEGORY_NAME	Product category name (e.g., "APPAREL", "ELECTRONICS")
Attribute	Extractable attribute for the category (e.g., "color", "price")

JSON Schema Overview

```
{
  "CATEGORY_NAME": [
    "attribute1",
    "attribute2",
    "attribute3"
    // ...more attributes
  ],
  // ...more categories
}
```

Example JSON output

```
{
  "APPAREL": [
    "productType",
    "material",
    "color",
    "size",
    "fit",
    "pattern",
    "sleeveLength",
    "neckStyle",
    "gender",
    "occasion",
    "price"
  ],
  "BAGS": [
    "productType",
    "material",
```

```
"color",
"capacity",
"closureType",
"style",
"price"
],
"COSMETICS": [
"productType",
"brand",
"shade",
"form",
"skinType",
"scent",
"size",
"price"
],
"ELECTRONICS": [
"productType",
"brand",
"model",
"color",
"memory",
"features",
"price"
],
"FOOTWEAR": [
"productType",
"material",
"color",
"size",
"closureType",
"soleMaterial",
"heelType",
"usage",
"price"
],
"FRESH PRODUCE": [
"productType",
"brand",
"weight",
"units",
"packaging",
"expiryDate",
"price"
],
"FURNITURE": [
"productType",
"material",
"color",
"dimensions",
"style",
"roomType",
"price"
],
"GROCERY": [
"productType",
```

```
"brand",
"flavor",
"weight",
"packaging",
"expiryDate",
"price"
],
"JEWELRY": [
"productType",
"material",
"stoneType",
"cutShape",
"color",
"sizeDimensions",
"price"
],
"PET SUPPLIES": [
"productType",
"brand",
"packaging",
"weight",
"flavor",
"petCategory",
"expiryDate",
"price"
],
"PHARMA": [
"productType",
"brand",
"flavor",
"packaging",
"expiryDate",
"price"
]
}
```

Update Item Attributes Customized

Endpoints

/RmsReSTServices/services/private/ai/item/attributes

Functional Area

Items Attributes Extraction

Business Overview

Allows you to add or remove attributes to be extracted through the Oracle Digital Assistant (ODA).

Service Type

POST

ReST URL

/RmsReSTServices/services/private/ai/item/attributes

Input

Name	Data Type	Description
<CategoryName>	Array of String	Each key in the JSON response is a product category name (e.g., APPAREL, ELECTRONICS). The value is an array of extractable attribute names (Strings) for that category.
attribute	String	Each element within the array is the name of an attribute extractable for the given category (e.g., color, brand, price)

Output

Name	Data Type	Description
status	String	Indicates the result of the data override action. The value will confirm successful completion if updates were processed without error.

Elements in JSON Input

Elements	Description
CATEGORY_NAME	Product category name (e.g., "APPAREL", "ELECTRONICS")
Attribute	Extractable attribute for the category (e.g., "color", "price")

Input JSON Schema Overview

```
{
  "CATEGORY_NAME": [
    "attribute1",
    "attribute2",
    "attribute3"
    // ...more attributes
  ],
  // ...more categories
}
```

Example JSON input

```
{
  "APPAREL": [
```

```
"productType",
"material",
"color",
"size",
"fit",
"pattern",
"sleeveLength",
"neckStyle",
"gender",
"occasion",
"price"
],
"BAGS": [
"productType",
"material",
"color",
"capacity",
"closureType",
"style",
"price"
],
"COSMETICS": [
"productType",
"brand",
"shade",
"form",
"skinType",
"scent",
"size",
"price"
],
"ELECTRONICS": [
"productType",
"brand",
"model",
"color",
"memory",
"features",
"price"
],
"FOOTWEAR": [
"productType",
"material",
"color",
"size",
"closureType",
"soleMaterial",
"heelType",
"usage",
"price"
],
"FRESH PRODUCE": [
"productType",
"brand",
"weight",
"units",
```

```
"packaging",
"expiryDate",
"price"
],
"FURNITURE": [
"productType",
"material",
"color",
"dimensions",
"style",
"roomType",
"price"
],
"GROCERY": [
"productType",
"brand",
"flavor",
"weight",
"packaging",
"expiryDate",
"price"
],
"JEWELRY": [
"productType",
"material",
"stoneType",
"cutShape",
"color",
"sizeDimensions",
"price"
],
"PET SUPPLIES": [
"productType",
"brand",
"packaging",
"weight",
"flavor",
"petCategory",
"expiryDate",
"price"
],
"PHARMA": [
"productType",
"brand",
"flavor",
"packaging",
"expiryDate",
"price"
]
}
```

Elements in JSON Output

Elements	Description
status	Indicates the result of the customization operation. Returns a message confirming success or providing details in case of failure.

Output JSON Schema Overview

```
{  
  "status": "string"  
}
```

Example JSON Output

```
{  
  "status": "Data overrides done successfully"  
}
```

Item Definition

The following services are included in this functional area:

- [Item Download Service](#)
 - [Get Item Attributes Base](#)
 - [Get Item Attributes Customized](#)
 - [Get Item Details](#)
 - [Get Item Details for All or Provided Items](#)
 - [Get Item Details for Provided Item](#)
 - [Get Item Location Details](#)
 - [Get Item VAT Details](#)
 - [Get Item VAT Details by Item Number](#)
- [Item ELC Upload Service](#)
 - [Create Item Conditional Tariff Treatments](#)
 - [Create Item Expenses](#)
 - [Create Item HTS](#)
 - [Create Item HTS Assessments](#)
 - [Delete Item Conditional Tariff Treatments](#)
 - [Delete Item Expenses](#)
 - [Delete Item HTS](#)
 - [Delete Item HTS Assessments](#)
 - [Update Item Expenses](#)
 - [Update Item HTS](#)
 - [Update Item HTS Assessments](#)

- [Item Location Upload Service](#)
 - [Create Item Locations](#)
 - [Manage Item Location Status Update Requests](#)
 - [Update Item Locations](#)
- [Item Number Reservation Upload Service](#)
 - [Generate and Reserve Item Numbers](#)
- [Item Placeholder Upload Service](#)
 - [Create Placeholder Items](#)
- [Item Sourcing Upload Service](#)
 - [Create Item Supplier Countries](#)
 - [Create Item Supplier Countries of Manufacture](#)
 - [Create Item Supplier Country Dimensions](#)
 - [Create Item Supplier Country Locations](#)
 - [Create Item Supplier Translations](#)
 - [Create Item Suppliers](#)
 - [Delete Item Supplier Countries](#)
 - [Delete Item Supplier Countries of Manufacture](#)
 - [Delete Item Supplier Country Dimensions](#)
 - [Delete Item Supplier Country Locations](#)
 - [Delete Item Supplier Translations](#)
 - [Delete Item Suppliers](#)
 - [Update Item Supplier Countries](#)
 - [Update Item Supplier Countries of Manufacture](#)
 - [Update Item Supplier Country Dimensions](#)
 - [Update Item Supplier Country Locations](#)
 - [Update Item Supplier Translations](#)
 - [Update Item Suppliers](#)
- [Item Taxes Upload Service](#)
 - [Create Item Taxes](#)
 - [Delete Item Taxes](#)
- [Item Up Charge Download Service](#)
 - [Get Item Up Charges for Future Date](#)
- [Item Upload Service](#)
 - [Create Item Description Translations](#)
 - [Create Item Image Translations](#)
 - [Create Item Images](#)
 - [Create Item Seasons](#)
 - [Create Item Tickets](#)

- [Create Item UDA Free Form Translations](#)
- [Create Item UDAs](#)
- [Create Item Up Charges](#)
- [Create Items](#)
- [Create Required Documents](#)
- [Delete Item Description Translations](#)
- [Delete Item Image Translations](#)
- [Delete Item Images](#)
- [Delete Item Seasons](#)
- [Delete Item Tickets](#)
- [Delete Item UDA Free Form Translations](#)
- [Delete Item UDAs](#)
- [Delete Item Up Charges](#)
- [Delete Items](#)
- [Delete Required Documents](#)
- [Update Item Description Translations](#)
- [Update Item Image Translations](#)
- [Update Item Images](#)
- [Update Item Tickets](#)
- [Update Item UDA Free Form Translations](#)
- [Update Item UDAs](#)
- [Update Item Up Charge Details](#)
- [Update Items](#)
- [Related Items Upload Service](#)
 - [Create Related Items](#)
 - [Create Related Item Translations](#)
 - [Delete Related Items](#)
 - [Delete Related Item Translations](#)
 - [Update Related Items](#)
 - [Update Related Item Translations](#)

Item Download Service

The following services are included in this functional area:

- [Get Item Attributes Base](#)
- [Get Item Attributes Customized](#)
- [Get Item Details](#)
- [Get Item Details for All or Provided Items](#)
- [Get Item Details for Provided Item](#)

- [Get Item Location Details](#)
- [Get Item VAT Details](#)
- [Get Item VAT Details by Item Number](#)

Get Item Details

This section describes the Item Detail service.

Business Overview

Item Detail service allows user to retrieve Item information for a selected item.

Service Type

Get

ReST URL

Item/itemDetail?item={itemNumber}

Input Parameters

Parameter Name	Required	Description
Item	Yes	Item number

Output

RestItemRecRDO

Parameter Name	Data Type
itemGrandparent	String
itemParent	String
item	String
itemDesc	String
shortDesc	String
packInd	String
status	String
itemLevel	BigDecimal
tranLevel	BigDecimal
dept	BigDecimal
classAttribute	BigDecimal
subclass	BigDecimal
diff1	String
diff2	String
diff3	String
diff4	String
primaryRefItemInd	String
originalRetail	BigDecimal
sellableInd	String
orderableInd	String

Parameter Name	Data Type
inventoryInd	String
packitemBreakout	List<RestPackitemBreakoutRecRDO>
itemSupplier	List<RestItemSupplierRecRDO>
itemSupplierCountry	List<RestItemSupplierCountryRecRDO>
vatItem	List<RestVatItemRecRDO>
longDesc	String

RestPackitemBreakoutRecRDO

Parameter Name	Data Type
item	String
seqNo	BigDecimal
packItemQty	BigDecimal

RestItemSupplierRecRDO

Parameter Name	Data Type
supplier	BigDecimal
vpn	String
primarySupplInd	String
directShipInd	String

RestItemSupplierCountryRecRDO

Parameter Name	Data Type
originCountryId	String
primaryCountryInd	String
unitCost	BigDecimal
suppPackSize	BigDecimal
innerPackSize	BigDecimal
leadTime	BigDecimal
pickupLeadTime	BigDecimal

RestVatItemRecRDO

Parameter Name	Data Type
vatRegion	BigDecimal
vatType	String
vatCode	String
vatRate	BigDecimal
activeDate	Timestamp

JSON Structure

```

{
  "itemGrandparent": null,
  "itemParent": null,
  "item": null,
  "itemDesc": null,
  "shortDesc": null,
  "packInd": null,
  "status": null,
  "itemLevel": null,
  "tranLevel": null,
  "dept": null,
  "classAttribute": null,
  "subclass": null,
  "diff1": null,
  "diff2": null,
  "diff3": null,
  "diff4": null,
  "primaryRefItemInd": null,
  "originalRetail": null,
  "sellableInd": null,
  "orderableInd": null,
  "inventoryInd": null,
  "packitemBreakout": [],
  "itemSupplier": [
    {
      "primarySuppInd": null,
      "itemSupplierCountry": [
        {
          "unitCost": null,
          "leadTime": null,
          "suppPackSize": null,
          "originCountryId": null,
          "primaryCountryInd": null,
          "pickupLeadTime": null,
          "innerPackSize": null,
          "links": [],
          "hyperMediaContent": {
            "linkRDO": []
          }
        }
      ],
      "supplier": null,
      "vpn": null,
      "directShipInd": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "vatItem": [
    {
      "vatRegion": null,

```

```
        "activeDate": null,
        "vatType": null,
        "vatCode": null,
        "vatRate": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    },
    "longDesc": null,
    "links": [],
    "hyperMediaContent": {
        "linkRDO": []
    }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_ITEM_MASTER	Yes	No	No	No
PACKITEM_BREAKOUT	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
VAT_ITEM	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Item Details for All or Provided Items

Endpoints

MerchIntegrations/services/foundation/item

MerchIntegrations/services/foundation/item/{itemId}

Functional Area

Items - Item Definition

Business Overview

This service publishes item master data for transaction-level items and higher merchandise hierarchy nodes to downstream consuming systems. The payload combines core item attributes with translations, supplier and sourcing details, UDA values, images, tickets, related items, compliance and import data, and custom flex attributes. End-to-end publication is limited to items that qualify through the refresh and delta flow: refresh seeds items in approved or deleted status, item-master inserts are not queued until the item is approved, and item-master updates in worksheet or submitted status are suppressed. Below-transaction-level reference or UPC items are not published as standalone records; their changes refresh the parent item's referenceItem node.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_MASTER, ITEM_MASTER_TL, ITEM_SUPPLIER, ITEM_SUPP_UOM, ITEM_SUPP_COUNTRY, ITEM_SUPP_COUNTRY_DIM, ITEM_SUPP_MANU_COUNTRY, ITEM_TICKET, UDA_ITEM_DATE, UDA_ITEM_LOV, UDA_ITEM_FF, ITEM_IMAGE, ITEM_SEASONS, ITEM HTS, ITEM_EXP_HEAD, ITEM_EXP_DETAIL, RELATED_ITEM_HEAD, RELATED_ITEM_DETAIL, REQ_DOC, ITEM_IMPORT_ATTR

Additional business tables: ITEM_MASTER_CFA_EXT, ITEM_SUPPLIER_CFA_EXT, ITEM_SUPP_COUNTRY_CFA_EXT, DEPS, GROUPS, DIVISION, CLASS, SUBCLASS, BRAND, DIFF_IDS, DIFF_GROUP_HEAD, SYSTEM_OPTIONS, SYSTEM_CONFIG_OPTIONS, CODE_DETAIL

JSON cache table: MERCHAPI_EXT_ITEM

JSON generation view: V_MERCHAPI_EXT_ITEM_JSON

Is keyed by ITEM_MASTER rows with ITEM_LEVEL <= TRAN_LEVEL, so the published entity is the transaction-level item or a higher hierarchy node.

The payload combines core item attributes with translations, supplier and sourcing data, UDAs, images, tickets, related items, HTS and expense data, import attributes, custom flex attributes, and supporting hierarchy and lookup values.

Below-transaction-level reference or UPC items are not published as standalone payloads; their data is rolled into the parent item's referenceItem node.

Initial Seeding and post enablement rebuild/repair

Initial seeding selects ITEM_MASTER rows where STATUS IN ('A', 'D') and ITEM_LEVEL <= TRAN_LEVEL, then builds the published JSON through V_MERCHAPI_EXT_ITEM_JSON.

Rebuild marks cache rows deleted when the item no longer exists in ITEM_MASTER.

ICL (Integration Change Log) table: ICL_RMS_ITEM_MASTER, ICL_RMS_ITEM_SUPPLIER, ICL_RMS_ITEM_SUPP_UOM, ICL_RMS_ITEM_SUPP_CTRY, ICL_RMS_ITEM_SUPP_CTRY_DIM, ICL_RMS_ITEM_SUPP_MANU_CTRY, ICL_RMS_ITEM_TICKET, ICL_RMS_UDA_ITEM_DATE, ICL_RMS_UDA_ITEM_LOV, ICL_RMS_UDA_ITEM_FF, ICL_RMS_ITEM_IMAGE, ICL_RMS_ITEM_SEASONS, ICL_RMS_ITEM HTS, ICL_RMS_ITEM_EXP, ICL_RMS_RELATED_ITEM, ICL_RMS_REQ_DOC, ICL_RMS_ITEM_IMPORT_ATTR (ICL consumer name = MERCHAPI)

Item-master inserts are staged only when the new item is approved, approval transitions are staged as insert-style publishes, and item-master updates in worksheet or submitted status are suppressed.

ITEM_MASTER_TL changes are queued only for items with ITEM_LEVEL <= TRAN_LEVEL, and below-transaction-level item changes are rolled up to ITEM_PARENT so the parent payload is refreshed.

Item Approval Handling

Approval of an item also results in publish of foundation/omnichannel/item/image, foundation/omnichannel/item/relateditem, foundation/item/location, foundation/item/vat, and foundation/taxLocationItem. The item approval flow writes the existing ITEM_IMAGE, RELATED_ITEM_HEAD/DETAIL, ITEM_LOC, and VAT_ITEM source rows for that item into those APIs' ICL tables so dependent approved-item data is queued together with the item.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ITEM

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job: MERCHAPI_DELTA_EXT_ITEM_ADHOC_PROCESS -> MERCHAPI_EXT_ITEM_ADHOC_JOB

Webhook configuration api name: foundation/item

Special handling

Additional GET query parameters: itemLevel, tranLevel, deptId, classId, subclassId, status, itemType, inventoryInd, supplier, referenceItem.

The filters itemLevel, tranLevel, deptId, classId, subclassId, status, and itemType are not backed by dedicated cache-table indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. The available configuration for this service should therefore be reviewed and validated against the customer production data profile, including expected volume, message size, and access pattern, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for initial seeding, while delta processing remains a single threaded process. Initial-seeding thread count is controlled through API_REFRESH_OVERRIDE_NO_OF_THREADS, and initial-seeding commit sizing is controlled through API_REFRESH_OVERRIDE_COMMIT_MAX_COUNTER.

For initial seeding, thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the item message from ITEM_MASTER together with supplier, UDA, image, ticket, related-item, HTS, expense, import, and hierarchy data. Common starting points for large-volume seeding are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment. The delivered refresh override is currently 4 threads.

Refresh commit sizing should be reviewed together with the thread setting. The delivered refresh override commit setting is currently 10000 and that should be used if the thread value is low. If running with higher thread value, the typical value can range from about 100 to 1,000 rows.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. Even with refresh-override threading, larger parallel insert volumes can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes.

This can improve data-load throughput, but the overall refresh plan should also account for the additional time needed for index rebuild after the seeding step finishes.

`GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a need to change it.

REST fetch considerations

Item payloads can be comparatively large, so REST page size should be selected based on average message size so that the average response remains below about 10 MB per call. Common page-limit values for this service should be 500 or 1,000, and very small limits, especially below 100, should generally be avoided.

For full downloads, do not provide the `since` query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned `before` timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

`MerchIntegrations/services/foundation/item`

`MerchIntegrations/services/foundation/item/{itemId}`

Input Parameters

Parameter Name	Required	Data Type	Description
<code>since</code>	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
<code>before</code>	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
<code>itemLevel</code>	No	String	Item Level - 1, 2, or 3

Parameter Name	Required	Data Type	Description
tranLevel	No	String	Transaction Level- 1, 2, or 3
deptId	No	String (4)	Department Id
classId	No	String (4)	Class Id
subclassId	No	String (4)	Sub Class Id
status	No	String	Status
itemType	No	String	Item Type
inventoryInd	No	String	Inventoried
supplier	No	String	Supplier
referenceItem	No	String	Reference Item
offsetkey	No	String	Offset key (Item)
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-398 Input Parameter for "MerchIntegrations/services/foundation/item/{itemId}"

Parameter Name	Required	Data Type	Description
itemId	Yes	String	Item Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-399 PageResultsItemsMerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-400 MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Item details
longtDescription	No	String	
item	No	String (25)	This field specifies Unique alphanumeric value that identifies the item.
itemNumberType	No	String (6)	This field holds the type of item number. Valid values are defined by code type UPCT.
status	No	String (1)	This field specifies status of the item.
itemLevel	No	Number(1,0)	This field specifies Number indicating which of the three levels the item resides. The item level determines if the item stands alone or if it is part of a family of related items. The item level also determines how the item may be used throughout the system.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
tranLevel	No	Number(1,0)	This field specifies Number indicating which of the three levels transactions occur for the item's group. The transaction level is the level at which the item's inventory is tracked in the system. The transaction level item will be counted, transferred, shipped, etc. The transaction level may be at the current item or up to 2 levels above or below the current item. Only one level of the hierarchy of an item family may contain transaction level items.
itemDescription	No	String (250)	This field specifies long description of the item. This description is used through out the system to help online users identify the item. For items that have parents, this description will default to the parents description plus any differentiators. For items without parents this description will default to null.
shortDescription	No	String (120)	This field specifies shortened description of the item. This description is the default for downloading to the POS. For items that have parents, this description will default to the parent's short description. For items without parents, this description will default to null.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemDescriptionUpper case	No	String (250)	This field specifies all upper case description of the item (same as upper(item_desc)). This field is not displayed in the on-line item maintenance dialog, but is used in item list of value search processing through out the system.
itemDescriptionSecondary	No	String (250)	This field specifies secondary descriptions of the item. This field can only be populated when system_options.secondary_desc_ind = Y.
longDescription	No	String (2000)	This field specifies long description of the item. This is a more descriptive writeup about the item targeted largely for platforms like Ecommerce platforms, Marketplaces, Order Management Systems etc.
itemParent	No	String (25)	This field specifies alphanumeric value that uniquely identifies the item/group at the level above the item.
itemGrandparent	No	String (25)	This field specifies alphanumeric value that uniquely identifies the item/group at the level above the item.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sellableInd	No	String (1)	This field indicates if pack item may be sold as a unit. If it is 'Y' then the pack will have its own unique unit retail. If it is 'N' then the pack's unit retail is the sum of each individual item's total retail within the pack. This field will only be available if the item is a pack item. Value values are: Y = Yes, this pack may be sold as a unit; N = No, this pack may not be sold as a unit.
orderableInd	No	String (1)	This field indicates if pack item is orderable. If it is 'Y' then the suppliers of the pack must supply all components in the pack. If it is 'N' then the components may have different suppliers. This field will only be available if the item is a pack item. Valid values are: Y = Yes, this pack may be ordered, N = No, this pack may not be ordered.
inventoryInd	No	String (1)	This field specifies inventory Indicator.
packInd	No	String (1)	This field indicates if the item is a pack. A pack item is a collection of items that may be either ordered or sold as a unit. Packs require details (i.e. component items and qtys, etc.) that other items do not. This field is required by the database.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
simplePackInd	No	String (1)	This field indicates if pack item is a simple pack or not. This field will only be available if the item is a pack item. A simple pack is an item whose components are all the same item (i.e. a six pack of cola, etc). Valid values are: Y = Yes, this item is a simple pack; N = No if this is not a simple pack.
containsInnerInd	No	String (1)	This field indicates if pack item contains inner packs. Vendor packs will never contain inner packs and this field will be defaulted to 'N'. This field will only be available if the item is a pack item. Valid values are Y = Yes, this pack contains inner packs; N = No, this pack does not contain inner packs.
dept	No	Number(4,0)	This field specifies number identifying the department to which the item is attached. The item's department will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the depts table.
deptName	No	String (120)	This field contains the name of the department.
class	No	Number	
className	No	String (120)	This field contains the name of the class.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uniqueClassId	No	Number(10,0)	This field uniquely identify the class which the item belongs to. Class is not unique ID the merchandise hierarchy. The combination of Dept/Class is unique, but requires use of a composite key. The composite key is generally used in user interfaces. The unique ID can be used in back end processing or in systems that can not have a composite key in the merchandise hierarchy.
subclass	No	Number(4,0)	This field specifies number identifying the subclass to which the item is attached. The item's subclass will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the subclass table.
subclassName	No	String (120)	This field contains the name of the subclass.
uniqueSubclassId	No	Number(10,0)	This field uniquely identify the subclass which the item belongs to. Subclass is not unique ID the merchandise hierarchy. The combination of Dept/Class/Subclass is unique, but requires use of a composite key. The composite key is generally used in user interfaces. The unique ID can be used in back end processing or in systems that can not have a composite key in the merchandise hierarchy.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitRetail	No	Number(20,4)	This field holds the unit retail in the standard unit of measure for the item/zone combination. This field is stored in the local currency.
variableUpcFormatId	No	String (1)	This field will hold the format ID that corresponds to the item's variable UPC. This value is only used for items with variable UPCs.
variableWeightUpcPrefix	No	Number(2,0)	This field holds the prefix for variable weight UPCs. The prefix determines the format of the eventual UPC and will be used to decode variable weight UPCs that are uploaded from the POS. It is the clients responsibility to download this value to their POS system.
diff1	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff1Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
diff2	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
diff2Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff3	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
diff3Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
diff4	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff4Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
costZoneGroupId	No	Number(4,0)	This field contains cost zone group associated with the item. This field is only required when elc_ind (landed cost indicator) is set to 'Y' on the system_options table.
standardUom	No	String (4)	This field contains unit of measure in which stock of the item is tracked at a corporate level.
uomConversionFactor	No	Number(20,10)	This field contains Conversion factor between an 'Each' and the standard_uom when the standard_uom is not in the quantity class (e.g. if standard_uom = lb and 1 lb = 10 eaches, this factor will be 10). This factor will be used to convert sales and stock data when an item is retailed in eaches but does not have eaches as its standard unit of measure.
packageSize	No	Number(12,4)	This field holds the size of the product printed on any packaging (i.e. 24 ounces). This field is used for reporting purposes, as well as by Retek Price Mangement to determine same sized and different sized items.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
packageUom	No	String (4)	This field holds the unit of measure associated with the package size. This field is used for reporting purposes and by Retek Price Management to determine same sized and different sized items.
merchandiseInd	No	String (1)	This field indicates if the item is a merchandise item (Y, N).
storeOrderMultiple	No	String (1)	This field indicates Merchandise shipped from the warehouses to the stores must be specified in this unit type. Valid values are: C = Cases; I = Inner; E = Eaches.
forecastInd	No	String (1)	This field indicates if this item will be interfaced to an external forecasting system (Y, N).
manufacturerRecomm endedRetail	No	Number(20,4)	This field contains manufacturer's recommended retail price for the item. Used for informational purposes only. This field is stored in the primary currency.
manufacturerRetailCu rrencyCode	No	String (3)	The field indicates the currency code of the Manufacturer's retail price.
originalRetail	No	Number(20,4)	The field indicated the original retail price of the item per unit. This field is stored in the primary currency.
originalRetailCurrency Code	No	String (3)	The field indicates the currency code of the original retail price.
retailLabelType	No	String (6)	This field indicates any special label type associated with an item.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
retailLabelTypeDescription	No	String (250)	This field holds the description of the retail label type.
retailLabelValue	No	Number(20,4)	This field represents the value associated with the retail label type.
handlingTemperature	No	String (6)	This field holds the temperature information associated with the item.
handlingTemperatureDescription	No	String (250)	This field holds the description of the handling temperature.
handlingSensitivity	No	String (6)	This field holds the sensitivity information associated with the item.
handlingSensitivityDescription	No	String (250)	This field holds the description of the handling sensitivity.
catchWeightInd	No	String (1)	This field indicates whether the item should be weighed when it arrives at a location.
catchWeightType	No	String (1)	The field will hold catch weight type for a simple pack catch weight item. The value is based on the component items order_type and sale_type: 2 - order_type = Variable Weight, sale_type = Loose Weight 4 - order_type = Variable Weight, sale_type = Variable Weight Each.
catchWeightUom	No	String (4)	The field indicates the UOM for Catchweight Items.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderType	No	String (6)	The field determines how catch weight items are ordered. Valid values are: F - fixed weight V - Variable weight Valid values are held on the CODE_DETAIL table with a code type = ORDT.
saleType	No	String (6)	The field indicates the method of how catch weight items are sold in store locations. Valid values are: V - variable weight each L - Loose weight. Valid values are held on the CODE_DETAIL table with a code type = STPE.
wasteType	No	String (6)	The field indicates identifies the wastage type as either sales or spoilage wastage. Sales wastage occurs during processes that make an item saleable (i.e. fat is trimmed off at customer request). Spoilage wastage occurs during the products shelf life (i.e. evaporation causes the product to weigh less after a period of time). Valid values are: SP - spoilage; and SL - Sales.
wasteTypeDescription	No	String (250)	The field contains the Waste Type description corresponding to the Waste_Type value in item_master.
averageWastePercentage	No	Number(12,4)	The field indicates average percent of wastage for the item over its shelf life. Used in inflating the retail price for wastage items.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
defaultWastePercentage	No	Number(12,4)	The field indicates default daily wastage percent for spoilage type wastage items. This value will default to all item locations and represents the average amount of wastage that occurs on a daily basis.
constantDimensionInd	No	String (1)	The field indicates that the dimensions of the product are always the same, regardless of the supplier. If this field is Y, the dimensions for all suppliers will be defaulted to the primary supplier/ primary country dimensions. Editing of dimensions for the item will only be allowed for the primary supplier/ primary country.
packType	No	String (1)	The field Indicates if pack item is a vendor pack or a buyer pack. A vendor pack is a pack that the vendor or supplier recognizes and sells to the retailer. If the pack item is a vendor pack, communication with the supplier will use the vendor pack number. A buyer pack is a pack that a buyer created for internal ease of use. If the pack item is a buyer pack communication with the supplier will explode the pack out to its component items. This field will only be used if the item is a pack item. If the pack item is not orderable this field must be null. Valid values are: V = Vendor; B = Buyer.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderAsType	No	String (1)	This field indicates if pack item is receivable at the component level or at the pack level (for a buyer pack only). This field is required if pack item is an orderable buyer pack. This field must be NULL if the pack is sellable only or a vendor pack. This field will only be available if the item is a pack item. Valid Values are: E = Eaches (component level) P = Pack (buyer pack only)..
comments	No	String (2000)	The field holds any comments associated with the item.
itemServiceLevel	No	String (6)	The field Holds a value that restricts the type of shipment methods that RCOM can select for an item.
giftWrapInd	No	String (1)	This field will contain a value of 'Y' if the item is eligible to be gift wrapped.
shipAloneInd	No	String (1)	This field will contain a value of Y if the item should be shipped to the customer in a separate package.
brandName	No	String (30)	This field contains the brand associated to an item.
brandDescription	No	String (120)	This field contains the description of the brand associated with the item.
perishableInd	No	String (1)	The field is not used by RMS.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemTransformInd	No	String (1)	The field show that an item is associated with an item transformation. The item will be either the sellable item or orderable item in the transformation process.
depositItemType	No	String (6)	The field contains deposit item component type. A NULL value in this field indicates that this item is not part of a deposit item relationship. The possible values are - E - Contents A - Container Z - Crate T - Returned Item (Empty bottle) P - Complex pack (with deposit items) The Returned item is flagged only to enable these items to be mapped to a separate GL account if required.
containerItem	No	String (25)	The field holds the container item number for a contents item. This field is only populated and required if the DEPOSIT_ITEM_TYPE = E.
depositInPricePerUom	No	String (6)	This field indicates if the deposit amount is included in the price per UOM calculation for a contents item ticket. This value is only required if the DEPOSIT_ITEM_TYPE = E. Valid values are I - Include deposit amount E - Exclude deposit amount.
storePackInventoryInd	No	String (1)	This field indicates whether downstream systems should track pack inventory at the pack level.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sohInquiryAtPackInd	No	String (1)	The field indicates to the Store Inventory System if a notional simple pack item's inventory should be displayed in packs.
purchaseType	No	String (1)	The field indicates if an item is a normal, consignment or concession item - 'N' for normal, 'S' for consignment, 'C' for concession.
productClassification	No	String (6)	The field contains item combinability codes (with code type 'PCLA') which provide a way to define which items can be combined (packed or boxed) together.
productClassificationDescription	No	String (250)	This field will hold the description of the product classification.
itemAggregateInd	No	String (1)	This field holds indicator to aggregate inventory and sales for the item. Currently, this indicator is used by allocation and MFP to aggregate inventory for items. For staple items this indicator should be N.
diff1AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (e.g. Style/Color or Style/Size).
diff2AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (e.g. Style/Color or Style/Size).

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff3AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/ Diff level (e.g. Style/ Color or Style/Size.
diff4AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/ Diff level (e.g. Style/ Color or Style/Size.
diff1Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff1Description	No	String (120)	This field will hold the description of differentiator.
diff2Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff2Description	No	String (120)	This field will hold the description of differentiator.
diff3Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff3Description	No	String (120)	This field will hold the description of differentiator.
diff4Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff4Description	No	String (120)	This field will hold the description of differentiator.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryImageUrl	No	String (375)	This field will hold the primary item image URL. It is only included for the ItemCre and ItemFulRep message types. Updates to the primary image for an item are published in the Item Image node.
transferAllocUpChargeSourceTypes	No	String (6)	This field impacts how up charges default to transfers and allocations when an item is added. When set to 'D', department-level up charges are applied if item-level up charges are not defined for the item/ from-location/to-location combination. When set to 'I', only item-level up charges are applied; department-level up charges are not inherited.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item.
itemTranslation	No	Collection of Object	This List holds translation attributes for the item.
itemSupplier	No	Collection of Object	This List holds supplier's attributes for the item.
itemUda	No	Object	This record holds uda attributes for the item.
itemImage	No	Collection of Object	This List holds image attributes for the item.
itemSeason	No	Collection of Object	This List holds season attributes for the item.
referenceItem	No	Collection of Object	This List holds referenc item's attributes for the item.

Table 5-400 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemBOM	No	Collection of Object	This List holds pack attributes for the item.
itemTicket	No	Collection of Object	This List holds ticket attributes for the item.
relatedItem	No	Collection of Object	This List holds related item's attributes for the item.
hts	No	Collection of Object	This List holds hts attributes for the item.
expense	No	Collection of Object	This List holds expense attributes for the item.
requiredDocuments	No	Collection of Object	This List holds the required documents of the item.
itemImportAttributes	No	Collection of Object	This List holds the import specific attributes associated with a given item.
cacheTimestamp	No	dateTime	This field specifies date and time when item record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies date and time when the record was first created for web service publication.

Table 5-401 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).

Table 5-401 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-402 ItemTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	This field contains the language in which the translated text is maintained
languageName	No	String (120)	This field contains description of language code in which the translated text is maintained
isoCode	No	String (6)	This field will hold the ISO code associated with the given language.
itemDescription	No	String (250)	This field will hold the ISO code associated with the given language.
itemDescriptionSecondary	No	String (250)	This field holds long description of the item. This description is used through out the system to help online users identify the item. For items that have parents, this description will default to the parents description plus any differentiators. For items without parents, this description will default to null

Table 5-402 (Cont.) ItemTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shortDescription	No	String (120)	This field holds shortened description of the item. This description is the default for downloading to the POS. For items that have parents, this description will default to the parents short description. For items without parents, this description will default to null
longDescription	No	String (2000)	This field specifies long description of the item. This is a more descriptive writeup about the item targeted largely for platforms like Ecommerce platforms, Marketplaces, Order Management Systems etc.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-403 ItemSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Number(10,0)	This field specifies unique identifier for the supplier.
primarySupplierInd	No	String (1)	This field indicates whether this supplier is the primary supplier for the item. An item can have one and only one primary supplier.
vpn	No	String (30)	This field contains the Vendor Product Number for the Item/ Supplier.
supplierLabel	No	String (15)	This field contains the supplier label. It will only be populated if item is a parent item.

Table 5-403 (Cont.) ItemSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
consignmentRate	No	Number(12,4)	This field contains the consignment rate for this item for the supplier.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff4	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
palletName	No	String (6)	This field specifies code referencing the name used to refer to the pallet.
caseName	No	String (6)	This field specifies code referencing the name used to refer to the case.
innerName	No	String (6)	This field specifies code referencing the name used to refer to the inner.
supplierDiscontinueDate	No	date	This field specifies the date when the supplier discontinues an item.

Table 5-403 (Cont.) ItemSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
directShipInd	No	String (1)	This field indicates that any item associated with this supplier is eligible for a direct shipment from the supplier to the customer.
primaryCaseSize	No	String (6)	This field indicates the primary case size for the item supplier when an orderable item is configured for informal case types.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item's supplier record.
itemSupplierCountry	No	Collection of Object	This List holds custom flex attributes for the item supplier's country record.
itemSupplierCountryOfManufacture	No	Collection of Object	This List holds custom flex attributes for the item supplier's manufacture record.
itemSupplierUom	No	Collection of Object	This List holds custom flex attributes for the item's supplier UOM record.
translation	No	Collection of Object	List of translations for Item Supplier

Table 5-404 ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	No	String (3)	This field specifies country where the item was manufactured or significantly altered.
primarySupplierInd	No	String (1)	This field indicates whether this supplier is the primary supplier for the item. Each item can have one and only one primary supplier.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryCountryInd	No	String (1)	This field indicates whether this country is the primary country for the item/supplier. Each item/supplier combination can have one and only one primary country.
unitCost	No	Number(20,4)	This field contains the current corporate unit cost for the item from the supplier /origin country. This field is stored in the Supplier's currency..
leadTime	No	Number(4,0)	This field contains the number of days that will elapse between the date an order is written and the delivery to the store or warehouse from the supplier.
pickupLeadTime	No	Number(4,0)	This field Contains the time that will be taken to get the item from the supplier to the initial receiving location.
supplierPackSize	No	Number(12,4)	This field contains the quantity that orders must be placed in multiples of for the supplier for the item.
innerPackSize	No	Number(12,4)	This field contains the break pack size for this item from the supplier.
roundLevel	No	String (6)	This field is used to determine how order quantities will be rounded to case, layer and pallet.
minimumOrderQuantity	No	Number(12,4)	This field contains the minimum quantity that can be ordered at once from the supplier for the item.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
maximumOrderQuantity	No	Number(12,4)	This field contains the maximum quantity that can be ordered at once from the supplier for the item.
packingMethod	No	String (6)	This field indicates whether the packing method for the item in the container is flat or hanging.
defaultUop	No	String (6)	This field contains the default unit of purchase for the item/supplier/country. Valid values include 'C' (Case) and 'P' (Pallet)
ti	No	Number(12,4)	This field contains number of shipping units (cases) that make up one tier of the pallet. Multiply TI x HI to get total number of units (cases) in a Pallet.
hi	No	Number(12,4)	This field contains number of tiers that make up a complete pallet. Multiply TI x HI to get total number of units (cases) in a Pallet.
costUom	No	String (4)	This field contains cost UOM is held to allow costs to be managed in a separate UOM to the standard UOM.
toleranceType	No	String (6)	The unit of tolerance for catch weight simple packs. Valid values are: 'A' - actual 'P' - percent.
maximumTolerance	No	Number(12,4)	The maximum tolerance value for the catch weight simple pack.
minimumTolerance	No	Number(12,4)	The minimum tolerance value for the catch weight simple pack.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
roundToInnerPercentage	No	Number(12,4)	This field will hold the Inner Rounding Threshold value. During rounding, this value is used to determine whether to round partial Inner quantities up or down. If the Inner-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
roundToCasePercentage	No	Number(12,4)	This field will hold the Case Rounding Threshold value. During rounding, this value is used to determine whether to round partial Case quantities up or down. If the Case-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
roundToLayerPercentage	No	Number(12,4)	This field will hold the Layer Rounding Threshold value. During rounding, this value is used to determine whether to round partial Layer quantities up or down. If the Layer-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
roundToPalletPercentage	No	Number(12,4)	This field will hold the Pallet Rounding Threshold value. During rounding, this value is used to determine whether to round partial Pallet quantities up or down. If the Pallet -fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
supplierHierarchyType1	No	String (6)	This field identifies partner type of supplier hierarchy level 1. This field will always have the partner type S1 which indicates manufacturer.
supplierHierarchyLevel1	No	String (10)	This field contains highest level of supplier hierarchy which is there to indicate a partner, such as a manufacturer, in the supply chain that gives rebates to the retailer.
supplierHierarchyType2	No	String (6)	This field identifies partner type of supplier hierarchy level 2. This field will always have the partner type S2 which indicates distributor.
supplierHierarchyLevel2	No	String (10)	This field contains second highest level of supplier hierarchy which is there to indicate a partner, such as a distributor, in the supply chain that gives rebates to the retailer..
supplierHierarchyType3	No	String (6)	This field identifies partner type of supplier hierarchy level 3. This field will always have the partner type S3 which indicates wholesaler.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy which is there to indicate a partner, such as a wholesaler, in the supply chain that gives rebates to the retailer.
negotiatedItemCost	No	Number(20,4)	This field will hold the supplier negotiated item cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations negotiated item cost will be stored in this field.
extendedBaseCost	No	Number(20,4)	This field will hold the extended base cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations extended base cost will be stored in this field. Extended base cost is the cost inclusive of all the taxes that affect the WAC. In case of GTAX, Extended Base Cost = Base Cost + Non-recoverable taxes. In case of VAT, Extended Base Cost = Base Cost.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inclusiveCost	No	Number(20,4)	This field will hold the inclusive cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations inclusive cost will be stored in this field. This cost will have both the recoverable and non-recoverable taxes included. In case of GTAX, Inclusive Cost = Base Cost + Non-recoverable taxes + Recoverable Taxes. In case of VAT, Inclusive Cost = Base Cost + VAT.
baseCost	No	Number(20,4)	This field will hold the tax exclusive cost of the item.
purchaseType	No	Number(1,0)	This field contains a code to indicate whether the item is normal merchandise (i.e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This field indicates if the cost for the consignment/ concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number(12,4)	This field contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 5-404 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pricingControl	No	String (1)	This field contains the pricing control of the item which will determine the item price is controlled by Retailer or Supplier. Valid value are R - Managed by Retailer, S - Managed by Supplier, NULL - For Owned Items
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item supplier's country record.
itemSupplierCountryDimension	No	Collection of Object	This List holds dimension attributes for the item supplier's country record.

Table 5-405 ItemSupplierCountryDimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	No	String (6)	This field Specific object whose dimensions are specified in this record (e.g. case, pallet, each). Valid values for this field are in the code type DIMO on the codes tables.
dimensionObjectDescription	No	String (250)	This field contains description of the dimension object.
presentationMethod	No	String (6)	This field describes the packaging (if any) being taken into consideration in the specified dimensions. Valid values for this field are in the code type PCKT on the codes tables.

Table 5-405 (Cont.) ItemSupplierCountryDimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
presentationMethodDescription	No	String (250)	This field contains description of the presentation method.
length	No	Number(12,4)	This field contains length of dim_object measured in units specified in terms of the LWH Unit of Measure.
width	No	Number(12,4)	This field contains width of dim_object measured in units specified in terms of the LWH Unit of Measure.
height	No	Number(12,4)	This field contains height of dim_object measured in units specified in terms of the LWH Unit of Measure.
lwhUom	No	String (4)	This field contains measurement for length, width, and height.
weight	No	Number(12,4)	This field contains weight of dim_object measured in units specified in terms of the Weight Unit of Measure.
netWeight	No	Number(12,4)	This field contains net weight of the dim_object (weight without packaging) measured in units specified in terms of the Weight Unit of Measure.
weightUom	No	String (4)	This field contains unit of measurement for Weight and Net Weight.
liquidVolume	No	Number(12,4)	This field contains unit of measurement for volume.

Table 5-405 (Cont.) ItemSupplierCountryDimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
liquidVolumeUom	No	String (4)	This field contains unit of measurement for liquid volume. Liquid volumes are only convertible to other liquid volumes.
statisticalCase	No	Number(12,4)	This field contains statistical value of the dim_object's dimensions to be used for loading purposes.
tareWeight	No	Number(12,4)	This field contains amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dim_object.
tareType	No	String (6)	This field indicates whether the tare weight for the dim_object is wet or dry.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-406 ItemSupplierCountryOfManufacture - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	No	String (3)	This field holds country ID of manufacturer.
primaryInd	No	String (1)	This field indicates primary country of manufacturer indicator.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-407 ItemSupplierUom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uom	No	String (4)	This field holds unit of measure that the item is being measured in.
value	No	Number(20,4)	This field holds value of measure.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-408 ItemSupplierTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	The code identifying the language.
languageName	No	String (120)	The display name of the language.
isoCode	No	String (6)	The ISO code of the language.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description.
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description.
supplierLabel	No	String (15)	The translated supplier label for the item.
createDateTime	No	dateTime	The creation date/time of the translation record.
updateDateTime	No	dateTime	The last update date/time of the translation record.

Table 5-409 ItemUda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaLov	No	Collection of Object	This List attributes for the item's uda list of values.
udaFreeform	No	Collection of Object	This List holds attributes for the item's uda free form .
udaDate	No	Collection of Object	This List holds attributes for the item's uda date.

Table 5-410 UdaLov - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	No	Number(5,0)	This field contains a number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains description of user-defined attribute.
udaValue	No	Number(5,0)	This field contains the unique identified of the Attribute Group being added or delete to the item list.
udaValueDescription	No	String (250)	This field contains the value of the Attribute Group.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-411 UdaFreeForm - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	No	Number(5,0)	This field contains a number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains description of user-defined attribute.
udaText	No	String (250)	This field contains the text value of the user-defined attribute for the item.

Table 5-411 (Cont.) UdaFreeForm - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
translation	No	Collection of Object	List of translations for UDA free form text.

Table 5-412 UdaFreeFormTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	The code identifying the language.
languageName	No	String (120)	The display name of the language.
isoCode	No	String (6)	The ISO code of the language.
udaTextDesc	No	String (250)	The translated UDA text.
createDateTime	No	dateTime	The creation date/time of the translation record.
updateDateTime	No	dateTime	The last update date/time of the translation record.

Table 5-413 UdaDate - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	No	Number(5,0)	This field contains a number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains description of user-defined attribute.
udaDate	No	date	This field contains the effective date for the UDA id.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-414 ItemImage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	No	String (120)	This field specifies name of the image.
imageAddress	No	String (255)	This field specifies path where the image file is stored.
imageType	No	String (6)	This field contains the type of the image of the item. Valid values are defined as member of IITD code type.
primaryImageInd	No	String (1)	This field specifies display sequence order of images associated to the item per priority.
displayPriority	No	Number(4,0)	This field specifies display sequence order of images associated to the item per priority.
imageDescription	No	String (40)	This field contains the description associated with the image of the item.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
translation	No	Collection of Object	List of translations for Item Image

Table 5-415 ItemImageTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	The code identifying the language.
languageName	No	String (120)	The display name of the language.
isoCode	No	String (6)	The ISO code of the language.
imageDescription	No	String (40)	This field contains the translated description associated with the image of the item.
createDateTime	No	dateTime	The creation date/time of the translation record.

Table 5-415 (Cont.) ItemImageTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	The last update date/time of the translation record.

Table 5-416 ItemSeason - Object. See list of elements for detail

Element Name	Required	Data Type	Description
seasonId	No	Number(3,0)	The season identifier.
phaseId	No	Number(3,0)	The phase identifier.
sequenceNo	No	Number(4,0)	This field contains a sequence number that combined with the item number will make each record unique.
diffId	No	String (10)	This field will hold a value for group/differentiator combination records only.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-417 ReferenceItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
referenceItem	No	String (25)	This field holds Unique identifier for the item.
primaryInd	No	String (1)	This field Indicates if the sub-transaction level item is designated as the primary sub-transaction level item. For transaction level items and above, the value in this field will be 'N'.
formatId	No	String (1)	This field will hold the format id that corresponds to the item's variable UPC. This value is only used for items with variable UPCs.

Table 5-417 (Cont.) Referenceltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
prefix	No	Number(2,0)	This field holds the prefix for variable weight UPCs. The prefix determines the format of the eventual UPC and will be used to decode variable weight UPCs that are uploaded from the POS.
itemNoType	No	String (6)	This field holds code specifying what type the item is. Valid values for this field are in the code type UPCT on the code_head and code_detail tables.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-418 ItemBom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentItem	No	String (25)	This field specifies unique identifier for the component item.
packQuantity	No	Number(12,4)	This field specifies contains the quantity of each item in the pack item.

Table 5-419 ItemTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketOverPercentage	No	Number	
printOnPcInd	No	String	
ticketTypeId	No	String (4)	This field uniquely identifies the ticket type which is associated with the item.
ticketTypeDescription	No	String (120)	This field contains a description of the ticket or label type.

Table 5-419 (Cont.) ItemTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shelfEdgeLabelInd	No	String (1)	This field contains a description of the ticket or label type.
poPrintType	No	String (1)	This field indicates when the ticket type for the given item should be printed by default, upon the approval or receipt of the purchase order. Valid values are: A and R.
printOnPriceChangeInd	No	String (1)	This field indicates whether or not this type of ticket should be printed for this item when a permanent price change goes into effect. Valid values in this field are:Y - Yes N - No.
ticketOverPrintPercentage	No	Number(12,4)	This field contains a percent which indicates the amount of tickets which should be printed for a given event. If the event causing tickets to be printed is approving or receiving a purchase order, then this field indicates the quantity of tickets which should be printed greater than the quantity of the purchase order. If the event causing the tickets to be printed is a permanent price change, this field would indicate the quantity of tickets which should be printed greater than the stock on hand.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-419 (Cont.) ItemTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
details	No	Collection of Object	This List holds attributes for the item ticket details.

Table 5-420 ItemTicketDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketItemId	No	String (4)	This field contains a character string which uniquely identifies an attribute which will appear on a ticket or label such as retail price or price per unit of measure.
ticketItemIdDescription	No	String (250)	This field contains the description of the ticket_item_id.
udaId	No	Number(5,0)	This field contains a number which uniquely defines a user-defined attribute which is to be printed on this ticket type.
sequenceNo	No	Number(4,0)	This field contains sequence number of item's ticket record.

Table 5-421 RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	No	Number(20,0)	A system generated numeric field that uniquely identifies the relationship record in related_item_head table.
relationshipName	No	String (255)	This field holds user entered name of the relationship.
relationshipType	No	String (6)	This field holds user entered name of the relationship.
mandatoryInd	No	String (1)	Indicates whether the relationship is mandatory or not. Valid values: 'Y' (yes), 'N' (no).

Table 5-421 (Cont.) RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
translation	No	Collection of Object	This List holds translation attributes for the relationship.
details	No	Collection of Object	This List holds attributes for the related item details.

Table 5-422 RelatedItemTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	This field contains the language in which the translated text is maintained
languageName	No	String (120)	This field contains description of language code in which the translated text is maintained
isoCode	No	String (6)	This field will hold the ISO code associated with the given language.
relationshipName	No	String (255)	This field holds the translated value of the relationship name
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-423 RelatedItemDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relatedItem	No	String (25)	Item for which related items are defined. Valid transaction level/sellable item. Multiple items can be defined for a relationship.

Table 5-423 (Cont.) RelatedItemDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
priority	No	Number(4,0)	This field indicates relationship priority. Applicable only for relationship type Substitute.
startDate	No	date	The related item can be used on transactions starting this date.
endDate	No	date	This field related item can be used on transactions until this date. A value of null means that it is always effective.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-424 ItemHts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hts	No	String (25)	The unique identifier for the Harmonized Tariff Schedule code.
importCountry	No	String (3)	The unique identifier for the country that the item will be imported into.
originCountry	No	String (3)	The country where the item was manufactured or significantly altered.
effectFrom	No	date	The date from which the Item/HTS relationship is valid. This field is used to maintain historical information related to the Item/HTS relationship.

Table 5-424 (Cont.) ItemHts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
effectTo	No	date	The date until when the Item/HTS relationship is valid. This field is used to maintain historical information related to the item/HTS relationship.
clearingZoneId	No	String (5)	The concept of 'clearing zone' represents a country's zone-level point of entry of goods (for example, zones within the country where the goods are clearing customs). No locations or cost zones are associated to a clearing zone. This field holds the Clearing Zone ID for the effective HTS code. Clearing zones are defined by import country.
status	No	String (1)	Used to indicate the status of the Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
assessments	No	Collection of Object	References a collection of assessments associated to the item HTS.

Table 5-425 ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	No	String (10)	The code representing the specific assessment component.

Table 5-425 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
computationValueBase	No	String (10)	The Computation Value Base representing the base value for calculating the assessment.
componentRate	No	Number(20,4)	The rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount if the Calculation Basis is Specific.
perCount	No	Number(12,4)	A count indicating the amount of the Per Count Unit of Measure to which the rate applies. This column is only used for specific (per-unit) assessments.
perCountUom	No	String (4)	The unit of measure in which the Per Count is specified. This column is only used from specific (per-unit) assessments.
estimatedAssessmentValue	No	Number(26,10)	Contains the calculated value of the Component.
componentCurrency	No	String (3)	This field contains component currency.

Table 5-425 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag1	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	No	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-425 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	No	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.

Table 5-425 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inAlc	No	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-426 ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Number(10,0)	The unique identifier for the supplier.
countryOfSourcing	No	String (3)	The country where the item was manufactured or significantly altered.
ladingPort	No	String (5)	The final destination lading port of the item if the Expense Type is Country.
dischargePort	No	String (5)	The port where the item will enter the importing country.
costZoneGroupId	No	Number(4,0)	This field holds the items Zone Group, it is stored here for performance reasons.
costZoneId	No	Number(10,0)	The final destination zone of the item if the Expense Type is Zone.

Table 5-426 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
baseExpenseInd	No	String (1)	This field indicates whether or not this Set of Expenses is also the base Set of Expenses.
componentId	No	String (10)	The code representing the specific expense component.
computationValueBase	No	String (10)	The Computation Value Base representing the base value for calculating the expense.
componentRate	No	Number(20,4)	The rate to be charged against the calculation base. This field will hold a percentage value if the Calculation basis of the Component is Value or a monetary amount if the Calculation Basis is Specific.
perCount	No	Number(12,4)	A count indicating the amount of the Per Count Unit of Measure to which the rate applies. This column is only used for specific (per-unit) expenses.
perCountUom	No	String (4)	The unit of measure in which the Per Count is specified. This column is only used from specific (per-unit) expenses.
estimatedExpenseValue	No	Number(26,10)	This field contains the calculated value of the Component.
componentCurrency	No	String (3)	This field contains the currency the expense is to be entered in. If the Calculation Basis is V (Value), then the comp_currency will be NULL.

Table 5-426 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag1	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	No	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-426 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	No	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.

Table 5-426 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inAlc	No	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-427 RequiredDocuments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
documentId	No	Number(6,0)	This field contains documents of the items.

Table 5-428 ItemImportAttr - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importDescription	No	String (2000)	This field contains description of the item used for importing purposes..
commodity	No	String (6)	This field contains the code used to classify the item by their type of merchandise.
tooling	No	Number(20,4)	This field contains the cost associated with a given Tooling/Assist charge.

Table 5-428 (Cont.) ItemImportAttr - Object. See list of elements for detail

Element Name	Required	Data Type	Description
firstOrderIndicator	No	String (1)	This field contains indicator to determine if duty is to be paid on the total value of the tooling expense the first time the item is shipped
amortizeBase	No	Number(12,4)	This field contains the number of units over which the tooling/assist cost will be amortized.
openBalance	No	Number(12,4)	This field contains the open balance of the number of units yet to be received over which the tooling costs will be amortized.
createDateTime	No	dateTime	This field holds the record creation date.

Table 5-429 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "longtDescription": "String",
      "item": "100100076",
      "itemNumberType": "MANL",
      "status": "A",
      "itemLevel": 1,
      "tranLevel": 1,
      "itemDescription": "Regular Item 100100076",
      "shortDescription": "100100076 Item",
      "itemDescriptionUppercase": "100100076 ITEM REGULAR",

```

```
"itemDescriptionSecondary": "100100076 Item Regular",
"longDescription": null,
"itemParent": null,
"itemGrandparent": null,
"sellableInd": "Y",
"orderableInd": "Y",
"inventoryInd": "Y",
"packInd": "Y",
"simplePackInd": "N",
"containsInnerInd": "N",
"dept": 3041,
"deptName": null,
"class": 1.0,
"className": null,
"uniqueClassId": 1,
"subclass": 1,
"subclassName": null,
"uniqueSubclassId": 1,
"unitRetail": 1.0,
"variableUpcFormatId": null,
"variableWeightUpcPrefix": 1,
"diff1": null,
"diff1Type": null,
"diff2": null,
"diff2Type": null,
"diff3": null,
"diff3Type": null,
"diff4": null,
"diff4Type": null,
"costZoneGroupId": 1000,
"standardUom": "EA",
"uomConversionFactor": 3,
"packageSize": 1.0,
"packageUom": null,
"merchandiseInd": "Y",
"storeOrderMultiple": "E",
"forecastInd": null,
"manufacturerRecommendedRetail": 12.99,
"manufacturerRetailCurrencyCode": "USD",
"originalRetail": 12.99,
"originalRetailCurrencyCode": "USD",
"retailLabelType": null,
"retailLabelTypeDescription": null,
"retailLabelValue": 1.0,
"handlingTemperature": null,
"handlingTemperatureDescription": null,
"handlingSensitivity": null,
"handlingSensitivityDescription": null,
"catchWeightInd": "Y",
"catchWeightType": null,
"catchWeightUom": "EA",
"orderType": null,
"saleType": null,
"wasteType": null,
"wasteTypeDescription": null,
"averageWastePercentage": 1.0,
```



```

"defaultWastePercentage": 1.0,
"constantDimensionInd": "Y",
"packType": null,
"orderAsType": null,
"comments": null,
"itemServiceLevel": null,
"giftWrapInd": "Y",
"shipAloneInd": "Y",
"brandName": null,
"brandDescription": null,
"perishableInd": "Y",
"itemTransformInd": "N",
"depositItemType": null,
"containerItem": null,
"depositInPricePerUom": null,
"storePackInventoryInd": "Y",
"sohInquiryAtPackInd": "Y",
"purchaseType": null,
"productClassification": null,
"productClassificationDescription": null,
"itemAggregateInd": "Y",
"diff1AggregateInd": "Y",
"diff2AggregateInd": "Y",
"diff3AggregateInd": "Y",
"diff4AggregateInd": "Y",
"diff1Level": null,
"diff1Description": null,
"diff2Level": null,
"diff2Description": null,
"diff3Level": null,
"diff3Description": null,
"diff4Level": null,
"diff4Description": null,
"primaryImageUrl": null,
"transferAllocUpChargeSourceType": "D",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemTranslation": [
{
"languageCode": 2,
"languageName": null,
"isoCode": "en-US",
"itemDescription": "itemDesc",
"itemDescriptionSecondary": "itemDescSecondary",
"shortDescription": "shortDesc",
"longDescription": null,
"createDateTime": "2001-12-31T23:59:59.000Z",

```

```

"updateDateTime": "2001-12-31T23:59:59.000Z"
},
],
"itemSupplier": [
{
"supplier": 2400,
"primarySupplierInd": "Y",
"vpn": null,
"supplierLabel": null,
"consignmentRate": 3,
"supplierDiff1": null,
"supplierDiff2": null,
"supplierDiff3": null,
"supplierDiff4": null,
"palletName": "PAL",
"caseName": "CS",
"innerName": "INR",
"supplierDiscontinueDate": "2001-12-31",
"directShipInd": "Y",
"primaryCaseSize": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemSupplierCountry": [
{
"originCountry": null,
"primarySupplierInd": null,
"primaryCountryInd": "Y",
"unitCost": 9.65,
"leadTime": 1,
"pickupLeadTime": 1,
"supplierPackSize": 1,
"innerPackSize": 1,
"roundLevel": "C",
"minimumOrderQuantity": 1.0,
"maximumOrderQuantity": 1,
"packingMethod": "HANG",
"defaultUop": null,
"ti": 1,
"hi": 1,
"costUom": "EA",
"toleranceType": null,
"maximumTolerance": 1.0,
"minimumTolerance": 1.0,
"roundToInnerPercentage": 50,
"roundToCasePercentage": 50,
"roundToLayerPercentage": 50,
"roundToPalletPercentage": 50,

```

```

"supplierHierarchyType1": null,
"supplierHierarchyLevel1": null,
"supplierHierarchyType2": null,
"supplierHierarchyLevel2": null,
"supplierHierarchyType3": null,
"supplierHierarchyLevel3": null,
"negotiatedItemCost": 1.0,
"extendedBaseCost": 1.0,
"inclusiveCost": 1.0,
"baseCost": 1.0,
"purchaseType": 0,
"calculationBasis": null,
"purchaseRate": 1.0,
"pricingControl": "R",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemSupplierCountryDimension": [
{
"dimensionObject": null,
"dimensionObjectDescription": null,
"presentationMethod": null,
"presentationMethodDescription": null,
"length": 1.0,
"width": 1.0,
"height": 1.0,
"lwhUom": null,
"weight": 1.0,
"netWeight": 1.0,
"weightUom": null,
"liquidVolume": 1.0,
"liquidVolumeUom": null,
"statisticalCase": 1.0,
"tareWeight": 1.0,
"tareType": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
},
"itemSupplierCountryOfManufacture": [
{
"countryId": "BR",
"primaryInd": "N",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
}

```

```

    ],
    "itemSupplierUom": [
      {
        "uom": null,
        "value": 1.0,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ],
    "translation": [
      {
        "languageCode": 2,
        "languageName": null,
        "isoCode": "en-US",
        "supplierDiff1": null,
        "supplierDiff2": null,
        "supplierDiff3": null,
        "supplierDiff4": null,
        "supplierLabel": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ]
  },
  "itemUda": {
    "udaLov": [
      {
        "udaId": 1,
        "udaDescription": null,
        "udaValue": 1,
        "udaValueDescription": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ],
    "udaFreeform": [
      {
        "udaId": 1,
        "udaDescription": null,
        "udaText": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z",
        "translation": [
          {
            "languageCode": 2,
            "languageName": null,
            "isoCode": "en-US",
            "udaTextDesc": null,
            "createDateTime": "2001-12-31T23:59:59.000Z",
            "updateDateTime": "2001-12-31T23:59:59.000Z"
          }
        ]
      }
    ]
  },
  "udaDate": [

```

```

{
  "udaId": 1,
  "udaDescription": null,
  "udaDate": "2001-12-31",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemImage": [
{
  "imageName": "imageName1",
  "imageAddress": "http://www.oracle.com/",
  "imageType": "T",
  "primaryImageInd": "Y",
  "displayPriority": 1,
  "imageDescription": "imageDesc1",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z",
  "translation": [
    {
      "languageCode": 2,
      "languageName": null,
      "isoCode": "en-US",
      "imageDescription": "Item Image",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ]
}
],
"itemSeason": [
{
  "seasonId": 1,
  "phaseId": 1,
  "sequenceNo": 1,
  "diffId": null,
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"referenceItem": [
{
  "referenceItem": null,
  "primaryInd": "N",
  "formatId": null,
  "prefix": 1,
  "itemNoType": "MANL",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemBOM": [
{
  "componentItem": null,
  "packQuantity": 1.0
}
]

```

```

    },
    "itemTicket": [
      {
        "ticketOverPercentage": 1.0,
        "printOnPcInd": "String",
        "ticketTypeId": null,
        "ticketTypeDescription": null,
        "shelfEdgeLabelInd": null,
        "poPrintType": null,
        "printOnPriceChangeInd": "Y",
        "ticketOverPrintPercentage": 3,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z",
        "details": [
          {
            "ticketItemId": null,
            "ticketItemIdDescription": null,
            "udaId": 1,
            "sequenceNo": 1
          }
        ]
      }
    ],
    "relatedItem": [
      {
        "relationshipId": 1,
        "relationshipName": null,
        "relationshipType": null,
        "mandatoryInd": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z",
        "translation": [
          {
            "languageCode": 2,
            "languageName": null,
            "isoCode": "en-US",
            "relationshipName": null,
            "createDateTime": "2001-12-31T23:59:59.000Z",
            "updateDateTime": "2001-12-31T23:59:59.000Z"
          }
        ],
        "details": [
          {
            "relatedItem": null,
            "priority": 1,
            "startDate": "2001-12-31",
            "endDate": "2001-12-31",
            "createDateTime": "2001-12-31T23:59:59.000Z",
            "updateDateTime": "2001-12-31T23:59:59.000Z"
          }
        ]
      }
    ],
    "hts": [
      {

```

```

    "hts": null,
    "importCountry": "US",
    "originCountry": "US",
    "effectFrom": "2001-12-31",
    "effectTo": "2001-12-31",
    "clearingZoneId": null,
    "status": null,
    "createDateTime": "2001-12-31T23:59:59.000Z",
    "updateDateTime": "2001-12-31T23:59:59.000Z",
    "assessments": [
      {
        "componentId": null,
        "computationValueBase": null,
        "componentRate": 1.0,
        "perCount": 1.0,
        "perCountUom": null,
        "estimatedAssessmentValue": 1.0,
        "componentCurrency": null,
        "nominationFlag1": null,
        "inDuty": null,
        "nominationFlag3": null,
        "inExpense": null,
        "inAlc": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ],
    "expense": [
      {
        "supplier": 2400,
        "countryOfSourcing": "US",
        "ladingPort": "7",
        "dischargePort": "480",
        "costZoneGroupId": 1,
        "costZoneId": 1,
        "baseExpenseInd": "N",
        "componentId": "AGCOMM",
        "computationValueBase": "TDTYUS",
        "componentRate": 3.55,
        "perCount": 3,
        "perCountUom": "EA",
        "estimatedExpenseValue": 1.0,
        "componentCurrency": "USD",
        "nominationFlag1": null,
        "inDuty": null,
        "nominationFlag3": null,
        "inExpense": null,
        "inAlc": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ],
    "requiredDocuments": [
      {

```

```

"documentId": 1
},
],
"itemImportAttributes": [
{
"importDescription": "import desc",
"commodity": "A123",
"tooling": 12,
"firstOrderIndicator": "Y",
"amortizeBase": 12.4568,
"openBalance": 12.4568,
"createDateTime": "2001-12-31T23:59:59.000Z"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_MASTER_CFA_EXT	Yes	No	No	No
ITEM_SUPPLIER_CFA_EXT	Yes	No	No	No
ITEM_SUPP_COUNTRY_CFA_EXT	Yes	No	No	No
ITEM_SUPP_COUNTRY_LOC_CFA_EXT	Yes	No	No	No
MERCHAPI_EXT_ITEM	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ITEM_JSON	Yes	No	No	No
ICL_RMS_ITEM_MASTER	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPPLIER	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPP_UOM	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPP_CTRY	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPP_CTRY_DIM	Yes	No	Yes	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
ICL_RMS_ITEM_SUPP_MANU_CTRY	Yes	No	Yes	Yes
ICL_RMS_ITEM_TICKET	Yes	No	Yes	Yes
ICL_RMS_UDA_ITEM_DATE	Yes	No	Yes	Yes
ICL_RMS_UDA_ITEM_LOV	Yes	No	Yes	Yes
ICL_RMS_UDA_ITEM_FF	Yes	No	Yes	Yes
ICL_RMS_ITEM_IMAGE	Yes	No	Yes	Yes
ICL_RMS_ITEM_SEASONS	Yes	No	Yes	Yes
ICL_RMS_RELATED_ITEM	No	Yes	No	No
ICL_RMS_ITEM HTS	Yes	No	Yes	Yes
ICL_RMS_ITEM_EXP	Yes	No	Yes	Yes

Get Item Details for Provided Item

Endpoints

MerchIntegrations/services/foundation/item

MerchIntegrations/services/foundation/item/{itemId}

Functional Area

Items - Item Definition

Business Overview

This service publishes item master data for transaction-level items and higher merchandise hierarchy nodes to downstream consuming systems. The payload combines core item attributes with translations, supplier and sourcing details, UDA values, images, tickets, related items, compliance and import data, and custom flex attributes. End-to-end publication is limited to items that qualify through the refresh and delta flow: refresh seeds items in approved or deleted status, item-master inserts are not queued until the item is approved, and item-master updates in worksheet or submitted status are suppressed. Below-transaction-level reference or UPC items are not published as standalone records; their changes refresh the parent item's referenceItem node.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_MASTER, ITEM_MASTER_TL, ITEM_SUPPLIER, ITEM_SUPP_UOM, ITEM_SUPP_COUNTRY, ITEM_SUPP_COUNTRY_DIM, ITEM_SUPP_MANU_COUNTRY, ITEM_TICKET, UDA_ITEM_DATE, UDA_ITEM_LOV, UDA_ITEM_FF, ITEM_IMAGE, ITEM_SEASONS, ITEM HTS, ITEM_EXP_HEAD, ITEM_EXP_DETAIL, RELATED_ITEM_HEAD, RELATED_ITEM_DETAIL, REQ_DOC, ITEM_IMPORT_ATTR

Additional business tables: ITEM_MASTER_CFA_EXT, ITEM_SUPPLIER_CFA_EXT, ITEM_SUPP_COUNTRY_CFA_EXT, DEPS, GROUPS, DIVISION, CLASS, SUBCLASS,

BRAND, DIFF_IDS, DIFF_GROUP_HEAD, SYSTEM_OPTIONS,
SYSTEM_CONFIG_OPTIONS, CODE_DETAIL

JSON cache table: MERCHAPI_EXT_ITEM

JSON generation view: V_MERCHAPI_EXT_ITEM_JSON

Is keyed by ITEM_MASTER rows with ITEM_LEVEL <= TRAN_LEVEL, so the published entity is the transaction-level item or a higher hierarchy node.

The payload combines core item attributes with translations, supplier and sourcing data, UDAs, images, tickets, related items, HTS and expense data, import attributes, custom flex attributes, and supporting hierarchy and lookup values.

Below-transaction-level reference or UPC items are not published as standalone payloads; their data is rolled into the parent item's referenceItem node.

Initial Seeding and post enablement rebuild/repair

Initial seeding selects ITEM_MASTER rows where STATUS IN ('A', 'D') and ITEM_LEVEL <= TRAN_LEVEL, then builds the published JSON through V_MERCHAPI_EXT_ITEM_JSON.

Rebuild marks cache rows deleted when the item no longer exists in ITEM_MASTER.

ICL (Integration Change Log) table: ICL_RMS_ITEM_MASTER, ICL_RMS_ITEM_SUPPLIER, ICL_RMS_ITEM_SUPP_UOM, ICL_RMS_ITEM_SUPP_CTRY, ICL_RMS_ITEM_SUPP_CTRY_DIM, ICL_RMS_ITEM_SUPP_MANU_CTRY, ICL_RMS_ITEM_TICKET, ICL_RMS_UDA_ITEM_DATE, ICL_RMS_UDA_ITEM_LOV, ICL_RMS_UDA_ITEM_FF, ICL_RMS_ITEM_IMAGE, ICL_RMS_ITEM_SEASONS, ICL_RMS_ITEM HTS, ICL_RMS_ITEM_EXP, ICL_RMS_RELATED_ITEM, ICL_RMS_REQ_DOC, ICL_RMS_ITEM_IMPORT_ATTR (ICL consumer name = MERCHAPI)

Item-master inserts are staged only when the new item is approved, approval transitions are staged as insert-style publishes, and item-master updates in worksheet or submitted status are suppressed.

ITEM_MASTER_TL changes are queued only for items with ITEM_LEVEL <= TRAN_LEVEL, and below-transaction-level item changes are rolled up to ITEM_PARENT so the parent payload is refreshed.

Item Approval Handling

Approval of an item also results in publish of foundation/omnichannel/item/image, foundation/omnichannel/item/relateditem, foundation/item/location, foundation/item/vat, and foundation/taxLocationItem. The item approval flow writes the existing ITEM_IMAGE, RELATED_ITEM_HEAD/DETAIL, ITEM_LOC, and VAT_ITEM source rows for that item into those APIs' ICL tables so dependent approved-item data is queued together with the item.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ITEM

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job: MERCHAPI_DELTA_EXT_ITEM_ADHOC_PROCESS -
> MERCHAPI_EXT_ITEM_ADHOC_JOB

Webhook configuration api name: foundation/item

Special handling

Additional GET query parameters: itemLevel, tranLevel, deptId, classId, subclassId, status, itemType, inventoryInd, supplier, referenceItem.

The filters itemLevel, tranLevel, deptId, classId, subclassId, status, and itemType are not backed by dedicated cache-table indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. The available configuration for this service should therefore be reviewed and validated against the customer production data profile, including expected volume, message size, and access pattern, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for initial seeding, while delta processing remains a single threaded process. Initial-seeding thread count is controlled through `API_REFRESH_OVERRIDE_NO_OF_THREADS`, and initial-seeding commit sizing is controlled through `API_REFRESH_OVERRIDE_COMMIT_MAX_COUNTER`.

For initial seeding, thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the item message from `ITEM_MASTER` together with supplier, UDA, image, ticket, related-item, HTS, expense, import, and hierarchy data. Common starting points for large-volume seeding are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment. The delivered refresh override is currently 4 threads.

Refresh commit sizing should be reviewed together with the thread setting. The delivered refresh override commit setting is currently 10000 and that should be used if the thread value is low. If running with higher thread value, the typical value can range from about 100 to 1,000 rows.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. Even with refresh-override threading, larger parallel insert volumes can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes.

This can improve data-load throughput, but the overall refresh plan should also account for the additional time needed for index rebuild after the seeding step finishes.

`GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a need to change it.

REST fetch considerations

Item payloads can be comparatively large, so REST page size should be selected based on average message size so that the average response remains below about 10 MB per call. Common page-limit values for this service should be 500 or 1,000, and very small limits, especially below 100, should generally be avoided.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the LAST_UPDATE_DATETIME index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping BUILD_HISTOGRAM enabled allows the service to support recovery through smaller, more selective polling windows using the optimalBefore query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides optimalBefore and leaves before empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups LAST_UPDATE_DATETIME values into buckets of 10,000 rows in MERCHAPI_TIMESTAMP_HISTOGRAM using the shared HISTOGRAM_ROW_COUNT configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/item

MerchIntegrations/services/foundation/item/{itemId}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
itemLevel	No	String	Item Level - 1, 2, or 3
tranLevel	No	String	Transaction Level- 1, 2, or 3
deptId	No	String (4)	Department Id
classId	No	String (4)	Class Id
subclassId	No	String (4)	Sub Class Id
status	No	String	Status
itemType	No	String	Item Type
inventoryInd	No	String	Inventoried
supplier	No	String	Supplier
referenceItem	No	String	Reference Item
offsetkey	No	String	Offset key (Item)
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-430 Input Parameter for "MerchIntegrations/services/foundation/item/{itemId}"

Parameter Name	Required	Data Type	Description
itemId	Yes	String	Item Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-431 PageResultsItemsMerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-431 (Cont.) PageResultsItemsMerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-432 MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for Item details
longtDescription	No	String	
item	No	String (25)	This field specifies Unique alphanumeric value that identifies the item.
itemNumberType	No	String (6)	This field holds the type of item number. Valid values are defined by code type UPCT.
status	No	String (1)	This field specifies status of the item.
itemLevel	No	Number(1,0)	This field specifies Number indicating which of the three levels the item resides. The item level determines if the item stands alone or if it is part of a family of related items. The item level also determines how the item may be used throughout the system.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
tranLevel	No	Number(1,0)	This field specifies Number indicating which of the three levels transactions occur for the item's group. The transaction level is the level at which the item's inventory is tracked in the system. The transaction level item will be counted, transferred, shipped, etc. The transaction level may be at the current item or up to 2 levels above or below the current item. Only one level of the hierarchy of an item family may contain transaction level items.
itemDescription	No	String (250)	This field specifies long description of the item. This description is used through out the system to help online users identify the item. For items that have parents, this description will default to the parents description plus any differentiators. For items without parents this description will default to null.
shortDescription	No	String (120)	This field specifies shortened description of the item. This description is the default for downloading to the POS. For items that have parents, this description will default to the parent's short description. For items without parents, this description will default to null.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemDescriptionUpper case	No	String (250)	This field specifies all upper case description of the item (same as upper(item_desc)). This field is not displayed in the on-line item maintenance dialog, but is used in item list of value search processing through out the system.
itemDescriptionSecondary	No	String (250)	This field specifies secondary descriptions of the item. This field can only be populated when system_options.secondary_desc_ind = Y.
longDescription	No	String (2000)	This field specifies long description of the item. This is a more descriptive writeup about the item targeted largely for platforms like Ecommerce platforms, Marketplaces, Order Management Systems etc.
itemParent	No	String (25)	This field specifies alphanumeric value that uniquely identifies the item/group at the level above the item.
itemGrandparent	No	String (25)	This field specifies alphanumeric value that uniquely identifies the item/group at the level above the item.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sellableInd	No	String (1)	This field indicates if pack item may be sold as a unit. If it is 'Y' then the pack will have its own unique unit retail. If it is 'N' then the pack's unit retail is the sum of each individual item's total retail within the pack. This field will only be available if the item is a pack item. Value values are: Y = Yes, this pack may be sold as a unit; N = No, this pack may not be sold as a unit.
orderableInd	No	String (1)	This field indicates if pack item is orderable. If it is 'Y' then the suppliers of the pack must supply all components in the pack. If it is 'N' then the components may have different suppliers. This field will only be available if the item is a pack item. Valid values are: Y = Yes, this pack may be ordered, N = No, this pack may not be ordered.
inventoryInd	No	String (1)	This field specifies inventory Indicator.
packInd	No	String (1)	This field indicates if the item is a pack. A pack item is a collection of items that may be either ordered or sold as a unit. Packs require details (i.e. component items and qtys, etc.) that other items do not. This field is required by the database.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
simplePackInd	No	String (1)	This field indicates if pack item is a simple pack or not. This field will only be available if the item is a pack item. A simple pack is an item whose components are all the same item (i.e. a six pack of cola, etc). Valid values are: Y = Yes, this item is a simple pack; N = No if this is not a simple pack.
containsInnerInd	No	String (1)	This field indicates if pack item contains inner packs. Vendor packs will never contain inner packs and this field will be defaulted to 'N'. This field will only be available if the item is a pack item. Valid values are Y = Yes, this pack contains inner packs; N = No, this pack does not contain inner packs.
dept	No	Number(4,0)	This field specifies number identifying the department to which the item is attached. The item's department will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the depts table.
deptName	No	String (120)	This field contains the name of the department.
class	No	Number	
className	No	String (120)	This field contains the name of the class.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uniqueClassId	No	Number(10,0)	This field uniquely identify the class which the item belongs to. Class is not unique ID the merchandise hierarchy. The combination of Dept/Class is unique, but requires use of a composite key. The composite key is generally used in user interfaces. The unique ID can be used in back end processing or in systems that can not have a composite key in the merchandise hierarchy.
subclass	No	Number(4,0)	This field specifies number identifying the subclass to which the item is attached. The item's subclass will be the same as that of its parent (and, by transitivity, to that of its grandparent). Valid values for this field are located on the subclass table.
subclassName	No	String (120)	This field contains the name of the subclass.
uniqueSubclassId	No	Number(10,0)	This field uniquely identify the subclass which the item belongs to. Subclass is not unique ID the merchandise hierarchy. The combination of Dept/Class/Subclass is unique, but requires use of a composite key. The composite key is generally used in user interfaces. The unique ID can be used in back end processing or in systems that can not have a composite key in the merchandise hierarchy.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitRetail	No	Number(20,4)	This field holds the unit retail in the standard unit of measure for the item/zone combination. This field is stored in the local currency.
variableUpcFormatId	No	String (1)	This field will hold the format ID that corresponds to the item's variable UPC. This value is only used for items with variable UPCs.
variableWeightUpcPrefix	No	Number(2,0)	This field holds the prefix for variable weight UPCs. The prefix determines the format of the eventual UPC and will be used to decode variable weight UPCs that are uploaded from the POS. It is the clients responsibility to download this value to their POS system.
diff1	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff1Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
diff2	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
diff2Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff3	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.
diff3Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
diff4	No	String (10)	This field contains diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Men's pant sizes) or a value (6 oz). For an item that is not a parent, this field may contain a value (34X34, Red, etc.). Valid values are found on the diff_group and diff_id tables.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff4Type	No	String (6)	This field will hold a value of all possible differential types. (for example, 'S' - size, 'C' - color, 'F' - flavor, 'E' - scent, 'P' - pattern). Valid values are stored on the code_detail table with the code_type 'DIFF' and on the diff_type table.
costZoneGroupId	No	Number(4,0)	This field contains cost zone group associated with the item. This field is only required when elc_ind (landed cost indicator) is set to 'Y' on the system_options table.
standardUom	No	String (4)	This field contains unit of measure in which stock of the item is tracked at a corporate level.
uomConversionFactor	No	Number(20,10)	This field contains Conversion factor between an 'Each' and the standard_uom when the standard_uom is not in the quantity class (e.g. if standard_uom = lb and 1 lb = 10 eaches, this factor will be 10). This factor will be used to convert sales and stock data when an item is retailed in eaches but does not have eaches as its standard unit of measure.
packageSize	No	Number(12,4)	This field holds the size of the product printed on any packaging (i.e. 24 ounces). This field is used for reporting purposes, as well as by Retek Price Mangement to determine same sized and different sized items.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
packageUom	No	String (4)	This field holds the unit of measure associated with the package size. This field is used for reporting purposes and by Retek Price Management to determine same sized and different sized items.
merchandiseInd	No	String (1)	This field indicates if the item is a merchandise item (Y, N).
storeOrderMultiple	No	String (1)	This field indicates Merchandise shipped from the warehouses to the stores must be specified in this unit type. Valid values are: C = Cases; I = Inner; E = Eaches.
forecastInd	No	String (1)	This field indicates if this item will be interfaced to an external forecasting system (Y, N).
manufacturerRecomm endedRetail	No	Number(20,4)	This field contains manufacturer's recommended retail price for the item. Used for informational purposes only. This field is stored in the primary currency.
manufacturerRetailCurrencyCode	No	String (3)	The field indicates the currency code of the Manufacturer's retail price.
originalRetail	No	Number(20,4)	The field indicated the original retail price of the item per unit. This field is stored in the primary currency.
originalRetailCurrency Code	No	String (3)	The field indicates the currency code of the original retail price.
retailLabelType	No	String (6)	This field indicates any special label type associated with an item.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
retailLabelTypeDescription	No	String (250)	This field holds the description of the retail label type.
retailLabelValue	No	Number(20,4)	This field represents the value associated with the retail label type.
handlingTemperature	No	String (6)	This field holds the temperature information associated with the item.
handlingTemperatureDescription	No	String (250)	This field holds the description of the handling temperature.
handlingSensitivity	No	String (6)	This field holds the sensitivity information associated with the item.
handlingSensitivityDescription	No	String (250)	This field holds the description of the handling sensitivity.
catchWeightInd	No	String (1)	This field indicates whether the item should be weighed when it arrives at a location.
catchWeightType	No	String (1)	The field will hold catch weight type for a simple pack catch weight item. The value is based on the component items order_type and sale_type: 2 - order_type = Variable Weight, sale_type = Loose Weight 4 - order_type = Variable Weight, sale_type = Variable Weight Each.
catchWeightUom	No	String (4)	The field indicates the UOM for Catchweight Items.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderType	No	String (6)	The field determines how catch weight items are ordered. Valid values are: F - fixed weight V - Variable weight Valid values are held on the CODE_DETAIL table with a code type = ORDT.
saleType	No	String (6)	The field indicates the method of how catch weight items are sold in store locations. Valid values are: V - variable weight each L - Loose weight. Valid values are held on the CODE_DETAIL table with a code type = STPE.
wasteType	No	String (6)	The field indicates identifies the wastage type as either sales or spoilage wastage. Sales wastage occurs during processes that make an item saleable (i.e. fat is trimmed off at customer request). Spoilage wastage occurs during the products shelf life (i.e. evaporation causes the product to weigh less after a period of time). Valid values are: SP - spoilage; and SL - Sales.
wasteTypeDescription	No	String (250)	The field contains the Waste Type description corresponding to the Waste_Type value in item_master.
averageWastePercentage	No	Number(12,4)	The field indicates average percent of wastage for the item over its shelf life. Used in inflating the retail price for wastage items.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
defaultWastePercentage	No	Number(12,4)	The field indicates default daily wastage percent for spoilage type wastage items. This value will default to all item locations and represents the average amount of wastage that occurs on a daily basis.
constantDimensionInd	No	String (1)	The field indicates that the dimensions of the product are always the same, regardless of the supplier. If this field is Y, the dimensions for all suppliers will be defaulted to the primary supplier/ primary country dimensions. Editing of dimensions for the item will only be allowed for the primary supplier/ primary country.
packType	No	String (1)	The field Indicates if pack item is a vendor pack or a buyer pack. A vendor pack is a pack that the vendor or supplier recognizes and sells to the retailer. If the pack item is a vendor pack, communication with the supplier will use the vendor pack number. A buyer pack is a pack that a buyer created for internal ease of use. If the pack item is a buyer pack communication with the supplier will explode the pack out to its component items. This field will only be used if the item is a pack item. If the pack item is not orderable this field must be null. Valid values are: V = Vendor; B = Buyer.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderAsType	No	String (1)	This field indicates if pack item is receivable at the component level or at the pack level (for a buyer pack only). This field is required if pack item is an orderable buyer pack. This field must be NULL if the pack is sellable only or a vendor pack. This field will only be available if the item is a pack item. Valid Values are: E = Eaches (component level) P = Pack (buyer pack only)..
comments	No	String (2000)	The field holds any comments associated with the item.
itemServiceLevel	No	String (6)	The field Holds a value that restricts the type of shipment methods that RCOM can select for an item.
giftWrapInd	No	String (1)	This field will contain a value of 'Y' if the item is eligible to be gift wrapped.
shipAloneInd	No	String (1)	This field will contain a value of Y if the item should be shipped to the customer in a separate package.
brandName	No	String (30)	This field contains the brand associated to an item.
brandDescription	No	String (120)	This field contains the description of the brand associated with the item.
perishableInd	No	String (1)	The field is not used by RMS.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemTransformInd	No	String (1)	The field show that an item is associated with an item transformation. The item will be either the sellable item or orderable item in the transformation process.
depositItemType	No	String (6)	The field contains deposit item component type. A NULL value in this field indicates that this item is not part of a deposit item relationship. The possible values are - E - Contents A - Container Z - Crate T - Returned Item (Empty bottle) P - Complex pack (with deposit items) The Returned item is flagged only to enable these items to be mapped to a separate GL account if required.
containerItem	No	String (25)	The field holds the container item number for a contents item. This field is only populated and required if the DEPOSIT_ITEM_TYPE = E.
depositInPricePerUom	No	String (6)	This field indicates if the deposit amount is included in the price per UOM calculation for a contents item ticket. This value is only required if the DEPOSIT_ITEM_TYPE = E. Valid values are I - Include deposit amount E - Exclude deposit amount.
storePackInventoryInd	No	String (1)	This field indicates whether downstream systems should track pack inventory at the pack level.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sohInquiryAtPackInd	No	String (1)	The field indicates to the Store Inventory System if a notional simple pack item's inventory should be displayed in packs.
purchaseType	No	String (1)	The field indicates if an item is a normal, consignment or concession item - 'N' for normal, 'S' for consignment, 'C' for concession.
productClassification	No	String (6)	The field contains item combinability codes (with code type 'PCLA') which provide a way to define which items can be combined (packed or boxed) together.
productClassificationDescription	No	String (250)	This field will hold the description of the product classification.
itemAggregateInd	No	String (1)	This field holds indicator to aggregate inventory and sales for the item. Currently, this indicator is used by allocation and MFP to aggregate inventory for items. For staple items this indicator should be N.
diff1AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (e.g. Style/Color or Style/Size).
diff2AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (e.g. Style/Color or Style/Size).

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff3AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/ Diff level (e.g. Style/ Color or Style/Size.
diff4AggregateInd	No	String (1)	This field holds indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/ Diff level (e.g. Style/ Color or Style/Size.
diff1Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff1Description	No	String (120)	This field will hold the description of differentiator.
diff2Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff2Description	No	String (120)	This field will hold the description of differentiator.
diff3Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff3Description	No	String (120)	This field will hold the description of differentiator.
diff4Level	No	String (6)	This field will contain either ID or GROUP, based on whether the diff_1 is a group diff or a diff id.
diff4Description	No	String (120)	This field will hold the description of differentiator.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryImageUrl	No	String (375)	This field will hold the primary item image URL. It is only included for the ItemCre and ItemFulRep message types. Updates to the primary image for an item are published in the Item Image node.
transferAllocUpChargeSourceTypes	No	String (6)	This field impacts how up charges default to transfers and allocations when an item is added. When set to 'D', department-level up charges are applied if item-level up charges are not defined for the item/ from-location/to-location combination. When set to 'I', only item-level up charges are applied; department-level up charges are not inherited.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item.
itemTranslation	No	Collection of Object	This List holds translation attributes for the item.
itemSupplier	No	Collection of Object	This List holds supplier's attributes for the item.
itemUda	No	Object	This record holds uda attributes for the item.
itemImage	No	Collection of Object	This List holds image attributes for the item.
itemSeason	No	Collection of Object	This List holds season attributes for the item.
referenceItem	No	Collection of Object	This List holds referenc item's attributes for the item.

Table 5-432 (Cont.) MerchApiExtItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemBOM	No	Collection of Object	This List holds pack attributes for the item.
itemTicket	No	Collection of Object	This List holds ticket attributes for the item.
relatedItem	No	Collection of Object	This List holds related item's attributes for the item.
hts	No	Collection of Object	This List holds hts attributes for the item.
expense	No	Collection of Object	This List holds expense attributes for the item.
requiredDocuments	No	Collection of Object	This List holds the required documents of the item.
itemImportAttributes	No	Collection of Object	This List holds the import specific attributes associated with a given item.
cacheTimestamp	No	dateTime	This field specifies date and time when item record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies date and time when the record was first created for web service publication.

Table 5-433 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).

Table 5-433 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-434 ItemTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	This field contains the language in which the translated text is maintained
languageName	No	String (120)	This field contains description of language code in which the translated text is maintained
isoCode	No	String (6)	This field will hold the ISO code associated with the given language.
itemDescription	No	String (250)	This field will hold the ISO code associated with the given language.
itemDescriptionSecondary	No	String (250)	This field holds long description of the item. This description is used through out the system to help online users identify the item. For items that have parents, this description will default to the parents description plus any differentiators. For items without parents, this description will default to null

Table 5-434 (Cont.) ItemTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shortDescription	No	String (120)	This field holds shortened description of the item. This description is the default for downloading to the POS. For items that have parents, this description will default to the parents short description. For items without parents, this description will default to null
longDescription	No	String (2000)	This field specifies long description of the item. This is a more descriptive writeup about the item targeted largely for platforms like Ecommerce platforms, Marketplaces, Order Management Systems etc.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-435 ItemSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Number(10,0)	This field specifies unique identifier for the supplier.
primarySupplierInd	No	String (1)	This field indicates whether this supplier is the primary supplier for the item. An item can have one and only one primary supplier.
vpn	No	String (30)	This field contains the Vendor Product Number for the Item/ Supplier.
supplierLabel	No	String (15)	This field contains the supplier label. It will only be populated if item is a parent item.

Table 5-435 (Cont.) ItemSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
consignmentRate	No	Number(12,4)	This field contains the consignment rate for this item for the supplier.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff4	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
palletName	No	String (6)	This field specifies code referencing the name used to refer to the pallet.
caseName	No	String (6)	This field specifies code referencing the name used to refer to the case.
innerName	No	String (6)	This field specifies code referencing the name used to refer to the inner.
supplierDiscontinueDate	No	date	This field specifies the date when the supplier discontinues an item.

Table 5-435 (Cont.) ItemSupplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
directShipInd	No	String (1)	This field indicates that any item associated with this supplier is eligible for a direct shipment from the supplier to the customer.
primaryCaseSize	No	String (6)	This field indicates the primary case size for the item supplier when an orderable item is configured for informal case types.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item's supplier record.
itemSupplierCountry	No	Collection of Object	This List holds custom flex attributes for the item supplier's country record.
itemSupplierCountryOfManufacture	No	Collection of Object	This List holds custom flex attributes for the item supplier's manufacture record.
itemSupplierUom	No	Collection of Object	This List holds custom flex attributes for the item's supplier UOM record.
translation	No	Collection of Object	List of translations for Item Supplier

Table 5-436 ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	No	String (3)	This field specifies country where the item was manufactured or significantly altered.
primarySupplierInd	No	String (1)	This field indicates whether this supplier is the primary supplier for the item. Each item can have one and only one primary supplier.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryCountryInd	No	String (1)	This field indicates whether this country is the primary country for the item/supplier. Each item/supplier combination can have one and only one primary country.
unitCost	No	Number(20,4)	This field contains the current corporate unit cost for the item from the supplier /origin country. This field is stored in the Supplier's currency..
leadTime	No	Number(4,0)	This field contains the number of days that will elapse between the date an order is written and the delivery to the store or warehouse from the supplier.
pickupLeadTime	No	Number(4,0)	This field Contains the time that will be taken to get the item from the supplier to the initial receiving location.
supplierPackSize	No	Number(12,4)	This field contains the quantity that orders must be placed in multiples of for the supplier for the item.
innerPackSize	No	Number(12,4)	This field contains the break pack size for this item from the supplier.
roundLevel	No	String (6)	This field is used to determine how order quantities will be rounded to case, layer and pallet.
minimumOrderQuantity	No	Number(12,4)	This field contains the minimum quantity that can be ordered at once from the supplier for the item.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
maximumOrderQuantity	No	Number(12,4)	This field contains the maximum quantity that can be ordered at once from the supplier for the item.
packingMethod	No	String (6)	This field indicates whether the packing method for the item in the container is flat or hanging.
defaultUop	No	String (6)	This field contains the default unit of purchase for the item/supplier/country. Valid values include 'C' (Case) and 'P' (Pallet)
ti	No	Number(12,4)	This field contains number of shipping units (cases) that make up one tier of the pallet. Multiply TI x HI to get total number of units (cases) in a Pallet.
hi	No	Number(12,4)	This field contains number of tiers that make up a complete pallet. Multiply TI x HI to get total number of units (cases) in a Pallet.
costUom	No	String (4)	This field contains cost UOM is held to allow costs to be managed in a separate UOM to the standard UOM.
toleranceType	No	String (6)	The unit of tolerance for catch weight simple packs. Valid values are: 'A' - actual 'P' - percent.
maximumTolerance	No	Number(12,4)	The maximum tolerance value for the catch weight simple pack.
minimumTolerance	No	Number(12,4)	The minimum tolerance value for the catch weight simple pack.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
roundToInnerPercentage	No	Number(12,4)	This field will hold the Inner Rounding Threshold value. During rounding, this value is used to determine whether to round partial Inner quantities up or down. If the Inner-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
roundToCasePercentage	No	Number(12,4)	This field will hold the Case Rounding Threshold value. During rounding, this value is used to determine whether to round partial Case quantities up or down. If the Case-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
roundToLayerPercentage	No	Number(12,4)	This field will hold the Layer Rounding Threshold value. During rounding, this value is used to determine whether to round partial Layer quantities up or down. If the Layer-fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
roundToPalletPercentage	No	Number(12,4)	This field will hold the Pallet Rounding Threshold value. During rounding, this value is used to determine whether to round partial Pallet quantities up or down. If the Pallet -fraction in question is less than the Threshold proportion, it is rounded down; if not, it is rounded up.
supplierHierarchyType1	No	String (6)	This field identifies partner type of supplier hierarchy level 1. This field will always have the partner type S1 which indicates manufacturer.
supplierHierarchyLevel1	No	String (10)	This field contains highest level of supplier hierarchy which is there to indicate a partner, such as a manufacturer, in the supply chain that gives rebates to the retailer.
supplierHierarchyType2	No	String (6)	This field identifies partner type of supplier hierarchy level 2. This field will always have the partner type S2 which indicates distributor.
supplierHierarchyLevel2	No	String (10)	This field contains second highest level of supplier hierarchy which is there to indicate a partner, such as a distributor, in the supply chain that gives rebates to the retailer..
supplierHierarchyType3	No	String (6)	This field identifies partner type of supplier hierarchy level 3. This field will always have the partner type S3 which indicates wholesaler.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy which is there to indicate a partner, such as a wholesaler, in the supply chain that gives rebates to the retailer.
negotiatedItemCost	No	Number(20,4)	This field will hold the supplier negotiated item cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations negotiated item cost will be stored in this field.
extendedBaseCost	No	Number(20,4)	This field will hold the extended base cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations extended base cost will be stored in this field. Extended base cost is the cost inclusive of all the taxes that affect the WAC. In case of GTAX, Extended Base Cost = Base Cost + Non-recoverable taxes. In case of VAT, Extended Base Cost = Base Cost.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inclusiveCost	No	Number(20,4)	This field will hold the inclusive cost for the primary delivery country of the item. Once a location is associated with the item, the primary locations inclusive cost will be stored in this field. This cost will have both the recoverable and non-recoverable taxes included. In case of GTAX, Inclusive Cost = Base Cost + Non-recoverable taxes + Recoverable Taxes. In case of VAT, Inclusive Cost = Base Cost + VAT.
baseCost	No	Number(20,4)	This field will hold the tax exclusive cost of the item.
purchaseType	No	Number(1,0)	This field contains a code to indicate whether the item is normal merchandise (i.e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This field indicates if the cost for the consignment/ concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number(12,4)	This field contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 5-436 (Cont.) ItemSupplierCountry - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pricingControl	No	String (1)	This field contains the pricing control of the item which will determine the item price is controlled by Retailer or Supplier. Valid value are R - Managed by Retailer, S - Managed by Supplier, NULL - For Owned Items
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item supplier's country record.
itemSupplierCountryDimension	No	Collection of Object	This List holds dimension attributes for the item supplier's country record.

Table 5-437 ItemSupplierCountryDimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	No	String (6)	This field Specific object whose dimensions are specified in this record (e.g. case, pallet, each). Valid values for this field are in the code type DIMO on the codes tables.
dimensionObjectDescription	No	String (250)	This field contains description of the dimension object.
presentationMethod	No	String (6)	This field describes the packaging (if any) being taken into consideration in the specified dimensions. Valid values for this field are in the code type PCKT on the codes tables.

Table 5-437 (Cont.) ItemSupplierCountryDimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
presentationMethodDescription	No	String (250)	This field contains description of the presentation method.
length	No	Number(12,4)	This field contains length of dim_object measured in units specified in terms of the LWH Unit of Measure.
width	No	Number(12,4)	This field contains width of dim_object measured in units specified in terms of the LWH Unit of Measure.
height	No	Number(12,4)	This field contains height of dim_object measured in units specified in terms of the LWH Unit of Measure.
lwhUom	No	String (4)	This field contains measurement for length, width, and height.
weight	No	Number(12,4)	This field contains weight of dim_object measured in units specified in terms of the Weight Unit of Measure.
netWeight	No	Number(12,4)	This field contains net weight of the dim_object (weight without packaging) measured in units specified in terms of the Weight Unit of Measure.
weightUom	No	String (4)	This field contains unit of measurement for Weight and Net Weight.
liquidVolume	No	Number(12,4)	This field contains unit of measurement for volume.

Table 5-437 (Cont.) ItemSupplierCountryDimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
liquidVolumeUom	No	String (4)	This field contains unit of measurement for liquid volume. Liquid volumes are only convertible to other liquid volumes.
statisticalCase	No	Number(12,4)	This field contains statistical value of the dim_object's dimensions to be used for loading purposes.
tareWeight	No	Number(12,4)	This field contains amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dim_object.
tareType	No	String (6)	This field indicates whether the tare weight for the dim_object is wet or dry.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-438 ItemSupplierCountryOfManufacture - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	No	String (3)	This field holds country ID of manufacturer.
primaryInd	No	String (1)	This field indicates primary country of manufacturer indicator.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-439 ItemSupplierUom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uom	No	String (4)	This field holds unit of measure that the item is being measured in.
value	No	Number(20,4)	This field holds value of measure.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-440 ItemSupplierTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	The code identifying the language.
languageName	No	String (120)	The display name of the language.
isoCode	No	String (6)	The ISO code of the language.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description.
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description.
supplierLabel	No	String (15)	The translated supplier label for the item.
createDateTime	No	dateTime	The creation date/time of the translation record.
updateDateTime	No	dateTime	The last update date/time of the translation record.

Table 5-441 ItemUda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaLov	No	Collection of Object	This List attributes for the item's uda list of values.
udaFreeform	No	Collection of Object	This List holds attributes for the item's uda free form .
udaDate	No	Collection of Object	This List holds attributes for the item's uda date.

Table 5-442 UdaLov - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	No	Number(5,0)	This field contains a number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains description of user-defined attribute.
udaValue	No	Number(5,0)	This field contains the unique identified of the Attribute Group being added or delete to the item list.
udaValueDescription	No	String (250)	This field contains the value of the Attribute Group.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-443 UdaFreeForm - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	No	Number(5,0)	This field contains a number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains description of user-defined attribute.
udaText	No	String (250)	This field contains the text value of the user-defined attribute for the item.

Table 5-443 (Cont.) UdaFreeForm - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
translation	No	Collection of Object	List of translations for UDA free form text.

Table 5-444 UdaFreeFormTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	The code identifying the language.
languageName	No	String (120)	The display name of the language.
isoCode	No	String (6)	The ISO code of the language.
udaTextDesc	No	String (250)	The translated UDA text.
createDateTime	No	dateTime	The creation date/time of the translation record.
updateDateTime	No	dateTime	The last update date/time of the translation record.

Table 5-445 UdaDate - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	No	Number(5,0)	This field contains a number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains description of user-defined attribute.
udaDate	No	date	This field contains the effective date for the UDA id.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-446 ItemImage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	No	String (120)	This field specifies name of the image.
imageAddress	No	String (255)	This field specifies path where the image file is stored.
imageType	No	String (6)	This field contains the type of the image of the item. Valid values are defined as member of IITD code type.
primaryImageInd	No	String (1)	This field specifies display sequence order of images associated to the item per priority.
displayPriority	No	Number(4,0)	This field specifies display sequence order of images associated to the item per priority.
imageDescription	No	String (40)	This field contains the description associated with the image of the item.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
translation	No	Collection of Object	List of translations for Item Image

Table 5-447 ItemImageTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	The code identifying the language.
languageName	No	String (120)	The display name of the language.
isoCode	No	String (6)	The ISO code of the language.
imageDescription	No	String (40)	This field contains the translated description associated with the image of the item.
createDateTime	No	dateTime	The creation date/time of the translation record.

Table 5-447 (Cont.) ItemImageTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	The last update date/time of the translation record.

Table 5-448 ItemSeason - Object. See list of elements for detail

Element Name	Required	Data Type	Description
seasonId	No	Number(3,0)	The season identifier.
phaseId	No	Number(3,0)	The phase identifier.
sequenceNo	No	Number(4,0)	This field contains a sequence number that combined with the item number will make each record unique.
diffId	No	String (10)	This field will hold a value for group/differentiator combination records only.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-449 ReferenceItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
referenceItem	No	String (25)	This field holds Unique identifier for the item.
primaryInd	No	String (1)	This field Indicates if the sub-transaction level item is designated as the primary sub-transaction level item. For transaction level items and above, the value in this field will be 'N'.
formatId	No	String (1)	This field will hold the format id that corresponds to the item's variable UPC. This value is only used for items with variable UPCs.

Table 5-449 (Cont.) Referenceltem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
prefix	No	Number(2,0)	This field holds the prefix for variable weight UPCs. The prefix determines the format of the eventual UPC and will be used to decode variable weight UPCs that are uploaded from the POS.
itemNoType	No	String (6)	This field holds code specifying what type the item is. Valid values for this field are in the code type UPCT on the code_head and code_detail tables.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-450 ItemBom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentItem	No	String (25)	This field specifies unique identifier for the component item.
packQuantity	No	Number(12,4)	This field specifies contains the quantity of each item in the pack item.

Table 5-451 ItemTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketOverPercentage	No	Number	
printOnPcInd	No	String	
ticketTypeId	No	String (4)	This field uniquely identifies the ticket type which is associated with the item.
ticketTypeDescription	No	String (120)	This field contains a description of the ticket or label type.

Table 5-451 (Cont.) ItemTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shelfEdgeLabelInd	No	String (1)	This field contains a description of the ticket or label type.
poPrintType	No	String (1)	This field indicates when the ticket type for the given item should be printed by default, upon the approval or receipt of the purchase order. Valid values are: A and R.
printOnPriceChangeInd	No	String (1)	This field indicates whether or not this type of ticket should be printed for this item when a permanent price change goes into effect. Valid values in this field are:Y - Yes N - No.
ticketOverPrintPercentage	No	Number(12,4)	This field contains a percent which indicates the amount of tickets which should be printed for a given event. If the event causing tickets to be printed is approving or receiving a purchase order, then this field indicates the quantity of tickets which should be printed greater than the quantity of the purchase order. If the event causing the tickets to be printed is a permanent price change, this field would indicate the quantity of tickets which should be printed greater than the stock on hand.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-451 (Cont.) ItemTicket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
details	No	Collection of Object	This List holds attributes for the item ticket details.

Table 5-452 ItemTicketDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketItemId	No	String (4)	This field contains a character string which uniquely identifies an attribute which will appear on a ticket or label such as retail price or price per unit of measure.
ticketItemIdDescription	No	String (250)	This field contains the description of the ticket_item_id.
udaId	No	Number(5,0)	This field contains a number which uniquely defines a user-defined attribute which is to be printed on this ticket type.
sequenceNo	No	Number(4,0)	This field contains sequence number of item's ticket record.

Table 5-453 RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	No	Number(20,0)	A system generated numeric field that uniquely identifies the relationship record in related_item_head table.
relationshipName	No	String (255)	This field holds user entered name of the relationship.
relationshipType	No	String (6)	This field holds user entered name of the relationship.
mandatoryInd	No	String (1)	Indicates whether the relationship is mandatory or not. Valid values: 'Y' (yes), 'N' (no).

Table 5-453 (Cont.) RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
translation	No	Collection of Object	This List holds translation attributes for the relationship.
details	No	Collection of Object	This List holds attributes for the related item details.

Table 5-454 RelatedItemTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	Number(6,0)	This field contains the language in which the translated text is maintained
languageName	No	String (120)	This field contains description of language code in which the translated text is maintained
isoCode	No	String (6)	This field will hold the ISO code associated with the given language.
relationshipName	No	String (255)	This field holds the translated value of the relationship name
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-455 RelatedItemDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relatedItem	No	String (25)	Item for which related items are defined. Valid transaction level/sellable item. Multiple items can be defined for a relationship.

Table 5-455 (Cont.) RelatedItemDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
priority	No	Number(4,0)	This field indicates relationship priority. Applicable only for relationship type Substitute.
startDate	No	date	The related item can be used on transactions starting this date.
endDate	No	date	This field related item can be used on transactions until this date. A value of null means that it is always effective.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-456 ItemHts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hts	No	String (25)	The unique identifier for the Harmonized Tariff Schedule code.
importCountry	No	String (3)	The unique identifier for the country that the item will be imported into.
originCountry	No	String (3)	The country where the item was manufactured or significantly altered.
effectFrom	No	date	The date from which the Item/HTS relationship is valid. This field is used to maintain historical information related to the Item/HTS relationship.

Table 5-456 (Cont.) ItemHts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
effectTo	No	date	The date until when the Item/HTS relationship is valid. This field is used to maintain historical information related to the item/HTS relationship.
clearingZoneId	No	String (5)	The concept of 'clearing zone' represents a country's zone-level point of entry of goods (for example, zones within the country where the goods are clearing customs). No locations or cost zones are associated to a clearing zone. This field holds the Clearing Zone ID for the effective HTS code. Clearing zones are defined by import country.
status	No	String (1)	Used to indicate the status of the Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.
assessments	No	Collection of Object	References a collection of assessments associated to the item HTS.

Table 5-457 ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	No	String (10)	The code representing the specific assessment component.

Table 5-457 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
computationValueBase	No	String (10)	The Computation Value Base representing the base value for calculating the assessment.
componentRate	No	Number(20,4)	The rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount if the Calculation Basis is Specific.
perCount	No	Number(12,4)	A count indicating the amount of the Per Count Unit of Measure to which the rate applies. This column is only used for specific (per-unit) assessments.
perCountUom	No	String (4)	The unit of measure in which the Per Count is specified. This column is only used from specific (per-unit) assessments.
estimatedAssessmentValue	No	Number(26,10)	Contains the calculated value of the Component.
componentCurrency	No	String (3)	This field contains component currency.

Table 5-457 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag1	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	No	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-457 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	No	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.

Table 5-457 (Cont.) ItemHtsAssess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inAlc	No	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-458 ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Number(10,0)	The unique identifier for the supplier.
countryOfSourcing	No	String (3)	The country where the item was manufactured or significantly altered.
ladingPort	No	String (5)	The final destination lading port of the item if the Expense Type is Country.
dischargePort	No	String (5)	The port where the item will enter the importing country.
costZoneGroupId	No	Number(4,0)	This field holds the items Zone Group, it is stored here for performance reasons.
costZoneId	No	Number(10,0)	The final destination zone of the item if the Expense Type is Zone.

Table 5-458 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
baseExpenseInd	No	String (1)	This field indicates whether or not this Set of Expenses is also the base Set of Expenses.
componentId	No	String (10)	The code representing the specific expense component.
computationValueBase	No	String (10)	The Computation Value Base representing the base value for calculating the expense.
componentRate	No	Number(20,4)	The rate to be charged against the calculation base. This field will hold a percentage value if the Calculation basis of the Component is Value or a monetary amount if the Calculation Basis is Specific.
perCount	No	Number(12,4)	A count indicating the amount of the Per Count Unit of Measure to which the rate applies. This column is only used for specific (per-unit) expenses.
perCountUom	No	String (4)	The unit of measure in which the Per Count is specified. This column is only used from specific (per-unit) expenses.
estimatedExpenseValue	No	Number(26,10)	This field contains the calculated value of the Component.
componentCurrency	No	String (3)	This field contains the currency the expense is to be entered in. If the Calculation Basis is V (Value), then the comp_currency will be NULL.

Table 5-458 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag1	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	No	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-458 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	No	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.

Table 5-458 (Cont.) ItemExpense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inAlc	No	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.
createDateTime	No	dateTime	This field holds the record creation date.
updateDateTime	No	dateTime	This field holds the date when the record was last updated.

Table 5-459 RequiredDocuments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
documentId	No	Number(6,0)	This field contains documents of the items.

Table 5-460 ItemImportAttr - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importDescription	No	String (2000)	This field contains description of the item used for importing purposes..
commodity	No	String (6)	This field contains the code used to classify the item by their type of merchandise.
tooling	No	Number(20,4)	This field contains the cost associated with a given Tooling/Assist charge.

Table 5-460 (Cont.) ItemImportAttr - Object. See list of elements for detail

Element Name	Required	Data Type	Description
firstOrderIndicator	No	String (1)	This field contains indicator to determine if duty is to be paid on the total value of the tooling expense the first time the item is shipped
amortizeBase	No	Number(12,4)	This field contains the number of units over which the tooling/assist cost will be amortized.
openBalance	No	Number(12,4)	This field contains the open balance of the number of units yet to be received over which the tooling costs will be amortized.
createDateTime	No	dateTime	This field holds the record creation date.

Table 5-461 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "longtDescription": "String",
      "item": "100100076",
      "itemNumberType": "MANL",
      "status": "A",
      "itemLevel": 1,
      "tranLevel": 1,
      "itemDescription": "Regular Item 100100076",
      "shortDescription": "100100076 Item",
      "itemDescriptionUppercase": "100100076 ITEM REGULAR",

```

```
"itemDescriptionSecondary": "100100076 Item Regular",
"longDescription": null,
"itemParent": null,
"itemGrandparent": null,
"sellableInd": "Y",
"orderableInd": "Y",
"inventoryInd": "Y",
"packInd": "Y",
"simplePackInd": "N",
"containsInnerInd": "N",
"dept": 3041,
"deptName": null,
"class": 1.0,
"className": null,
"uniqueClassId": 1,
"subclass": 1,
"subclassName": null,
"uniqueSubclassId": 1,
"unitRetail": 1.0,
"variableUpcFormatId": null,
"variableWeightUpcPrefix": 1,
"diff1": null,
"diff1Type": null,
"diff2": null,
"diff2Type": null,
"diff3": null,
"diff3Type": null,
"diff4": null,
"diff4Type": null,
"costZoneGroupId": 1000,
"standardUom": "EA",
"uomConversionFactor": 3,
"packageSize": 1.0,
"packageUom": null,
"merchandiseInd": "Y",
"storeOrderMultiple": "E",
"forecastInd": null,
"manufacturerRecommendedRetail": 12.99,
"manufacturerRetailCurrencyCode": "USD",
"originalRetail": 12.99,
"originalRetailCurrencyCode": "USD",
"retailLabelType": null,
"retailLabelTypeDescription": null,
"retailLabelValue": 1.0,
"handlingTemperature": null,
"handlingTemperatureDescription": null,
"handlingSensitivity": null,
"handlingSensitivityDescription": null,
"catchWeightInd": "Y",
"catchWeightType": null,
"catchWeightUom": "EA",
"orderType": null,
"saleType": null,
"wasteType": null,
"wasteTypeDescription": null,
"averageWastePercentage": 1.0,
```

```

"defaultWastePercentage": 1.0,
"constantDimensionInd": "Y",
"packType": null,
"orderAsType": null,
"comments": null,
"itemServiceLevel": null,
"giftWrapInd": "Y",
"shipAloneInd": "Y",
"brandName": null,
"brandDescription": null,
"perishableInd": "Y",
"itemTransformInd": "N",
"depositItemType": null,
"containerItem": null,
"depositInPricePerUom": null,
"storePackInventoryInd": "Y",
"sohInquiryAtPackInd": "Y",
"purchaseType": null,
"productClassification": null,
"productClassificationDescription": null,
"itemAggregateInd": "Y",
"diff1AggregateInd": "Y",
"diff2AggregateInd": "Y",
"diff3AggregateInd": "Y",
"diff4AggregateInd": "Y",
"diff1Level": null,
"diff1Description": null,
"diff2Level": null,
"diff2Description": null,
"diff3Level": null,
"diff3Description": null,
"diff4Level": null,
"diff4Description": null,
"primaryImageUrl": null,
"transferAllocUpChargeSourceType": "D",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemTranslation": [
{
"languageCode": 2,
"languageName": null,
"isoCode": "en-US",
"itemDescription": "itemDesc",
"itemDescriptionSecondary": "itemDescSecondary",
"shortDescription": "shortDesc",
"longDescription": null,
"createDateTime": "2001-12-31T23:59:59.000Z",

```

```

"updateDateTime": "2001-12-31T23:59:59.000Z"
},
],
"itemSupplier": [
{
"supplier": 2400,
"primarySupplierInd": "Y",
"vpn": null,
"supplierLabel": null,
"consignmentRate": 3,
"supplierDiff1": null,
"supplierDiff2": null,
"supplierDiff3": null,
"supplierDiff4": null,
"palletName": "PAL",
"caseName": "CS",
"innerName": "INR",
"supplierDiscontinueDate": "2001-12-31",
"directShipInd": "Y",
"primaryCaseSize": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemSupplierCountry": [
{
"originCountry": null,
"primarySupplierInd": null,
"primaryCountryInd": "Y",
"unitCost": 9.65,
"leadTime": 1,
"pickupLeadTime": 1,
"supplierPackSize": 1,
"innerPackSize": 1,
"roundLevel": "C",
"minimumOrderQuantity": 1.0,
"maximumOrderQuantity": 1,
"packingMethod": "HANG",
"defaultUop": null,
"ti": 1,
"hi": 1,
"costUom": "EA",
"toleranceType": null,
"maximumTolerance": 1.0,
"minimumTolerance": 1.0,
"roundToInnerPercentage": 50,
"roundToCasePercentage": 50,
"roundToLayerPercentage": 50,
"roundToPalletPercentage": 50,

```

```

"supplierHierarchyType1": null,
"supplierHierarchyLevel1": null,
"supplierHierarchyType2": null,
"supplierHierarchyLevel2": null,
"supplierHierarchyType3": null,
"supplierHierarchyLevel3": null,
"negotiatedItemCost": 1.0,
"extendedBaseCost": 1.0,
"inclusiveCost": 1.0,
"baseCost": 1.0,
"purchaseType": 0,
"calculationBasis": null,
"purchaseRate": 1.0,
"pricingControl": "R",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemSupplierCountryDimension": [
{
"dimensionObject": null,
"dimensionObjectDescription": null,
"presentationMethod": null,
"presentationMethodDescription": null,
"length": 1.0,
"width": 1.0,
"height": 1.0,
"lwhUom": null,
"weight": 1.0,
"netWeight": 1.0,
"weightUom": null,
"liquidVolume": 1.0,
"liquidVolumeUom": null,
"statisticalCase": 1.0,
"tareWeight": 1.0,
"tareType": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
},
"itemSupplierCountryOfManufacture": [
{
"countryId": "BR",
"primaryInd": "N",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
}

```

```

    ],
    "itemSupplierUom": [
    {
      "uom": null,
      "value": 1.0,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
    ],
    "translation": [
    {
      "languageCode": 2,
      "languageName": null,
      "isoCode": "en-US",
      "supplierDiff1": null,
      "supplierDiff2": null,
      "supplierDiff3": null,
      "supplierDiff4": null,
      "supplierLabel": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
    ]
  },
  "itemUda": {
    "udaLov": [
    {
      "udaId": 1,
      "udaDescription": null,
      "udaValue": 1,
      "udaValueDescription": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
    ],
    "udaFreeform": [
    {
      "udaId": 1,
      "udaDescription": null,
      "udaText": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "translation": [
      {
        "languageCode": 2,
        "languageName": null,
        "isoCode": "en-US",
        "udaTextDesc": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
      ]
    }
    ]
  },
  "udaDate": [

```

```

{
  "udaId": 1,
  "udaDescription": null,
  "udaDate": "2001-12-31",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemImage": [
{
  "imageName": "imageName1",
  "imageAddress": "http://www.oracle.com/",
  "imageType": "T",
  "primaryImageInd": "Y",
  "displayPriority": 1,
  "imageDescription": "imageDesc1",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z",
  "translation": [
    {
      "languageCode": 2,
      "languageName": null,
      "isoCode": "en-US",
      "imageDescription": "Item Image",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
  ]
}
],
"itemSeason": [
{
  "seasonId": 1,
  "phaseId": 1,
  "sequenceNo": 1,
  "diffId": null,
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"referenceItem": [
{
  "referenceItem": null,
  "primaryInd": "N",
  "formatId": null,
  "prefix": 1,
  "itemNoType": "MANL",
  "createDateTime": "2001-12-31T23:59:59.000Z",
  "updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"itemBOM": [
{
  "componentItem": null,
  "packQuantity": 1.0
}
]

```



```

    },
    ],
    "itemTicket": [
    {
        "ticketOverPercentage": 1.0,
        "printOnPcInd": "String",
        "ticketTypeId": null,
        "ticketTypeDescription": null,
        "shelfEdgeLabelInd": null,
        "poPrintType": null,
        "printOnPriceChangeInd": "Y",
        "ticketOverPrintPercentage": 3,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z",
        "details": [
        {
            "ticketItemId": null,
            "ticketItemIdDescription": null,
            "udaId": 1,
            "sequenceNo": 1
        }
        ],
    },
    ],
    "relatedItem": [
    {
        "relationshipId": 1,
        "relationshipName": null,
        "relationshipType": null,
        "mandatoryInd": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z",
        "translation": [
        {
            "languageCode": 2,
            "languageName": null,
            "isoCode": "en-US",
            "relationshipName": null,
            "createDateTime": "2001-12-31T23:59:59.000Z",
            "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
        ],
    },
    ],
    "details": [
    {
        "relatedItem": null,
        "priority": 1,
        "startDate": "2001-12-31",
        "endDate": "2001-12-31",
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
    ],
    ],
    "hts": [
    {

```

```

    "hts": null,
    "importCountry": "US",
    "originCountry": "US",
    "effectFrom": "2001-12-31",
    "effectTo": "2001-12-31",
    "clearingZoneId": null,
    "status": null,
    "createDateTime": "2001-12-31T23:59:59.000Z",
    "updateDateTime": "2001-12-31T23:59:59.000Z",
    "assessments": [
      {
        "componentId": null,
        "computationValueBase": null,
        "componentRate": 1.0,
        "perCount": 1.0,
        "perCountUom": null,
        "estimatedAssessmentValue": 1.0,
        "componentCurrency": null,
        "nominationFlag1": null,
        "inDuty": null,
        "nominationFlag3": null,
        "inExpense": null,
        "inAlc": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ],
    "expense": [
      {
        "supplier": 2400,
        "countryOfSourcing": "US",
        "ladingPort": "7",
        "dischargePort": "480",
        "costZoneGroupId": 1,
        "costZoneId": 1,
        "baseExpenseInd": "N",
        "componentId": "AGCOMM",
        "computationValueBase": "TDTYUS",
        "componentRate": 3.55,
        "perCount": 3,
        "perCountUom": "EA",
        "estimatedExpenseValue": 1.0,
        "componentCurrency": "USD",
        "nominationFlag1": null,
        "inDuty": null,
        "nominationFlag3": null,
        "inExpense": null,
        "inAlc": null,
        "createDateTime": "2001-12-31T23:59:59.000Z",
        "updateDateTime": "2001-12-31T23:59:59.000Z"
      }
    ],
    "requiredDocuments": [
      {

```

```

"documentId": 1
},
],
"itemImportAttributes": [
{
"importDescription": "import desc",
"commodity": "A123",
"tooling": 12,
"firstOrderIndicator": "Y",
"amortizeBase": 12.4568,
"openBalance": 12.4568,
"createDateTime": "2001-12-31T23:59:59.000Z"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_MASTER_CFA_EXT	Yes	No	No	No
ITEM_SUPPLIER_CFA_EXT	Yes	No	No	No
ITEM_SUPP_COUNTRY_CFA_EXT	Yes	No	No	No
ITEM_SUPP_COUNTRY_LOC_CFA_EXT	Yes	No	No	No
MERCHAPI_EXT_ITEM	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ITEM_JSON	Yes	No	No	No
ICL_RMS_ITEM_MASTER	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPPLIER	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPP_UOM	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPP_CTRY	Yes	No	Yes	Yes
ICL_RMS_ITEM_SUPP_CTRY_DIM	Yes	No	Yes	Yes

TABLE	SELECT	INSERT	UPDATE	DELETE
ICL_RMS_ITEM_SUPP_MANU_CTRY	Yes	No	Yes	Yes
ICL_RMS_ITEM_TICKET	Yes	No	Yes	Yes
ICL_RMS_UDA_ITEM_DATE	Yes	No	Yes	Yes
ICL_RMS_UDA_ITEM_LOV	Yes	No	Yes	Yes
ICL_RMS_UDA_ITEM_FF	Yes	No	Yes	Yes
ICL_RMS_ITEM_IMAGE	Yes	No	Yes	Yes
ICL_RMS_ITEM_SEASONS	Yes	No	Yes	Yes
ICL_RMS_RELATED_ITEM	No	Yes	No	No
ICL_RMS_ITEM HTS	Yes	No	Yes	Yes
ICL_RMS_ITEM_EXP	Yes	No	Yes	Yes

Get Item Location Details

Endpoints

MerchIntegrations/services/foundation/item/location

Functional Area

Items - Item Definition

Business Overview

This service publishes item/location relationships to downstream consuming systems. It covers item/location combinations whose source view still resolves to a store, warehouse, or external finisher and whose item is currently in status A or D. The payload carries the location-specific selling, sourcing, replenishment, VAT, substitute-item, and custom-flex details that downstream commerce and inventory integrations use to interpret item behavior at a location.

This API follows a broader merged-cache pattern than the simpler single-entity publish services. The external foundation/item/location endpoint shares its cache and runtime configuration with foundation/omnichannel/item/itemlocation, and the GET-all query switches between the delta and full access paths based on whether the requested window is 30 days or less or greater than 30 days.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ITEM_LOC, ITEM_LOC_TRAITS, REPL_ITEM_LOC, SUB_ITEMS_HEAD, SUB_ITEMS_DETAIL

Additional business tables: ITEM_MASTER, ITEM_SUPP_COUNTRY, ITEM_SUPP_COUNTRY_LOC, VAT_ITEM, RELATED_ITEM_HEAD, SYSTEM_OPTIONS, UOM_CLASS, STORE, WH, PARTNER, ITEM_LOC_CFA_EXT

JSON cache table: MERCHAPI_ITEM_LOC

JSON generation view: V_MERCHAPI_ITEM_LOC_JSON

Builds the payload from ITEM_LOC, optional traits from ITEM_LOC_TRAITS, replenishment attributes from REPL_ITEM_LOC, item hierarchy and search attributes from ITEM_MASTER, supplier/country and location cost fields from ITEM_SUPP_COUNTRY and ITEM_SUPP_COUNTRY_LOC, custom flex attributes from MERCHAPI_EXT_ITEM_LOC_CFA, VAT data from VAT_ITEM, related-item flags from RELATED_ITEM_HEAD, and location enrichment from a loc CTE over STORE, WH, and PARTNER where PARTNER_TYPE = 'E'.

The source view is limited to item/location combinations whose item is currently in status A or D and whose location resolves to a store, warehouse, or external finisher.

The published payload carries the location-specific selling, sourcing, replenishment, VAT, substitute-item, and custom-flex details that downstream commerce and inventory integrations use to interpret item behavior at a location.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_ITEM_LOC_JSON and apply ITEM_MASTER.STATUS IN ('A','D'); loc includes stores, warehouses, and partners with PARTNER.PARTNER_TYPE = 'E'.

Rebuild marks cache rows deleted when the item/location no longer exists in V_MERCHAPI_ITEM_LOC_JSON.

ICL (Integration Change Log) table: ICL_RMS_ITEM_LOC, ICL_RMS_REPL_ITEM_LOC (ICL consumer name = MERCHAPI_2)

ICL_RMS_ITEM_LOC is fed by triggers on ITEM_LOC, ITEM_LOC_TRAITS, ITEM_LOC_CFA_EXT, SUB_ITEMS_HEAD, SUB_ITEMS_DETAIL, and ownership-publish updates through OWNERSHIP_PUB_GTT.

ICL_RMS_REPL_ITEM_LOC rows are staged from REPL_ITEM_LOC and are republished as updates to the parent item-location payload.

Item Approval Handling

On item approval, the main item publish flow writes existing store and warehouse ITEM_LOC rows to ICL_RMS_ITEM_LOC so this API is also queued once the parent item becomes approved. ITEM_LOC rows are not backfilled through this path while the item remains in worksheet or submitted status.

Batch-driven bulk rebuild handling for specific events

Store VAT region change and warehouse VAT region change do not rely only on ICL staging. Those source events insert a record into MERCHAPI_ASYNC_REQUEST with REQUEST_TYPE = DATA_REFRESH, REFRESH_TYPE = REBUILD, and REQUEST_PARAMETERS set to the affected location so the rebuild stays scoped to that store or warehouse.

Tomorrow-effective VAT rate activation through VAT code-rate maintenance using batch job VATDLXPL and its prepost (post) can also insert a record into MERCHAPI_ASYNC_REQUEST with REQUEST_TYPE = DATA_REFRESH and REFRESH_TYPE = REBUILD when the VAT change can affect a broad item/location population. This broader rebuild path is handled through the Vat Rate Change Explode (VATDLXPL) flow as part of prepost.pc. Smaller tomorrow-effective item VAT changes are handled through targeted item/location delta staging instead of the broader rebuild path.

These requests are processed through MERCHAPI_DATA_REBUILD_ADHOC_PROCESS -> MERCHAPI_DATA_REBUILD_ADHOC_JOB and refresh the shared MERCHAPI_ITEM_LOC cache used by foundation/item/location and foundation/omnichannel/item/itemlocation.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_OMNI_ITEM_LOC

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_OMNI_ITEM_LOC_ADHOC_PROCESS ->
MERCHAPI_OMNI_ITEM_LOC_ADHOC_JOB

Webhook configuration api name: foundation/item/location

Special handling

Additional GET query parameters: item, locationType, location, itemLocationStatus, purchaseType, tranLevelInd, itemNumberType, departmentId, classId, subclassId, inventoried, sellable, orderable, changeType.

The filters itemLocationStatus, purchaseType, tranLevelInd, itemNumberType, departmentId, classId, subclassId, inventoried, sellable, and orderable are not backed by dedicated cacheable indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. Because this service builds a merged item-location cache from multiple base and supporting tables, the available configuration should be validated against the customer production item-location profile, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through API_NO_OF_THREADS, and commit sizing is controlled through API_COMMIT_MAX_COUNTER.

Thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the shared item-location cache from multiple base and supporting tables. Common starting points for large-volume (multi-millions) runs are 16 or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment.

API_COMMIT_MAX_COUNTER should be sized together with the thread setting. Typical values range from about 1,000 rows to 10,000.

For delta processing, the same parameters apply. It is advisable to reduce the thread values to 2 or 4 after the seeding is complete.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes. `GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a specific reason to change it.

REST fetch considerations

Item-location payloads can be comparatively large, so REST page size should be selected based on average message size so the average response remains below about 10 MB per call. Common page-limit values between 1,000 or 10,000 rows can be used.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves before empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 100,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using API specific `HISTOGRAM_ROW_COUNT` configuration.

Handling Bulk Publish on VAT Rate change event or Store/Warehouse attribute change

This API also uses batch-driven `REBUILD` handling for store or warehouse VAT-region changes and broader VAT rate events that can materially increase the publish population.

Because the VAT effect must be recalculated for all affected items at affected locations, those changes can expand into a much larger item/location republish set.

A larger VAT code-rate change can result in a very large number of records being published to downstream systems and should be planned with the downstream performance implication of that message volume in mind.

Full-refresh workload balancing and monitoring

This API uses a different full-refresh balancing pattern than the simpler fixed-thread publish services because item/location volume is not evenly distributed by location. A smaller store can contribute only a limited number of rows while a larger store or warehouse can contribute millions, so the refresh balances work by item/location workload rather than by location number alone.

As processing continues, `MERCHAPI_BOOKMARK` can show newly active entries for individual locations created from that workload balancing while earlier entries complete, and the rebuild remains in progress as long as at least one `MERCHAPI_BOOKMARK` row is still active.

Service Type

GET

ReST URL

`MerchIntegrations/services/foundation/item/location`

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
item	No	String	Item
locationType	No	String	Location Type - STORE, WAREHOUSE or EXTERNAL_FINISHES
location	No	String (10)	Location
itemLocationStatus	No	String	Item Location Status - A, I, C or D
purchaseType	No	String	Purchase Type - 0, 1 or 2
tranLevelInd	No	String	Transaction Level Indicator - Y or N
itemNumberType	No	String	Item Type
departmentId	No	String (4)	Department
classId	No	String (4)	Class
subclassId	No	String (4)	Subclass
inventoried	No	String	Inventoried - Y or N
sellable	No	String	Sellable - Y or N
orderable	No	String	Orderable - Y or N
offsetkey	No	String	Offset key ([location type],[location],[item]) Example: S,120091,Item-35)
limit	No	String (6)	Pagination limit. Default value is 1000.
changeType	No	String	Filter records that were created or deleted within since/ before time range. Valid values create or delete
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-462 PageResultsItemsMerchApiItemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-463 MerchApiItemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for item-location details.
item	No	String (25)	This field specifies the unique alphanumeric value that identifies the item.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemLevel	No	Number(1,0)	This field specifies Number indicating which of the three levels the item resides. The item level determines if the item stands alone or if it is part of a family of related items. The item level also determines how the item may be used throughout the system.
tranLevel	No	Number(1,0)	This field specifies Number indicating which of the three levels transactions occur for the item's group. The transaction level is the level at which the item's inventory is tracked in the system. The transaction level item will be counted, transferred, shipped, etc. The transaction level may be at the current item or up to 2 levels above or below the current item. Only one level of the hierarchy of an item family may contain transaction level items.
location	No	Number(10,0)	This field contains the numeric identifier of the location in which the item is to be found. This field may contain a store, warehouse, or external finisher.
locationType	No	String (1)	This field contains the type of location in which the item is to be found. Valid values are Store (S), Warehouse (W), and External Finisher (E).

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	No	String (1)	Current status of item at the store or warehouse. Valid values are: A = Active, item is valid and can be ordered and sold; I = Inactive, item is valid but cannot be ordered or sold; C = Discontinued, item is valid and sellable but no longer orderable; D = Delete, item is invalid and cannot be ordered or sold.
physicalWarehouse	No	Number(10,0)	This field contains the numeric identifier of the physical warehouse in which the item is to be found.
storeType	No	String (6)	This field indicates whether a particular store is a franchise or company store.
stockholdingInd	No	String (1)	This field indicates whether or not the location can hold stock.
localItemDescription	No	String (250)	This field specifies selling local description of the item. This field will default to the item's description but will be over-ridable. This value will be downloaded to the POS.
localShortDescription	No	String (120)	Contains the local short description of the item. This field will default to the item's short description but will be over-ridable. This value will be downloaded to the POS.
primarySupplier	No	Number(10,0)	This field contains the primary supplier of the item at the location.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryCountry	No	String (3)	This field contains the primary country of the item at the location.
unitCost	No	Number(20,4)	This field is the current unit cost of the item for the item/supplier/origin_country combination. This field is stored in the supplier's currency.
costUom	No	String (4)	A cost UOM is held to allow costs to be managed in a separate UOM than the standard UOM.
purchaseType	No	Number(1,0)	This field contains a code to indicate whether the item is normal merchandise (i.e. owned by the retailer), consignment stock, or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This field indicates if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number(12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 5-463 (Cont.) MerchApitemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pricingControl	No	String (1)	This field contains the pricing control of the item which will determine whether the item price is controlled by the retailer or supplier. Valid values are R - Managed by Retailer, S - Managed by Supplier, NULL - For Owned Items.
receiveAsType	No	String (1)	This column determines whether the stock on hand for a pack component item or the pack itself will be updated when a pack is received at a warehouse. Valid values are Each (E) or Pack (P). This value is required for all pack items where the location type is Warehouse (W). For vendor packs, this must be set to Pack (P); for buyer packs, it can be either Each (E) or Pack (P).
taxableInd	No	String (1)	This field indicates whether or not the item is taxable at the store. Valid values are: Y = Yes, the item is taxable; N = No, the item is not taxable.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sourceMethod	No	String (1)	This field will be used to specify how the ad-hoc PO/TSF creation process should source the item/location request. If the value is Warehouse, the process will attempt to fill the request by creating a transfer from the warehouse mentioned in the source_wh field. If this warehouse does not have enough inventory to fill the request, a purchase order will be created for the item/location's primary supplier. For warehouses, it is used by Oracle Retail Allocation to determine the valid sources and destinations for warehouse-to-warehouse allocations.
sourceWarehouse	No	Number(10,0)	This field will be used by the ad-hoc PO/Transfer creation process to determine which warehouse to fill the store's request from. It will also be used by the Allocation process to support warehouse-to-warehouse allocations. A value will be required in this field if the sourcing method is Warehouse.
unitRetail	No	Number(20,4)	This field contains the unit retail price in the standard unit of measure for the item/location combination. This field is stored in the local currency.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sellingUnitRetail	No	Number(20,4)	This field contains the unit retail price in the selling unit of measure for the item/location combination. This field is stored in the local currency.
sellingUom	No	String (4)	This field contains the selling unit of measure for an item's single-unit retail.
storePriceInd	No	String (1)	This field indicates if an item at a particular store location can have the unit retail marked down by the store.
uinType	No	String (6)	This field contains the unique identification number (UIN) used to identify the instances of the item at the location.
uinLabel	No	String (6)	This field contains the label for the UIN when displayed in the store system.
captureTime	No	String (6)	This field indicates when the UIN should be captured for an item during transaction processing.
externalUinInd	No	String (1)	This Yes/No indicator indicates if UIN is being generated in the external system.
rangedInd	No	String (1)	This field determines if the location is ranged intentionally by the user for replenishment/selling or incidentally ranged by the Merchandising programs when the item is not ranged to a specific location on the transaction.
itemParent	No	String (25)	This field contains the alphanumeric value that uniquely identifies the item/group at the level above the item.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemGrandparent	No	String (25)	This field contains the alphanumeric value that uniquely identifies the item/ group two levels above the item.
ti	No	Number(12,4)	This field contains the number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get the total number of cases for a pallet.
hi	No	Number(12,4)	This field contains the number of tiers that make up a complete pallet (height). Multiply TI x HI to get the total number of cases for a pallet.
storeOrderMultiple	No	String (1)	This field contains the multiple in which the item needs to be shipped from a warehouse to the location.
dailyWastePercent	No	Number(12,4)	This field contains the average percentage lost from inventory on a daily basis due to natural wastage.
measureOfEach	No	Number(12,4)	This field contains the size of an each in terms of the uom_of_price. For example, 12 oz. Used in ticketing.
measureOfPrice	No	Number(12,4)	This field contains the size to be used on the ticket in terms of the uom_of_price. For example, if the user wants the ticket to have the label print the price per ounce, this value would be 1. If the user wanted the price per 100 grams, this value would be 100. This is used in ticketing.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uomOfPrice	No	String (4)	This field contains the unit of measure that will be used on the ticket for this item.
primaryVariant	No	String (25)	This field is used to address sales of PLUs (i.e. above transaction level items) when inventory is tracked at a lower level (i.e. UPC). This field will only contain a value for items one level higher than the transaction level. Valid choices will be any transaction-level item that is a child of this item. In order to select a transaction-level item as the primary variant, the item/location relationship must exist at the transaction level. Both the transaction-level item (i.e. UPC) and the higher-than-transaction-level item (i.e. PLU) will be sent to the POS to allow the store to sell the PLU. The information sent for the PLU will be the same information sent for the transaction-level item (i.e. UPC).
primaryCostPack	No	String (25)	This field contains an item number that is a simple pack containing the item in the item column for this record. If populated, the cost of the future cost table will be driven from the simple pack and the deals and cost changes for the simple pack.
inboundHandlingDays	No	Number(2,0)	This field indicates the number of inbound handling days for an item at a warehouse type location.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
regularUnitRetail	No	Number(20,4)	This field holds the unit retail in the standard unit of measure for the item/location (zone) combination. This field is stored in the local currency.
multiUnits	No	Number(12,4)	This field contains the multi-units for the item/location (zone) combination.
multiUnitRetail	No	Number(20,4)	This field holds the multi-unit retail in the multi-selling unit of measure for the item/location (zone) combination. This field is stored in the local currency.
multiSellingUom	No	String (4)	This field holds the selling unit of measure for this item/location (zone) combination's multi-unit retail.
clearanceInd	No	String (1)	Indicates if item is on clearance at the store. Valid values are: Y = Yes, the item is on clearance; N = No, the item is not on clearance.
costingLocation	No	Number(10,0)	This field contains the identifier of the costing location for the franchise store. This field may contain a store or a warehouse.
costingLocationType	No	String (1)	This field holds the type of costing location in the costing location field.
currencyCode	No	String (3)	This field contains the currency code under which the location operates.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pickupLeadTime	No	Number(4,0)	This field contains the time it takes to get the item from the Supplier to the Initial Receiving Location. This value will be defaulted to the item_supp_country_loc pickup lead time field. The ordering dialog will reference the item/supplier/country/location pickup lead time as the value may vary by location.
rfidInd	No	String (1)	This field allows the retailer to specify if the item should be RFID tagged or not.
primaryReplenishmentSupplier	No	Number(10,0)	This field contains the numeric identifier of the supplier from which the specified location will source the replenishment demand for the specified item location. Used for items on Direct to Store/ Warehouse or Cross-docked replenishment.
replenishmentMethod	No	String (6)	This field contains the character code for the algorithm that will be used to calculate the recommended order quantity for the item location. Valid values include Constant, Min/Max, Floating point, Time Supply, Dynamic, SO Store Orders.
rejectStoreOrderInd	No	String (1)	This field contains an indicator that determines if uploaded store orders should be rejected. If the indicator is N, then store orders for all need dates are valid. If Y, store orders with needs date on or after the NEXT_DELIVERY_DATE are valid.

Table 5-463 (Cont.) MerchApitemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nextDeliveryDate	No	date	This field contains the next delivery date calculated for the next review cycle. If, for the next review cycle, an order will be created that is calculated to arrive on Thursday, this field will be updated with Thursday's date.
multipleReplenishmentsPerDayInd	No	String (1)	This field indicates if an item can be replenished multiple times per day at the location. Default value is N.
launchDate	No	date	This field holds the date that the item should first be sold at the location. This value will be downloaded to the POS.
quantityKeyOptions	No	String (6)	This field determines whether the qty key on a POS should be used for this item at the location. Valid values are in the code_type RPO. Current values include R - required, P - Prohibited, and O - Optional. Additional values can be added to the code type as needed. This value will be downloaded to the POS as needed.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manualPriceEntry	No	String (6)	Determines whether the price can/should be entered manually on a POS for this item at the location. Valid values are in the code_type RPO. Current values include R - required, P - Prohibited, and O - Optional. Additional values can be added or removed as needed. This value will be downloaded to the POS.
depositCode	No	String (6)	This field indicates whether a deposit is associated with this item at the location. Valid values are in the code_type DEPO. Additional values may be added or removed as needed. The only processing in Merchandising involving the deposit code is downloading it to the POS. Deposits are not subtracted from the retail of an item uploaded to Merchandising, etc. This kind of processing is the responsibility of the client and should occur before sales are sent to any Oracle Retail application.
foodStampInd	No	String (1)	This field indicates whether the item is approved for food stamps at the location. This value will be downloaded to the POS.
wicInd	No	String (1)	This field indicates whether the item is approved for WIC at the location. This value will be downloaded to the POS.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
proportionalTarePercent	No	Number(12,4)	This field holds the value associated with the packaging in items sold by weight at the location. The proportional tare is the proportion of the total weight of a unit of an item that is packaging (i.e. if the tare item is bulk candy, this is the proportion of the total weight of one piece of candy that is the candy wrapper). The only processing Merchandising does involving the proportional tare percent is downloading it to the POS. Proportional tare is not subtracted from items sold by weight when sales are uploaded to Merchandising, etc. This kind of processing is the responsibility of the client and should occur before sales are sent to any Oracle Retail application.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fixedTareValue	No	Number(12,4)	This field holds the value associated with the packaging in items sold by weight at the location. Fixed tare is the tare of the packaging used (i.e. if the tare item is bulk candy, this is the weight of the bag and twist tie). The only processing Merchandising does involving the fixed tare value is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to Merchandising, etc. This kind of processing is the responsibility of the client and should occur before sales are sent to any Oracle Retail application.
fixedTareUom	No	String (4)	This field holds the unit of measure value associated with the tare value. The only processing Merchandising does involving the fixed tare value and UOM is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to Merchandising, etc. This kind of processing is the responsibility of the client and should occur before sales are sent to any Oracle Retail application.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rewardEligibleInd	No	String (1)	This field holds whether the item is legally valid for various types of bonus point/award programs at the location. This value will be downloaded to the POS.
nationalBrandCompetitorItem	No	String (25)	This field holds the nationally branded item to which you would like to compare the current item. This nationally branded item must exist as an item in Merchandising (i.e. on the item_master table).
returnPolicy	No	String (6)	This field contains the return policy for the item at the location.
stopSaleInd	No	String (1)	This field indicates that sale of the item should be stopped immediately at the location (i.e. in case of recall etc). This value will be downloaded to the POS.
electronicMarketClubs	No	String (6)	This field holds the code that represents the marketing clubs to which the item belongs at the location. Valid values can belong to the code_type MTKC. Additional values can be added or removed from the code type as needed. This value will be downloaded to the POS.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reportCode	No	String (6)	This field contains the code to determine which reports the location should run. Valid values are defined in the code type REPC. Values may be added and removed as necessary. This field is not required. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
requiredShelfLifeOnSelection	No	Number(4,0)	This field holds the required shelf life for an item on selection in days. This field is not required. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
requiredShelfLifeOnReceipt	No	Number(4,0)	This field holds the required shelf life for an item on receipt in days. This field is not required. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeReorderableInd	No	String (1)	This field indicates whether the store may re-order the item. This field is required to be either Y - yes or N - no. The field will default to N. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
rackSize	No	String (6)	This field indicates the rack size that should be used for the item. This field is not required. Valid values for the field can be found and defined in the code_type RACK. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
fullPalletItem	No	String (1)	This field indicates whether a store must reorder an item in full pallets only. This field is required to be either Y - yes or N - no. The field will default to N. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inStoreMarketBasket	No	String (6)	This field holds the in store market basket code for this item/ location combination. Valid values for the field can be found in the code_type STMB. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
storageLocation	No	String (7)	This field holds the current storage location or bin number for the item at the location. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
alternateStorageLocation	No	String (7)	This field holds the preferred alternate storage location or bin number for the item at the location. No Merchandising processing is based on the value in this field. It is the client's responsibility to communicate this value to external systems that may need it for processing.
returnableInd	No	String (1)	This field contains a value of Yes when the item can be returned to the location.
refundableInd	No	String (1)	This field contains a value of Yes when the item is refundable at the location.

Table 5-463 (Cont.) MerchApiltemLoc - Object. See list of elements for detail

Element Name	Required	Data Type	Description
backOrderInd	No	String (1)	This field contains a value of Yes when the item can be back ordered to the location.
promotableInd	No	String (1)	This attribute, which has valid values of Y or N, determines whether or not the item can be promoted. This flag will be sent to the selling solutions (e.g. POS, OMS) to help their pricing engines know whether or not to include this item location in the items eligible for a specific promotion. This new attribute is intended to be used for item-location combinations that are never promoted so that users setting up promotions do not need to specify them as exclusions on the promotion.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the item-location.
cacheTimestamp	No	dateTime	This field specifies the date and time when the item-location record was last maintained.
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication.

Table 5-464 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-465 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "item": "100100076",
      "itemLevel": 1,
      "tranLevel": 1,
      "location": 1,
      "locationType": null,
      "status": null,
      "physicalWarehouse": 1,
      "storeType": null,
      "stockholdingInd": null,

```

```
"localItemDescription": "localItemDesc",
"localShortDescription": "localShortDesc",
"primarySupplier": 2400,
"primaryCountry": null,
"unitCost": 9.65,
"costUom": "EA",
"purchaseType": 1,
"calculationBasis": null,
"purchaseRate": 3,
"pricingControl": "R",
"receiveAsType": null,
"taxableInd": "Y",
"sourceMethod": null,
"sourceWarehouse": 1,
"unitRetail": 1.0,
"sellingUnitRetail": 1.0,
"sellingUom": null,
"storePriceInd": null,
"uinType": null,
"uinLabel": null,
"captureTime": null,
"externalUinInd": "Y",
"rangedInd": "Y",
"itemParent": null,
"itemGrandparent": null,
"ti": 3,
"hi": 3,
"storeOrderMultiple": null,
"dailyWastePercent": 3,
"measureOfEach": 2,
"measureOfPrice": 1,
"uomOfPrice": "EA",
"primaryVariant": "100100090",
"primaryCostPack": "100100080",
"inboundHandlingDays": 1,
"regularUnitRetail": 1.0,
"multiUnits": 1.0,
"multiUnitRetail": 1.0,
"multiSellingUom": null,
"clearanceInd": null,
"costingLocation": 199,
"costingLocationType": null,
"currencyCode": null,
"pickupLeadTime": 1,
"rfidInd": "Y",
"primaryReplenishmentSupplier": 1,
"replenishmentMethod": null,
"rejectStoreOrderInd": null,
"nextDeliveryDate": "2001-12-31",
"multipleReplenishmentsPerDayInd": null,
"launchDate": "2001-12-31",
"quantityKeyOptions": null,
"manualPriceEntry": null,
"depositCode": null,
"foodStampInd": "Y",
"wicInd": "Y",
```

```
"proportionalTarePercent": 3,
"fixedTareValue": 3,
"fixedTareUom": "EA",
"rewardEligibleInd": "Y",
"nationalBrandCompetitorItem": null,
"returnPolicy": null,
"stopSaleInd": "Y",
"electronicMarketClubs": null,
"reportCode": null,
"requiredShelfLifeOnSelection": 1,
"requiredShelfLifeOnReceipt": 1,
"storeReorderableInd": "Y",
"rackSize": null,
"fullPalletItem": null,
"inStoreMarketBasket": null,
"storageLocation": null,
"alternateStorageLocation": null,
"returnableInd": "Y",
"refundableInd": "Y",
"backOrderInd": "Y",
"promotableInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
{
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}
]
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_LOC_TRAITS	Yes	No	No	No
ITEM_SUPP_COUNTRY_LOC	Yes	No	No	No
REPL_ITEM_LOC	Yes	No	No	No
MERCHAPI_ITEM_LOC	Yes	Yes	Yes	Yes
V_MERCHAPI_ITEM_LOC_JSON	Yes	No	No	No
ICL_RMS_ITEM_LOC	Yes	No	Yes	Yes
ICL_RMS_REPL_ITEM_LOC	Yes	No	Yes	Yes
MERCHAPI_EXT_ITEM_LOC_CFA	Yes	Yes	No	Yes

Get Item Location Basic Details

Functional Area

Items

Business Overview

Retrieves basic item-location details for the specified items and locations. Provide a comma-separated list of up to 10 item IDs and a comma-separated list of up to 10 location IDs. The service returns the matching records from ITEM_LOC table. If no matching item-location combinations are found, the request succeeds and the response contains no records.

Service Type

GET

ReST URL

RmsReSTServices/services/private/Item/itemLocationDetail

Input Parameters

Parameter Name	Required	Data Type	Description
items	Yes	String	Up to 10 comma-separated item IDs. No whitespace.
locations	Yes	String	Up to 10 comma-separated numeric location IDs. No whitespace.

Response Code: 200 (Success)

Table 5-466 ItemLocation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Unique alphanumeric value that identifies the item.
location	No	Number(10,0)	Numeric identifier of the location where the item is ranged. The location can be a store, warehouse, or external finisher.
locationType	No	String (1)	Type of location in the location field. Valid values are S (Store), W (Warehouse), and E (External Finisher).
status	No	String (1)	Current status of the item at the location. Valid values are A (Active), I (Inactive), C (Discontinued), and D (Deleted).
unitRetail	No	Number(20,4)	Unit retail price for the item-location combination in the standard unit of measure. The value is stored in the local currency.
sellingUnitRetail	No	Number(20,4)	Unit retail price for the item-location combination in the selling unit of measure. The value is stored in the local currency.
sellingUom	No	String (4)	Selling unit of measure used for the item's single-unit retail.
clearanceInd	No	String (1)	Indicates whether the item is on clearance at the location. Valid values are Y (Yes) and N (No).
localItemDescription	No	String (250)	Local description of the item. This value defaults to the item description and can be overridden for the location.

Sample Response Message

```
[
{
  "item": "211001211",
  "location": 10000089,
  "locationType": "S",
  "status": "A",
  "unitRetail": 19.95,
  "sellingUnitRetail": 19.95,
  "sellingUom": "EA",
  "clearanceInd": "N",
  "localItemDescription": "Graphic T-shirt - Red"
}
]
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Get Item VAT Details**Endpoints**

MerchIntegrations/services/foundation/item/vat

MerchIntegrations/services/foundation/item/vat/{item}

Functional Area

Items - Item Definition

Business Overview

This service publishes item VAT data to downstream consuming systems. The payload includes the item, VAT region, active date, VAT type, VAT code, VAT rate, reverse-VAT indicator, and audit timestamps. Functionally, the JSON view publishes only approved transaction-level items because it requires ITEM_MASTER.STATUS = 'A' and ITEM_MASTER.ITEM_LEVEL = ITEM_MASTER.TRAN_LEVEL.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: VAT_ITEM, ITEM_MASTER

Additional business tables: none

JSON cache table: MERCHAPI_EXT_ITEM_VAT

JSON generation view: V_MERCHAPI_EXT_ITEM_VAT_JSON

The JSON view is built from VAT_ITEM joined to ITEM_MASTER, and it publishes the item key, VAT region, active date, VAT type, VAT code, VAT rate, reverse-VAT indicator, and audit timestamps.

The view only publishes rows where the item is approved and transaction-level because it requires ITEM_MASTER.STATUS = 'A' and ITEM_MASTER.ITEM_LEVEL = ITEM_MASTER.TRAN_LEVEL.

Initial refresh supports both truncate-and-load and rebuild behavior, and rebuild marks cache rows deleted when an item no longer resolves in the JSON view.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_ITEM_VAT_JSON and apply ITEM_MASTER.STATUS = 'A' and ITEM_MASTER.ITEM_LEVEL = ITEM_MASTER.TRAN_LEVEL.

Mark cache rows deleted when the item no longer exists in V_MERCHAPI_EXT_ITEM_VAT_JSON for the current thread.

ICL (Integration Change Log) table: ICL_RMS_VAT_ITEM (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on VAT_ITEM, ITEM_MASTER.

Item Approval Handling

On item approval, the main item publish flow writes existing VAT_ITEM rows for the item to ICL_RMS_VAT_ITEM so this API is also queued when the item becomes approved. VAT rows are not staged through this path while the item remains in worksheet or submitted status.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ITEM_VAT

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ITEM_VAT_ADHOC_PROCESS ->
MERCHAPI_EXT_ITEM_VAT_ADHOC_JOB

Webhook configuration api name: foundation/item/vat

Special handling

Additional GET query parameters: item, vatRegion, activeDate, vatType.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. The available configuration for this service should therefore be reviewed and validated against the customer production data profile, including expected volume, message size, and access pattern, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for initial seeding, while delta processing remains single-threaded. Initial-seeding thread count is controlled through `API_REFRESH_OVERRIDE_NO_OF_THREADS`, and initial-seeding commit sizing is controlled through `API_REFRESH_OVERRIDE_COMMIT_MAX_COUNTER`.

For initial seeding, thread and commit settings should be chosen together based on expected row volume, average message size, and the simpler `VAT_ITEM` and `ITEM_MASTER` join used to build the payload. Common starting points for large-volume seeding are 8, 16, or 32 threads. The delivered refresh override is currently 4 threads.

Refresh commit sizing should be reviewed together with the thread setting. Typical values can range from 1,000 rows or higher values such as 10,000. The delivered refresh override commit setting is currently 5000.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. Even with refresh-override threading, larger parallel insert volumes can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes.

This can improve data-load throughput, but the overall refresh plan should also account for the additional time needed for index rebuild after the seeding step finishes.

`GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a need to change it.

REST fetch considerations

Because the payload is relatively narrow, larger page-limit values are often practical after testing. REST page size should still be selected so that the average response remains below about 10 MB per call, using common page-limit values such as 1,000, 5,000, or 10,000 rows.

For full downloads, do not provide the `since` query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned `before` timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

`MerchIntegrations/services/foundation/item/vat`

`MerchIntegrations/services/foundation/item/vat/{item}`

Input Parameters

Parameter Name	Required	Data Type	Description
<code>since</code>	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
<code>before</code>	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
<code>item</code>	No	String	Item
<code>vatRegion</code>	No	String (4)	VAT Region
<code>activeDate</code>	No	String	Active Date format required: YYYY-MM-DD
<code>vatType</code>	No	String	VAT Type
<code>offsetkey</code>	No	String	Offset key ([item],[vat region],[active date],[vat type] Example: Cross_1001,1000,2020-10-22,R)
<code>limit</code>	No	String (6)	Pagination limit. Default value is 1000.
<code>optimalBefore</code>	No	String	Enables histogram-based calculation of the <code>before</code> timestamp when <code>before</code> is not provided. Valid values are Y or N. Default value is N.
<code>include</code>	No	String	Fields to include - comma separated (Example items.name,items.address.city)
<code>exclude</code>	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-467 Input Parameter for "MerchIntegrations/services/foundation/item/vat/{item}"

Parameter Name	Required	Data Type	Description
item	Yes	String	
offsetkey	No	String	Offset key ([item],[vat region],[active date],[vat type] Example: Cross_1001,1000,2020-10-22,R)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-468 PageResultsItemsMerchApiItemVat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-469 MerchApiltemVat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for item VAT details.
item	No	String (25)	This field contains the unique number that identifies an item.
vatRegion	No	Number(4,0)	This field contains the number of the Value Added Tax region a location belongs to.
activeDate	No	date	This field contains the date that the VAT rate became active for the item/VAT region combination.
vatType	No	String (1)	This field indicates the type of VAT. Valid values include R (Retail), C (Cost), B (Both).
vatCode	No	String (6)	This field contains the alphanumeric identification for the VAT code.
vatRate	No	Number(20,10)	This field contains the VAT rate for the item/ location.
reverseVatInd	No	String (1)	This field indicates if the item is subject to reverse charge VAT at the VAT region. Valid values are Y and N.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the item VAT record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-470 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "item": "100100076",
      "vatRegion": 1000,
      "activeDate": "2001-12-31",
      "vatType": "R",
      "vatCode": "S",
      "vatRate": 1.0,
      "reverseVatInd": "N",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
VAT_ITEM	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
MERCHAPI_EXT_ITEM_VAT	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ITEM_VAT_JSON	Yes	No	No	No
ICL_RMS_VAT_ITEM	Yes	No	Yes	Yes

Get Item VAT Details by Item Number

Endpoints

MerchIntegrations/services/foundation/item/vat
 MerchIntegrations/services/foundation/item/vat/{item}

Functional Area

Items

Business Overview

This service can be used by external applications to retrieve VAT information for all or selected items from Merchandising.

Service Type

GET

ReST URL

/MerchIntegrations/services/foundation/vat/item?
 item={item}&vatRegion={vatRegion}&activeDate={activeDate}&vatType={vatType}&of
 fsetkey={offsetkey}&limit={limit}&since={since}&before={before}
 /MerchIntegrations/services/foundation/item/vat/{item}

Input Parameters

Table 5-471 /MerchIntegrations/services/foundation/item/vat

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
item	No	String	Item Number.
vatRegion	No	Number	Vat Region.

Table 5-471 (Cont.) /MerchIntegrations/services/foundation/item/vat

Parameter Name	Required	Data Type	Description
activeDate	No	String	Active date for item VAT on future cost. Format: YYYY-MM-DD
vatType	No	String	VAT Type. Valid values are R - Retail, C - Cost, B - Both.
offsetkey	No	String	Offset Key. Valid Value is an Item ID.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.

Table 5-472 /MerchIntegrations/services/foundation/item/vat/{item}

Parameter Name	Required	Data Type	Description
item	Yes	Number	Item Number.

Output

Name	Data Type	Description
action	String	Action for Item VAT details. Valid Values are INSERT, UPDATE and DELETE.
item	String	This field holds the ID of the item.
vatRegion	Number	This field contains the number of the Value Added Tax region a location belongs to.
activeDate	String	This field holds the active date for record on future cost.
vatType	String	This field indicates the type of VAT. Valid values include R (Retail), C (Cost), B (Both).
vatCode	String	This field contains the alphanumeric identification for the VAT code.
vatRate	Number	This field contains the VAT rate for the item-location.
reverseVatInd	String	This field contains the VAT rate for the item-location.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when the item VAT record was last maintained.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.

Elements	Description
hasMore	This is dependent on limit value. If the total rows are more than the limit defined, then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists, then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self": This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON input:

```
{
  "items": [
    {
      "action": null,
      "item": "104900054",
      "vatRegion": 1000,
      "activeDate": "2021-10-01",
      "vatType": "B",
      "vatCode": "S",
      "vatRate": 10,
      "reverseVatInd": "N",
      "createDateTime": "2022-03-29T10:07:45.000Z",
      "updateDateTime": "2022-03-29T10:07:45.000Z",
      "cacheTimestamp": "2022-03-29T12:32:58.009Z"
    }
  ],
  "hasMore": false,
  "limit": 10000,
  "count": 1,
  "links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
VAT_ITEM	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_EXT_ITEM_VAT	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ITEM_VAT_JSON	Yes	No	No	No
ICL_RMS_VAT_ITEM	Yes	No	Yes	Yes

Item ELC Upload Service

The following services are included in this functional area:

- [Create Item Conditional Tariff Treatments](#)
- [Create Item Expenses](#)
- [Create Item HTS](#)
- [Create Item HTS Assessments](#)
- [Delete Item Conditional Tariff Treatments](#)
- [Delete Item Expenses](#)
- [Delete Item HTS](#)
- [Delete Item HTS Assessments](#)
- [Update Item Expenses](#)
- [Update Item HTS](#)
- [Update Item HTS Assessments](#)

Create Item Conditional Tariff Treatments

Functional Area

Items - Item Definition

Business Overview

Tariff Treatments, also known as Special Programs, are used to determine the applicable duty to be assessed when importing goods into a particular country. Tariff treatments may or may not be conditional. A conditional tariff treatment indicates that only certain types of goods will be eligible for the special rates. When import items are defined in the system, the user must indicate for which conditional tariff treatments the item is eligible.

This service allows the user to add one or more conditional tariff treatments to an item that is at or above the transaction level, including vendor packs.

Recalculate Orders

- The Recalculate Orders parameter can be passed in as Yes (Y) or No (N) when a conditional tariff treatment is being added to or removed from an item. Adding and removing conditional tariff treatments to or from an item changes the item's eligibility for the tariff treatment which may result in a different duty rate. This parameter provides a way to indicate whether or not open orders should be recalculated.
- When Recalculate Orders is passed in as Yes (Y) on a 'Create' or 'Delete', after successful creation or deletion of the item conditional tariff treatment record, during the nightly batch run, the system searches for and recalculates all the order/item/HTS/assessments for the item followed by all expenses for all locations for the same item in order to update the value of any expenses that are based on assessment values, where the order/item or order/item/location has no receipts and the order is not closed.

- When Recalculate Orders is passed in as Yes (Y) on a parent item, then all child items will be queued for order/item recalculation.

Service Type

POST

ReST URL

MerchIntegrations/services/item/conditionalTariffTreatments/create

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-473 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-474 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
conditionalTariffTreatment	No	Collection of Object	Child node.

Table 5-475 ConditionalTariffTreatment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
tariffTreatment	Yes	String (10)	The unique identifier for the Conditional Tariff Program code. This refers to the tariff_treatment field from tariff_treatment table which are conditional. Example values are - CA (NAFTA for Canada CA), A+ (Generalized System of Preferences A+).
recalculateOrderInd	No	String (1)	The optional field indicates if open orders associated with the item should have assessments and expenses recalculated.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "conditionalTariffTreatment": [
        {
          "tariffTreatment": "CA",
          "recalculateOrderInd": "Y"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item Expenses

Functional Area

Items - Item Definition

Business Overview

This service is used to add expenses to an existing item in Merchandising. Expenses can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/expenses/create

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-476 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-477 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
expense	No	Collection of Object	References a collection of item expenses.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-478 Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	The unique identifier for the supplier.
componentId	Yes	String (10)	The code representing the specific expense component.
dischargePort	Yes	String (5)	The port where the item will enter the importing country.
originCountry	No	String (3)	The country where the item was manufactured or significantly altered.
ladingPort	No	String (5)	The final destination lading port of the item if the Expense Type is Country.
costZoneId	No	Number (10)	The final destination zone of the item if the Expense Type is Zone.

Table 5-478 (Cont.) Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
costZoneGroupId	No	Number (4)	This field holds the items Zone Group, it is stored here for performance reasons.
baseExpenseInd	No	String (1)	Indicates whether or not this Set of Expenses is also the base Set of Expenses.
computationValueBase	No	String (10)	The Computation Value Base representing the base value for calculating the expense.
componentRate	No	Number (20,4)	The rate to be charged against the calculation base. This field will hold a percentage value if the Calculation basis of the Component is Value or a monetary amount if the Calculation Basis is Specific.
perCount	No	Number (12,4)	A count indicating the amount of the Per Count Unit of Measure to which the rate applies. This column is only used for specific (per-unit) expenses.
perCountUom	No	String (4)	The unit of measure in which the Per Count is specified. This column is only used from specific (per-unit) expenses.
componentCurrency	No	String (3)	Contains the currency the expense is to be entered in. If the Calculation Basis is V (Value), then the comp_currency will be NULL.
updateOrdersInd	No	String (1)	Indicates that new changes will be reflected to associated order-location expense record.
nominationFlag1	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.

Table 5-478 (Cont.) Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inDuty	No	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.
nominationFlag3	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	No	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.
inAlc	No	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Table 5-479 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-480 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "expense": [
        {
          "supplier": 2400,
          "componentId": "AGCOMM",
          "dischargePort": "480",
          "originCountry": "US",
          "ladingPort": "7",
          "costZoneId": 696323,
          "costZoneGroupId": 1000,
          "baseExpenseInd": "N",
          "computationValueBase": null,
          "componentRate": 3.55,
          "perCount": 1,
          "perCountUom": "EA",
          "componentCurrency": "USD",
          "updateOrdersInd": "Y",
          "nominationFlag1": "N",
          "inDuty": "N",
          "nominationFlag3": "N",
          "inExpense": "+",
          "inAlc": "N"
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}

```

```
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item HTS

Functional Area

Items - Item Definition

Business Overview

This service is used to add HTS and Assessments to an existing item in Merchandising. HTS and Assessments can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

`MerchIntegrations/services/item/hts/create`

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Hts - Object. See list of elements for detail

Assessments - Object. See list of elements for detail

LocalizationExtensions - Object. See list of elements for detail

Attributes - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Create Item HTS Assessments

Functional Area

Items - Item Definition

Business Overview

This service is used to add HTS Assessments to an existing item in Merchandising. HTS Assessments can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/hts/assessments/create

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Hts - Object. See list of elements for detail

Assessments - Object. See list of elements for detail

LocalizationExtensions - Object. See list of elements for detail

Attributes - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Delete Item Conditional Tariff Treatments

Functional Area

Items - Item Definition

Business Overview

Tariff Treatments, also known as Special Programs, are used to determine the applicable duty to be assessed when importing goods into a particular country. Tariff treatments may or may

not be conditional. A conditional tariff treatment indicates that only certain types of goods will be eligible for the special rates. When import items are defined in the system, the user must indicate for which conditional tariff treatments the item is eligible.

This service will allow the user to add one or more conditional tariff treatments to an item that is at or above the transaction level, including vendor packs.

Recalculate Orders

- The Recalculate Orders parameter can be passed in as Yes (Y) or No (N) when a conditional tariff treatment is being added to or removed from an item. Adding and removing conditional tariff treatments to or from an item changes the item's eligibility for the tariff treatment which may result in a different duty rate. This parameter provides a way to indicate whether or not open orders should be recalculated.
- When Recalculate Orders is passed in as Yes (Y) on a 'Create' or 'Delete', after successful creation or deletion of the item conditional tariff treatment record, during the nightly batch run, the system will search for and recalculate all the order/item/HTS/assessments for the item followed by all expenses for all locations for the same item in order to update the value of any expenses that are based on assessment values, where the order/item or order/item/location has no receipts and the order is not closed.
- When Recalculate Orders is passed in as Yes (Y) on a parent item, then all child items will be queued for order/item recalculation.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/conditionalTariffTreatments/delete

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-481 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-482 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
conditionalTariffTreatment	No	Collection of Object	Child node.

Table 5-483 ConditionalTariffTreatment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
tariffTreatment	Yes	String (10)	The code representing the Conditional Tariff Program code to be deleted. This refers to the tariff_treatment field from tariff_treatment table which are conditional. Example values are - CA (NAFTA for Canada CA), A+ (Generalized System of Preferences A+).
recalculateOrderInd	No	String (1)	The optional field indicates if open orders associated with the item should have assessments and expenses recalculated.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "conditionalTariffTreatment": [
        {
          "tariffTreatment": "CA",
          "recalculateOrderInd": "Y"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
```

```

        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
    }
  ]
}

```

Delete Item Expenses

Functional Area

Items - Item Definition

Business Overview

This service is used to delete any expenses that are associated with an item in Merchandising. In order to perform this deletion, the item/supplier/component ID/discharge port must exist for the item in Merchandising, else an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/expenses/delete

Input Payload Details

Table 5-484 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-485 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
expense	No	Collection of Object	References a collection of item expense.

Table 5-486 Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	The supplier affected by the deletion activity.
componentId	Yes	String (10)	The code representing the specific expense component to be deleted.
dischargePort	Yes	String (5)	The port where the item will enter the importing country.

Table 5-486 (Cont.) Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	No	String (3)	The country where the item was manufactured or significantly altered.
ladingPort	No	String (5)	The final destination lading port of the item if the Expense Type is Country.
costZoneId	No	Number (10)	The final destination zone of the item if the Expense Type is Zone.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "expense": [
        {
          "supplier": 2400,
          "componentId": "ORDCST",
          "dischargePort": "480",
          "originCountry": "US",
          "ladingPort": "7",
          "costZoneId": 696323
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
```



```

    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Delete Item HTS

Functional Area

Items - Item Definition

Business Overview

This service is used to delete any Harmonized Tariff Schedule (HTS) that is linked with an item within Merchandising. In order to perform the deletion, the record to be deleted must exist in Merchandising, else an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/hts/delete

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-487 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-488 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
hts	No	Collection of Object	References a collection of item hts.

Table 5-489 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hts	Yes	String (25)	The unique identifier for the Harmonized Tariff Schedule code.
importCountry	Yes	String (3)	The unique identifier for the country that the item will be imported into.

Table 5-489 (Cont.) Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The country where the item was manufactured or significantly altered.
effectFrom	Yes	date	The date from which the Item/HTS relationship is valid. This field is used to maintain historical information related to the Item/HTS relationship.
effectTo	Yes	date	The date until when the Item/HTS relationship is valid. This field is used to maintain historical information related to the item/HTS relationship.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "hts": [
        {
          "hts": "9014111112",
          "importCountry": "US",
          "originCountry": "CA",
          "effectFrom": "2001-12-31",
          "effectTo": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
```

```

    "message": "Error found in validation of input payload",
    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}

```

Delete Item HTS Assessments

Functional Area

Items - Item Definition

Business Overview

This service is used to delete any assessments that are associated with an existing item HTS record within Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/hts/assessments/delete

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-490 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-491 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
hts	No	Collection of Object	References a collection of item hts.

Table 5-492 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hts	Yes	String (25)	The unique identifier for the Harmonized Tariff Schedule code.
importCountry	Yes	String (3)	The unique identifier for the country that the item will be imported into.

Table 5-492 (Cont.) Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The country where the item was manufactured or significantly altered.
effectFrom	Yes	date	The date from which the Item/HTS relationship is valid. This field is used to maintain historical information related to the Item/HTS relationship.
effectTo	Yes	date	The date until when the Item/HTS relationship is valid. This field is used to maintain historical information related to the item/HTS relationship.
assessments	No	Collection of Object	References a collection of assessments associated to the item HTS.

Table 5-493 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	Yes	String (10)	The code representing the specific assessment component to be deleted.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "106150023",
      "hts": [
        {
          "hts": "9014111112",
          "importCountry": "US",
          "originCountry": "CA",
          "effectFrom": "2001-12-31",
          "effectTo": "2001-12-31",
          "assessments": [
            {
              "componentId": "DTYCAUS"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Expenses

Functional Area

Items - Item Definition

Business Overview

This service is used to update one or more expenses that are associated with an item within Merchandising. In order to perform an update, the specified item/supplier/component ID must exist within Merchandising, else an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/expenses/update

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-494 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-494 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-495 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
expense	No	Collection of Object	References a collection of item expenses.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-496 Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	The unique identifier for the supplier.
componentId	Yes	String (10)	The code representing the specific expense component.
dischargePort	Yes	String (5)	The port where the item will enter the importing country.
originCountry	No	String (3)	The country where the item was manufactured or significantly altered.
ladingPort	No	String (5)	The final destination lading port of the item if the Expense Type is Country.
costZoneId	No	Number (10)	The final destination zone of the item if the Expense Type is Zone.
costZoneGroupId	No	Number (4)	This field holds the items Zone Group, it is stored here for performance reasons.
baseExpenseInd	No	String (1)	Indicates whether or not this Set of Expenses is also the base Set of Expenses.
computationValueBase	No	String (10)	The Computation Value Base representing the base value for calculating the expense.

Table 5-496 (Cont.) Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentRate	No	Number (20,4)	The rate to be charged against the calculation, base. This field will hold a percentage value if the Calculation basis of the Component is Value or a monetary amount if the Calculation Basis is Specific.
perCount	No	Number (12,4)	A count indicating the amount of the Per Count Unit of Measure to which the rate applies. This column is only used for specific (per-unit) expenses.
perCountUom	No	String (4)	The unit of measure in which the Per Count is specified. This column is only used from specific (per-unit) expenses.
componentCurrency	No	String (3)	Contains the currency the expense is to be entered in. If the Calculation Basis is V (Value), then the comp_currency will be NULL.
updateOrdersInd	No	String (1)	Indicates that new changes will be reflected to associated order-location expense record.
nominationFlag1	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.

Table 5-496 (Cont.) Expense - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inDuty	No	String (1)	See description for nomFlag1. If ALC is turned on (i. e. system_options.import_ind = Y), inDuty flag is used to indicate the status of a component in duty calculation, inExpense flag is used to indicate the status of a component in ELC calculation, inALC flag is used to indicate the status of a component in ALC calculation, i. e. these 3 flags can not be used for any other purposes if ALC is turned on.
nominationFlag3	No	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	No	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.
inAlc	No	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Table 5-497 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-498 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "expense": [
        {
          "supplier": 2400,
          "componentId": "AGCOMM",
          "dischargePort": "480",
          "originCountry": "US",
          "ladingPort": "7",
          "costZoneId": 696323,
          "costZoneGroupId": 1000,
          "baseExpenseInd": "N",
          "computationValueBase": null,
          "componentRate": 5.6,
          "perCount": 3,
          "perCountUom": "EA",
          "componentCurrency": "USD",
          "updateOrdersInd": "Y",
          "nominationFlag1": "N",
          "inDuty": "N",
          "nominationFlag3": "N",
          "inExpense": "+",
          "inAlc": "N"
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
```

```
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item HTS

Functional Area

Items - Item Definition

Business Overview

This service is used to update HTS of an existing item in Merchandising. In order to update HTS, the record to be updated must exist in Merchandising, otherwise, an error will be returned. Status and origin country ID can be updated at the HTS level.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/hts/update

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-499 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-500 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
hts	No	Collection of Object	References a collection of item HTS.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-501 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hts	Yes	String (25)	The unique identifier for the Harmonized Tariff Schedule code.
importCountry	Yes	String (3)	The unique identifier for the country that the item will be imported into.
originCountry	Yes	String (3)	The country where the item was manufactured or significantly altered.
effectFrom	Yes	date	The date from which the Item/HTS relationship is valid. This field is used to maintain historical information related to the Item/HTS relationship.
effectTo	Yes	date	The date until when the Item/HTS relationship is valid. This field is used to maintain historical information related to the item/HTS relationship.
clearingZoneId	No	String (5)	The concept of 'clearing zone' represents a country's zone-level point of entry of goods. This field holds the Clearing Zone ID for the effective HTS code. Clearing zones are defined by import country.

Table 5-501 (Cont.) Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	No	String (1)	Used to indicate the status of the Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.

Table 5-502 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-503 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "hts": [
        {
          "hts": "9014111111",
          "importCountry": "US",
          "originCountry": "CA",
          "effectFrom": "2001-12-31",
          "effectTo": "2001-12-31",
          "clearingZoneId": null,
          "status": "W"
        }
      ]
    },
    {
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

```
    ]
  }
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item HTS Assessments

Functional Area

Items - Item Definition

Business Overview

This service is used to update existing item HTS assessments in Merchandising. To update HTS assessments, the record to be updated must exist in Merchandising; otherwise, an error will be returned. All HTS assessment fields except for component ID can be updated.

Service Type

PUT

ReST URL

`MerchIntegrations/services/item/hts/assessments/update`

Input Payload Details

Update - Object. See list of elements for detail

Items - Object. See list of elements for detail

Hts - Object. See list of elements for detail

Assessments - Object. See list of elements for detail

LocalizationExtensions - Object. See list of elements for detail

Attributes - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Item Location Upload Service

The following services are included in this functional area:

- [Create Item Locations](#)
- [Manage Item Location Status Update Requests](#)
- [Update Item Locations](#)

Create Item Locations

Functional Area

Items - Item Definition

Business Overview

This service subscribes to items and their locations from external systems to create item location combinations in Merchandising. Item/location relationships can be created for an item and a single location or using one of the levels of the organizational hierarchy.

When a new item location is created, this service will first validate that all required fields are present in the message. Additionally, when creating a new item location at least one detail line must also be included in the request payload. After that, business level validation on the input information will be performed.

Service Type

POST

ReST URL

MerchIntegrations/services/item/locations/create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-504 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-504 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of item-location records.

Table 5-505 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The identifier for the item.
hierarchyLevel	Yes	String (2)	The level of the organizational hierarchy to which the hier_value field pertains. Valid values are CH (chain), AR (area), RE (region), DI (district), S (store), and W (warehouse).
locations	Yes	Collection of Object	References a collection of hierarchy details.

Table 5-506 Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	Valid ID for a chain, area, region, district, store or warehouse given based on the hierarchy Level.
primarySupplier	No	Number (10)	The primary supplier for the item at the location. If not populated, this will default from the primary supplier on the ITEM_SUPPLIER table.
primaryCountry	No	String (3)	The primary country for the item at the location. If not populated on a create message this will default from the primary country on the ITEM_SUPP_COUNTRY table.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
localItemDescription	No	String (250)	Local description for the item at the location. If not populated on the message this will default from the ITEM_MASTER table.
status	Yes	String (1)	The status of the item at the location.
storeOrderMultiple	Yes	String (1)	The unit type of the item shipped from the warehouse to the location. Valid values are C (cases), I (inners), and E (eaches). Eaches will default if left null on a create message.
receiveAsType	No	String (1)	The unit in which the warehouse will receive goods. Valid values are E (eaches) and P (pack). This field only applies to buyer packs. If the item is not a buyer pack this field should be null.
taxableInd	Yes	String (1)	Indicates if the item is taxable at the location. Valid values are 'Y' and 'N'. If not populated on the message it will default to 'Y'.
ti	No	Number (12)	Number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of cases for a pallet.
hi	No	Number (12)	Number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of cases for a pallet.
dailyWastePercent	No	Number (12,4)	Average percentage lost from inventory on a daily basis due to natural wastage.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
localShortDescription	No	String (120)	The local short description of the item. Will default to the item's short description. Will be downloaded to the POS.
uinType	No	String (6)	This attribute will contain the type of the unique identification number (UIN) used to identify instances of the item at the location. Valid values for this attribute should be user-definable on the Merchandising code tables. This attribute should be nullable.
uinLabel	No	String (6)	This attribute will contain the label for the UIN when displayed in Store Inventory Operations. Valid values should be user-definable on the Merchandising code tables. This attribute should be nullable but a value is required when a UIN Type is specified.
captureTime	No	String (6)	This attribute will indicate when the UIN should be captured for an item during transaction processing. There are 2 valid values for this field which should be stored on the Merchandising code tables: Sale and Store Receiving. This attribute should be nullable.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
externalUinInd	No	String (1)	This Yes/No indicator indicates if UIN is being generated in the external system (e. g. SIM). This value will always be N (no) when the Capture Time attribute is set to Sale.
sourceMethod	No	String (1)	Determines the source method for a requested item at a store. Valid values: 'W'(warehouse) and 'S'(supplier), default value is 'S'.
sourceWarehouse	No	Number (10)	If the source_method is 'W', this field contains the source warehouse for fulfilling item requests at a store.
unitCost	No	Number (20,4)	The current corporate unit cost for the item in the location's currency. This field will only be used to establish the item supplier country loc unit cost; it cannot be modified.
purchaseType	No	String (1)	This contains a code to indicate whether the item is normal merchandise (i. e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This indicates if the cost for the consignment/ concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.
pricingControl	No	String (1)	This indicates whether the price for the consignment/ concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/ Concession ownership model is used within the Merchandising suite of solutions. Valid values are: R - Retailer S - Supplier If no value is provided for consignment/ concession purchase types, this will be defaulted from item/ supplier/country of sourcing /location if purchase type is Consignment/ Concession at item/ supplier/ country of sourcing/location level. If purchase type is Owned at item/ supplier/country of sourcing/location level, then this will be defaulted from item/ supplier/country of sourcing if purchase type is Consignment/ Concession at item/ supplier/country of sourcing level. If purchase type is Owned at item/ supplier/country of sourcing level, then this will be defaulted based on Default Pricing Control at supplier site level.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
promotableInd	No	String (1)	This field will indicate, if the item can be included on promotions or not.
rfidInd	No	String (1)	This indicates if the item is RFID tagged or not. Valid values are Yes (Y) and No (N).
ticketMeasureOfAnEach	No	Number (12,4)	Size of an each in terms of the uom_of_price. For example 12 oz. Used in ticketing.
ticketPerCount	No	Number (12,4)	Size to be used on the ticket in terms of the uom_of_price. For example, if the user wants the ticket to have the label print the price per ounce, this value would be 1. If the user wanted the price per 100 grams this value would be 100. Used in ticketing.
ticketUom	No	String (4)	Unit of measure that will be used on the ticket for this item.
primaryVariant	No	String (25)	This field is used to address sales of PLUs (i. e. above transaction level items) when inventory is tracked at a lower level (i. e. UPC). This field will only contain a value for items one level higher than the transaction level.
primaryCostPack	No	String (25)	This field contains an item number that is a simple pack containing the item in the item column for this record. If populated, the cost of the future cost table will be driven from the simple pack and the deals and cost changes for the simple pack.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inboundHandlingDays	No	Number (2)	This field indicates the number of inbound handling days for an item at a warehouse type location.
rangedInd	No	String (1)	This column determines if the location is ranged intentionally by the user for replenishment/selling or incidentally ranged by the RMS programs when item is not ranged to a specific location on the transaction.
costingLocation	No	Number (10)	Numeric identifier of the costing location for the franchise store. This field may contain a store or a warehouse.
sellingPhaseStartDate	No	date	It is used to determine the timing of the generation of markdown recommendation, which will occur x weeks after the start of this phase.
updSellDateOnStOrdRcpt	No	String (1)	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a purchase order of this item is received at the store.
updSellDateOnWh2stRcpt	No	String (1)	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a warehouse to store transfer/ allocation of this item is received at the store.
updSellDateOnSt2stRcpt	No	String (1)	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a store to store transfer of this item is received at the store.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
minUnitsForSellDateReset	No	Number (12,4)	This field contains the minimum shipped quantity of an item on a transfer/allocation that will trigger the resetting of the Selling Phase Start Date. This value is only used if either of the Set Selling Start Date upon Receipt of a Warehouse to Store Transfer/Allocation or Set Selling Start Date upon Receipt of a Store to Store Transfer are set to Yes (Y).
mfgSugRetailInclTax	No	Number (20,4)	This field holds the tax-inclusive Manufacturer Suggested Retail in local currency and in item's standard UOM. It is used when default_tax_type is GTS or SVAT.
mfgSugRetailExclTax	No	Number (20,4)	This field holds the tax-exclusive Manufacturer Suggested Retail in local currency and in item's standard UOM. It is used when default_tax_type is SALES.
postingClassifier	No	String (6)	'Contains a classifier that will be used to segregate the sales posting of the item-loc combination to various GL accounts. Valid values are user-definable on the RMS code tables under code_type 'POCL'.

Table 5-506 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
trait	No	Record	References a location trait record.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-507 Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
launchDate	No	date	The date the item should first be sold at the location. Value will be downloaded to the POS. No internal processing occurs from this field.
quantityKeyOptions	No	String (6)	Determines whether the qty key on a POS should be used for this item at the location. Valid values are in the code_type 'RPO'. Current values include 'R - required', 'P - Prohibited', and 'O - Optional'. Additional values can be added to the code type

Table 5-507 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manualPriceEntry	No	String (6)	Determines whether the price can/should be entered manually on a POS for this item at the location. Valid values are in the code_type 'RPO'. Current values include 'R - required', 'P - Prohibited', and 'O - Optional'. Additional values can be added.
depositCode	No	String (6)	Indicates whether a deposit is associated with this item at the location. Values are in the code_type 'DEPO'. The only processing in Merchandising is downloading it to the POS. Examples are None (NONE), Maine - 0.10 (ME2) and Maine - 0.05 (ME1).
foodStampInd	No	String (1)	Indicates whether the item is approved for food stamps at the location. This value will be downloaded to the POS.
wicInd	No	String (1)	Indicates whether the item is approved for WIC at the location. This value will be downloaded to the POS.
proportionalTarePercent	No	Number (12,4)	For items sold by weight, this is the proportion of the total weight of a unit of an item that is packaging.
fixedTareValue	No	Number (12,4)	Holds the weight of the packaging in items sold by weight at the location; i. e. if the tare item is bulk candy, this is weight of the bag and twist tie). The only processing Merchandising does is to download it to POS.

Table 5-507 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fixedTareUom	No	String (4)	Holds the unit of measure value associated with the tare value. The only processing Merchandising does involving the fixed tare value and UOM is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to Merchandising, etc
rewardEligibleInd	No	String (1)	Holds whether the item is legally valid for various types of bonus point/award programs at the location. This value will be downloaded to the POS.
nationalBrandCompetitorItem	No	String (25)	Holds the nationally branded item to which you would like to compare the current item. This nationally branded item must exist as an item in Merchandising (i. e. on the item_master table).
returnPolicy	No	String (6)	The return policy for the item at the location. Valid values are in the code_type 'RETP'. This value will be downloaded to the POS. Examples are All returns accepted (ALL), Customer ID needed for return (ID) and Manager approval needed for return (MNGR).
stopSaleInd	No	String (1)	Indicates that sale of the item should be stopped immediately at the location (i. e. in case of recall etc). This value will be downloaded to the POS.

Table 5-507 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
electronicMarketClubs	No	String (6)	Code for the marketing clubs to which the item belongs at the location. Valid values are in the code_type 'MTKC'. Examples are Baby Club (B), Baby/Family Clubs (BF) and Deli Club (D).
reportCode	No	String (6)	Code to determine which reports the location should run. Values are in the code type REPC (Report Code). Examples are All Locations Reports(ALL) and No Location Report (NO).
requiredShelfLifeOnSelection	No	Number (4)	Holds the required shelf life for an item on selection in days. This field is not required. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it.
requiredShelfLifeOnReceipt	No	Number (4)	Holds the required shelf life for an item on receipt in days. This field is not required. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it for processing
investmentBuyShelfLife	No	Number (4)	The Investment Buy-specific shelf life for the item/location.

Table 5-507 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeReorderableInd	No	String (1)	A code (Y or N) to indicate whether the store may re-order the item. Will default to N. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems.
rackSize	No	String (6)	Indicates the rack size that should be used for the item. Valid values for the field are in the code_type 'RACK'. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate to external systems. Examples are large (L), Medium (M) and Small (S).
fullPalletItem	No	String (1)	A code (Y or N) to indicate whether a store must reorder an item in full pallets only. Will default to N. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate to external systems.

Table 5-507 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inStoreMarketBasket	No	String (6)	The in store market basket code for this item/location. Valid values for the field are in the code_type 'STMB'. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems. Examples are A item at this loc (A), B item at this loc (B) and C item at this loc (C).
storageLocation	No	String (7)	Holds the current storage location or bin number for the item at the location. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it for processing.
alternateStorageLocation	No	String (7)	Holds the preferred alternate storage location or bin number for the item at the location. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it for processing.
returnableInd	No	String (1)	This field will contain a value of 'Y'es when the item can be returned to the location,
refundableInd	No	String (1)	This field will contain a value of 'Y'es when the item is refundable at the location.

Table 5-507 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
backOrderInd	No	String (1)	This field will contain a value of 'Y'es when the item can be back ordered to the location.

Table 5-508 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-509 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-510 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100000024",
      "hierarchyLevel": "S",
      "locations": [
```

```

{
  "hierarchyValue": 6000,
  "primarySupplier": 6100,
  "primaryCountry": "US",
  "localItemDescription": "100000024 at 6000",
  "status": "A",
  "storeOrderMultiple": "E",
  "receiveAsType": null,
  "taxableInd": "N",
  "ti": 1,
  "hi": 1,
  "dailyWastePercent": 1,
  "localShortDescription": "100000024 at 6000",
  "uinType": null,
  "uinLabel": null,
  "captureTime": null,
  "externalUinInd": "N",
  "sourceMethod": "S",
  "sourceWarehouse": null,
  "unitCost": 9.65,
  "purchaseType": null,
  "calculationBasis": null,
  "purchaseRate": 1,
  "pricingControl": "R",
  "promotableInd": "N",
  "rfidInd": "N",
  "ticketMeasureOfAnEach": 2,
  "ticketPerCount": 1,
  "ticketUom": "EA",
  "primaryVariant": "100100090",
  "primaryCostPack": "100100080",
  "inboundHandlingDays": 1,
  "rangedInd": "Y",
  "costingLocation": 199,
  "sellingPhaseStartDate": "2001-12-31",
  "updSellDateOnStOrdRcpt": "N",
  "updSellDateOnWh2stRcpt": "N",
  "updSellDateOnSt2stRcpt": "N",
  "minUnitsForSellDateReset": 50,
  "mfgSugRetailInclTax": 20,
  "mfgSugRetailExclTax": 18.12,
  "postingClassifier": "String",
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": null
    }
  ],
  "trait": {
    "launchDate": "2001-12-31",
    "quantityKeyOptions": null,
    "manualPriceEntry": null,
    "depositCode": null,
    "foodStampInd": "N",
    "wicInd": "N",

```

```

    "proportionalTarePercent": 3,
    "fixedTareValue": 3,
    "fixedTareUom": "EA",
    "rewardEligibleInd": "N",
    "nationalBrandCompetitorItem": null,
    "returnPolicy": null,
    "stopSaleInd": "N",
    "electronicMarketClubs": null,
    "reportCode": null,
    "requiredShelfLifeOnSelection": 1000,
    "requiredShelfLifeOnReceipt": 1000,
    "investmentBuyShelfLife": 1000,
    "storeReorderableInd": "Y",
    "rackSize": null,
    "fullPalletItem": null,
    "inStoreMarketBasket": null,
    "storageLocation": null,
    "alternateStorageLocation": null,
    "returnableInd": "N",
    "refundableInd": "N",
    "backOrderInd": "N"
  },
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Manage Item Location Status Update Requests**Functional Area**

Items - Item Definition

Business Overview

This service provides the capability to create and maintain requests to update item/location status from an external system or source. This service takes in a collection of item locations along with the desired status and the effective date upon which each update should take place.

Possible status values can be found under the Item/Location Statuses (STST) code type, which includes values such as Active (A), Inactive (I), Discontinued (C) and Deleted (D). The effective date on a new request must be at least one day later than the current business date (virtual date + 1). The service will accept approved transaction level items or approved reference items. When a reference item is provided, the system will update the item/location status for the associated transaction level item. Locations must be a store, physical warehouse or virtual warehouse. If a physical warehouse is provided, requests will be created/updated/deleted for all virtual warehouses under the physical warehouse. Requests can be created for item/location combinations where the item is not yet ranged to the location. In this case, when the request is processed by the Scheduled Item Maintenance (sitmain) batch program, the item will be ranged to the location with the provided status.

This single service supports complete maintenance of item location status update requests, which includes the ability to create, update and delete requests. The service takes in an external request reference number. The external system can update or delete a previously created request by passing updates against an external request reference number sent earlier. Removal of item locations or scheduled status date can be accomplished by omitting them in an update request. Deleting of an existing request will require the Delete flag to be set on the update request. No details (item locations or status-date combinations) are required on a Request having the Delete flag set.

Item/location status update requests created via this service can also be viewed and maintained via the Item/Location Update screen which is accessed via the Manage Scheduled Item Updates link under the Items > Scheduled Item Maintenance folder in the Tasks list.

This webservice will enable interfacing of item locations status change requests into Merchandising. Processing of these requests out will continue to be carried out by the Scheduled Item Maintenance (sitmain) batch process. This batch program will update the item-location status the night before the effective date on the request.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/location/statusUpdateRequests/manage

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-511 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	Collection of the input details for which item location status update is requested.

Table 5-512 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
externalRequestId	Yes	Number (10)	External request id against which the item location status update request will be created.
description	No	String (120)	Request description.
deleteFlag	No	String (1)	Indicate if the request is to be deleted. Valid values are Y - Yes and N - No.
details	No	Collection of Object	Collection of status/update dates on which the new status will be updated. The maximum number of the status/update dates allowed is 999.
itemLocations	No	Collection of Object	Collection of item/locations for which the status updates are requested. The maximum number of the items/locations allowed is 999.

Table 5-513 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	Yes	String (1)	The status to update. Valid values are A - Active, I - Inactive, C - Discontinued and D - Delete.
updateDate	Yes	date	The future date on which status update will take place. The expected date format is YYYY-MM-DD.

Table 5-514 ItemLocations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The transaction level or below transaction level item for which status update is requested.
location	Yes	Number (10)	The location at which item status will be updated. Valid location can be a store, physical warehouse or virtual warehouse.

Table 5-515 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
externalRequestId	Yes	Number (10)	External requested id against which errors are encountered.
errorMessages	Yes	Array of String	List of errors identified during business data processing of the request.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "externalRequestId": 12122,
      "description": "Request description",
      "deleteFlag": "N",
      "details": [
        {
          "status": "I",
          "updateDate": "2001-12-31"
        }
      ],
      "itemLocations": [
        {
          "item": "101050103",
          "location": 1531
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-516 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
externalRequestId	Yes	Number (10)	External requested id against which errors are encountered.
errorMessages	Yes	Array of String	List of errors identified during business data processing of the request.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "externalRequestId": 12122,
      "errorMessages": [
        "Input location is not a valid store or warehouse. Input location is
1531."
      ]
    }
  ]
}
```

Update Item Locations

Functional Area

Items - Item Definition

Business Overview

This service is used to subscribe to items and their linked locations from external systems in order to update this data at the item/location level within Merchandising. Item/location relationships can be updated for an item and a single location or using one of the levels of the organizational hierarchy.

When an item location is updated, this service will first validate that all the required fields are present in the message. Additionally, when creating a new item/location record, at least one detail line must also be included in the request payload in order to be able to perform business level validation on the input information.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/locations/update

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-517 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of item-location records.

Table 5-518 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The identifier for the item.
hierarchyLevel	Yes	String (2)	The level of the organizational hierarchy to which the hier_value field pertains. Valid values are CH (chain), AR (area), RE (region), DI (district), S (store), and W (warehouse).
locations	Yes	Collection of Object	References a collection of hierarchy details.

Table 5-519 Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The same value as in the hier_level field.
primarySupplier	No	Number (10)	The primary supplier for the item at the location. If not populated on a create message this will default from the primary supplier on the ITEM_SUPPLIER table.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryCountry	No	String (3)	The primary country for the item at the location. If not populated on a create message this will default from the primary country on the ITEM_SUPP_COUNTRY table.
localItemDescription	No	String (250)	Local description for the item at the location. If not populated on the message this will default from the ITEM_MASTER table.
status	Yes	String (1)	The status of the item at the location.
storeOrderMultiple	Yes	String (1)	The unit type of the item shipped from the warehouse to the location. Valid values are C (cases), I (inners), and E (eaches). Eaches will default if left null on a create message.
receiveAsType	No	String (1)	The unit in which the warehouse will receive goods. Valid values are E (eaches) and P (pack). This field only applies to buyer packs. If the item is not a buyer pack this field should be null.
taxableInd	Yes	String (1)	Indicates if the item is taxable at the location. Valid values are 'Y' and 'N'. If not populated on the message it will default to 'Y'.
ti	No	Number (12)	Number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of cases for a pallet.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hi	No	Number (12)	Number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of cases for a pallet.
dailyWastePercent	No	Number (12,4)	Average percentage lost from inventory on a daily basis due to natural wastage.
localShortDescription	No	String (120)	The local short description of the item. Will default to the item's short description. Will be downloaded to the POS.
uinType	No	String (6)	This attribute will contain the type of the unique identification number (UIN) used to identify instances of the item at the location. Valid values for this attribute should be user-definable on the Merchandising code tables. This attribute should be nullable.
uinLabel	No	String (6)	This attribute will contain the label for the UIN when displayed in Store Inventory Operations. Valid values should be user-definable on the Merchandising code tables. This attribute should be nullable but a value is required when a UIN Type is specified.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
captureTime	No	String (6)	This attribute will indicate when the UIN should be captured for an item during transaction processing. There are 2 valid values for this field which should be stored on the Merchandising code tables: Sale and Store Receiving. This attribute should be nullable.
externalUinInd	No	String (1)	This Yes/No indicator indicates if UIN is being generated in the external system (e. g. SIM). This value will always be N (no) when the Capture Time attribute is set to Sale.
sourceMethod	No	String (1)	Determines the source method for a requested item at a store. Valid values: 'W'(warehouse) and 'S'(supplier), default value is 'S'.
sourceWarehouse	No	Number (10)	If the source_method is 'W', this field contains the source warehouse for fulfilling item requests at a store.
promotableInd	No	String (1)	This field will indicate, if the item can be included on promotions or not.
rfidInd	No	String (1)	This indicates if the item is RFID tagged or not. Valid values are Yes (Y) and No (N).
ticketMeasureOfAnEach	No	Number (12,4)	Size of an each in terms of the uom_of_price. For example 12 oz. Used in ticketing.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketPerCount	No	Number (12,4)	Size to be used on the ticket in terms of the uom_of_price. For example, if the user wants the ticket to have the label print the price per ounce, this value would be 1. If the user wanted the price per 100 grams this value would be 100. Used in ticketing.
ticketUom	No	String (4)	Unit of measure that will be used on the ticket for this item.
primaryVariant	No	String (25)	This field is used to address sales of PLUs (i. e. above transaction level items) when inventory is tracked at a lower level (i. e. UPC). This field will only contain a value for items one level higher than the transaction level.
primaryCostPack	No	String (25)	This field contains an item number that is a simple pack containing the item in the item column for this record. If populated, the cost of the future cost table will be driven from the simple pack and the deals and cost changes for the simple pack.
inboundHandlingDays	No	Number (2)	This field indicates the number of inbound handling days for an item at a warehouse type location.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rangedInd	No	String (1)	This column determines if the location is ranged intentionally by the user for replenishment/selling or incidentally ranged by the RMS programs when item is not ranged to a specific location on the transaction.
costingLocation	No	Number (10)	Numeric identifier of the costing location for the franchise store. This field may contain a store or a warehouse.
sellingPhaseStartDate	No	date	It is used to determine the timing of the generation of markdown recommendation, which will occur x weeks after the start of this phase.
updSellDateOnStOrdRcpt	No	String (1)	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a purchase order of this item is received at the store.
updSellDateOnWh2stRcpt	No	String (1)	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a warehouse to store transfer/ allocation of this item is received at the store.
updSellDateOnSt2stRcpt	No	String (1)	This indicates whether or not to update the Selling Phase Start Date for the item/loc when a store to store transfer of this item is received at the store.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
minUnitsForSellDateReset	No	Number (12,4)	This field contains the minimum shipped quantity of an item on a transfer/allocation that will trigger the resetting of the Selling Phase Start Date. This value is only used if either of the Set Selling Start Date upon Receipt of a Warehouse to Store Transfer/Allocation or Set Selling Start Date upon Receipt of a Store to Store Transfer are set to Yes (Y).
mfgSugRetailInclTax	No	Number (20,4)	This field holds the tax-inclusive Manufacturer Suggested Retail in local currency and in item's standard UOM. It is used when default_tax_type is GTS or SVAT.
mfgSugRetailExclTax	No	Number (20,4)	This field holds the tax-exclusive Manufacturer Suggested Retail in local currency and in item's standard UOM. It is used when default_tax_type is SALES.
postingClassifier	No	String (6)	'Contains a classifier that will be used to segregate the sales posting of the item-loc combination to various GL accounts. Valid values are user-definable on the RMS code tables under code_type 'POCL'.

Table 5-519 (Cont.) Locations - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
trait	No	Record	Child node.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-520 Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
launchDate	No	date	The date the item should first be sold at the location. Value will be downloaded to the POS. No internal processing occurs from this field.
quantityKeyOptions	No	String (6)	Determines whether the qty key on a POS should be used for this item at the location. Valid values are in the code_type 'RPO'. Current values include 'R - required', 'P - Prohibited', and 'O - Optional'. Additional values can be added to the code type

Table 5-520 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manualPriceEntry	No	String (6)	Determines whether the price can/should be entered manually on a POS for this item at the location. Valid values are in the code_type 'RPO'. Current values include 'R - required', 'P - Prohibited', and 'O - Optional'. Additional values can be added.
depositCode	No	String (6)	Indicates whether a deposit is associated with this item at the location. Values are in the code_type 'DEPO'. The only processing in Merchandising is downloading it to the POS. Examples are None (NONE), Maine - 0.10 (ME2) and Maine - 0.05 (ME1).
foodStampInd	No	String (1)	Indicates whether the item is approved for food stamps at the location. This value will be downloaded to the POS.
wicInd	No	String (1)	Indicates whether the item is approved for WIC at the location. This value will be downloaded to the POS.
proportionalTarePercent	No	Number (12,4)	For items sold by weight, this is the proportion of the total weight of a unit of an item that is packaging.
fixedTareValue	No	Number (12,4)	Holds the weight of the packaging in items sold by weight at the location; i. e. if the tare item is bulk candy, this is weight of the bag and twist tie). The only processing Merchandising does is to download it to POS.

Table 5-520 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fixedTareUom	No	String (4)	Holds the unit of measure value associated with the tare value. The only processing Merchandising does involving the fixed tare value and UOM is downloading it to the POS. Fixed tare is not subtracted from items sold by weight when sales are uploaded to Merchandising, etc
rewardEligibleInd	No	String (1)	Holds whether the item is legally valid for various types of bonus point/award programs at the location. This value will be downloaded to the POS.
nationalBrandCompetitorItem	No	String (25)	Holds the nationally branded item to which you would like to compare the current item. This nationally branded item must exist as an item in Merchandising (i. e. on the item_master table).
returnPolicy	No	String (6)	The return policy for the item at the location. Valid values are in the code_type 'RETP'. This value will be downloaded to the POS. Examples are All returns accepted (ALL), Customer ID needed for return (ID) and Manager approval needed for return (MNGR).
stopSaleInd	No	String (1)	Indicates that sale of the item should be stopped immediately at the location (i. e. in case of recall etc). This value will be downloaded to the POS.

Table 5-520 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
electronicMarketClubs	No	String (6)	Code for the marketing clubs to which the item belongs at the location. Valid values are in the code_type 'MTKC'. Examples are Baby Club (B), Baby/Family Clubs (BF) and Deli Club (D).
reportCode	No	String (6)	Code to determine which reports the location should run. Values are in the code type REPC (Report Code). Examples are All Locations Reports(ALL) and No Location Report (NO).
requiredShelfLifeOnSelection	No	Number (4)	Holds the required shelf life for an item on selection in days. This field is not required. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it.
requiredShelfLifeOnReceipt	No	Number (4)	Holds the required shelf life for an item on receipt in days. This field is not required. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it for processing
investmentBuyShelfLife	No	Number (4)	The Investment Buy-specific shelf life for the item/location

Table 5-520 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeReorderableInd	No	String (1)	A code (Y or N) to indicate whether the store may re-order the item. Will default to N. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems.
rackSize	No	String (6)	Indicates the rack size that should be used for the item. Valid values for the field are in the code_type 'RACK'. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate to external systems. Examples are large (L), Medium (M) and Small (S).
fullPalletItem	No	String (1)	A code (Y or N) to indicate whether a store must reorder an item in full pallets only. Will default to N. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate to external systems.

Table 5-520 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inStoreMarketBasket	No	String (6)	The in store market basket code for this item/location. Valid values for the field are in the code_type 'STMB'. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems. Examples are A item at this loc (A), B item at this loc (B) and C item at this loc (C).
storageLocation	No	String (7)	Holds the current storage location or bin number for the item at the location. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it for processing.
alternateStorageLocation	No	String (7)	Holds the preferred alternate storage location or bin number for the item at the location. No Merchandising processing is based on the value in this field. It is clients responsibility to communicate this value to external systems that may need it for processing.
returnableInd	No	String (1)	This field will contain a value of 'Y'es when the item can be returned to the location
refundableInd	No	String (1)	This field will contain a value of 'Y'es when the item is refundable at the location.

Table 5-520 (Cont.) Trait - Object. See list of elements for detail

Element Name	Required	Data Type	Description
backOrderInd	No	String (1)	This field will contain a value of 'Y'es when the item can be back ordered to the location

Table 5-521 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-522 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-523 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100000024",
      "hierarchyLevel": "S",
      "locations": [
```

```

{
  "hierarchyValue": 6000,
  "primarySupplier": 6100,
  "primaryCountry": "US",
  "localItemDescription": "100000024 at 6000",
  "status": "I",
  "storeOrderMultiple": "E",
  "receiveAsType": null,
  "taxableInd": "N",
  "ti": 1,
  "hi": 1,
  "dailyWastePercent": 1,
  "localShortDescription": "100000024 at 6000",
  "uinType": null,
  "uinLabel": null,
  "captureTime": null,
  "externalUinInd": "N",
  "sourceMethod": "S",
  "sourceWarehouse": null,
  "promotableInd": "N",
  "rfidInd": "N",
  "ticketMeasureOfAnEach": 2,
  "ticketPerCount": 1,
  "ticketUom": "EA",
  "primaryVariant": "100100090",
  "primaryCostPack": "100100080",
  "inboundHandlingDays": 1,
  "rangedInd": "Y",
  "costingLocation": 199,
  "sellingPhaseStartDate": "2001-12-31",
  "updSellDateOnStOrdRcpt": "N",
  "updSellDateOnWh2stRcpt": "N",
  "updSellDateOnSt2stRcpt": "N",
  "minUnitsForSellDateReset": 50,
  "mfgSugRetailInclTax": 20,
  "mfgSugRetailExclTax": 18.16,
  "postingClassifier": "String",
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": null
    }
  ],
  "trait": {
    "launchDate": "2001-12-31",
    "quantityKeyOptions": null,
    "manualPriceEntry": null,
    "depositCode": null,
    "foodStampInd": "N",
    "wicInd": "N",
    "proportionalTarePercent": 3,
    "fixedTareValue": 3,
    "fixedTareUom": "EA",
    "rewardEligibleInd": "N",
    "nationalBrandCompetitorItem": null,

```

```

    "returnPolicy": null,
    "stopSaleInd": "N",
    "electronicMarketClubs": null,
    "reportCode": null,
    "requiredShelfLifeOnSelection": 1000,
    "requiredShelfLifeOnReceipt": 1000,
    "investmentBuyShelfLife": 1000,
    "storeReorderableInd": "Y",
    "rackSize": null,
    "fullPalletItem": null,
    "inStoreMarketBasket": null,
    "storageLocation": null,
    "alternateStorageLocation": null,
    "returnableInd": "N",
    "refundableInd": "N",
    "backOrderInd": "N"
  },
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {

```

```
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
]
}
```

Item Number Reservation Upload Service

The following services are included in this functional area:

- [Generate and Reserve Item Numbers](#)

Manage Item Numbers

Functional Area

Items - Item Definition

Business Overview

This service allows external systems to create or delete item number reservations. A list of item identifiers can be provided to be reserved (or removed from reservation) within Merchandising. These pre-reserved numbers will not be considered for any auto-generated item numbers or manual item entry in Merchandising. Reserved numbers can be used as identifiers on new items through Item spreadsheet upload by providing the desired reserved item identifier and setting the Pre-Reserved flag to Yes on the Item_Master worksheet. During upload into Merchandising, the system will validate that the number exists as a reserved number. If found, the new item will be uploaded with the reserved number as its identifier and the reserved number will be deleted from the item reservation table (SVC_ITEM_RESERVATION).

- With an Action of CREATE, this service accepts one or more item numbers to be reserved. If the Action is left blank, it will be assumed to be CREATE.
 - The provided item identifiers must not correspond to existing items in Merchandising, nor should they be an existing reserved item number.
- With an Action of DELETE, this service accepts one or more currently reserved item numbers. These reservations will be removed, freeing up the identifiers for future item creation in Merchandising.
 - The provided item identifiers must be existing reserved item numbers.

Service Type

POST

ReST URL

MerchIntegrations/services/item/itemNumbers/manage

Input Payload Details

Table 5-524 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items identifiers to be reserved.

Table 5-525 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (10)	This optional field should be used to delete an existing reserved item number. Valid values are CREATE or DELETE. If this is not provided, it is defaulted to CREATE.
item	Yes	String (25)	References the item identifier that the external system wants to reserve. This item identifier should not exist in item_master and svc_item_reservation table.

Table 5-526 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Reference the item identifier that failed business validation
errorMessage	Yes	String (2000)	Contains the error message

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "item": "1000012332"
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service. The element businessError will be present if the payload passes schema validation but an exception is caught while processing the business logic.

Table 5-527 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Reference the item identifier that failed business validation
errorMessage	Yes	String (2000)	Contains the error message

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "item": "1000012332",
      "errorMessage": "An item with this number is already in use"
    }
  ]
}
```

Generate and Reserve Item Numbers**Functional Area**

Items - Item Definition

Business Overview

This service allows external systems such as Oracle Retail Assortment Planning (AP) to send a request to Merchandising to generate and reserve one or more item numbers. The next available item numbers are generated by Merchandising and are sent back to the calling solution as a response. These reserved numbers will not be considered for any auto-generated item numbers or manual item entry in Merchandising.

- This service allows the calling system to ask for 1 to n item numbers where n cannot exceed the 'Maximum Item Reserved Quantity', which is defined in the 'Coresvc_Item_Configuration' worksheet of the 'Item Data Loading Configuration' foundation data spreadsheet template, which is listed under the 'Administration' template type.
- The service generates the numbers and they are considered reserved until they expire. The calling system can specify for how long the numbers are reserved. The days until expiry cannot exceed the 'Maximum Item Expiry Days', which is defined in the 'Coresvc_Item_Configuration' worksheet of the 'Item Data Loading Configuration' foundation data spreadsheet template, which is listed under the 'Administration' template type.
- The generated numbers can be of any item number type for which Merchandising supports auto-generation: ITEM, UPC-A, UPC-AS, EAN13, or EAN13S.
- Reserved numbers are stored in the Item Reservation table (SVC_ITEM_RESERVATION).
- Reserved numbers can be used as identifiers on new items through Item spreadsheet upload by providing the desired, reserved item identifier and setting the Pre-Reserved flag to Yes on the Item_Master worksheet. During upload into Merchandising, the system will validate that the number exists as a reserved number. If found, the new item will be uploaded with the reserved number as its identifier and the reserved number will be deleted from the item reservation table (SVC_ITEM_RESERVATION).

Service Type

POST

ReST URL

MerchIntegrations/services/item/itemNumbers/reserve

Input Payload Details

Table 5-528 Reserve - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemNumberType	Yes	String (6)	Contains the type of item number to reserve. It must be a valid number type in RMS that supports auto generation of item numbers, including ITEM, UPC-A, UPC-AS, EAN13.
quantity	Yes	Number (10)	Contains the number of item numbers to reserve. The maximum number of items that can be requested is configurable using Item Service configuration.
daysUntilExpiry	Yes	Number (10)	Indicates how long the calling solution wants Merchandising to retain the reservation. After this many days, the reservation will be released allowing these numbers to be used for other purposes. This is required and must be a value greater than 0. The max number of days is configurable using Item Service configuration.

Sample Input Message

```
{  
  "itemNumberType": "ITEM",  
  "quantity": 50,  
  "daysUntilExpiry": 14  
}
```

Response Code: 200 (Success)

Table 5-529 ReserveResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	This is collection record. Refer to corresponding record details

Table 5-530 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the unique item number reserved in RMS.
itemNumberType	Yes	String (6)	Contains the type of item number in the item field.
expiryDate	Yes	date	The date when the reserved item number will expire.

Sample Response Message

```
{
  "items": [
    {
      "item": "100100076",
      "itemNumberType": "UPC-A",
      "expiryDate": "2001-12-31"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Item Placeholder Upload Service

The following services are included in this functional area:

- [Create Placeholder Items](#)

Create Placeholder Items

Functional Area

Items - Item Definition

Business Overview

This service supports subscribing to new placeholder items (Style/Color or Styles) created in A&IP as part of the slow item creation workflow. Using this service, A&IP can create items in the Item induction staging area within Merchandising with partial information of the item that can be enriched at a later point using the item induction process. The item number fetched using the Reserve item number service for the placeholder item number is used by A&IP.

The following item attributes are defaulted in this service -

- The transaction level of the item will be 2.
- The item will be a regular (non-pack, non-catchweight), sellable, orderable, inventoried, merchandise one.
- The item will be created in the Worksheet status
- The standard UOM will be set to Each (EA)
- The Store order multiple will be set to 'E' (Eaches).
- The item number type will be Manual.
- The aggregate indicators for the 2nd, 3rd and 4th differentiators will be set to 'N'.
- The pallet/case/inner name will be set to PAL/CS/INR respectively.
- The only UDA supported is LV (List of Values)

Service Type

POST

ReST URL

MerchIntegrations/services/item/placeholderItems/create

Input Payload Details

Table 5-531 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	It's a collection of Items record. Refer to items details

Table 5-532 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
itemParent	No	String (25)	The item at the level above this item. This value must be predefined as an item on the ITEM_MASTER table.
itemLevel	Yes	Number (1)	The level of the item. Valid values are 1 - Style item, 2 - Sku

Table 5-532 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff1	No	String (10)	A differentiator id which differentiates a transaction level item from its parent item. If the item on the message is a parent this field may hold a differentiator group id.
dept	Yes	Number (4)	The number of the department to which this item belongs. This value must be predefined on the DEPS table. It cannot be modified.
class	Yes	Number (4)	The number of the class to which the item belongs. This value must be predefined on the CLASS table. It cannot be modified.
subclass	Yes	Number (4)	The number of the subclass to which the item belongs. This value must be predefined on the SUBCLASS table. It cannot be modified.
itemDescription	Yes	String (250)	The long description of the item.
itemSuppliers	No	Collection of Object	It is a collection of ItemSupplier record. Refer to itemSupplier details
itemUdas	No	Collection of Object	It is a collection of ItemUda's record. Refer to itemUdas details
itemAggregateInd	No	String (1)	Indicator to aggregate inventory and sales for the item.
diff1AggregateInd	No	String (1)	Indicator for the corresponding diff. Indicator to aggregate inventory and sales for an item at Parent/Diff level (e.g Style/Color or Style/Size).
brand	No	String (120)	This is added to capture the brand information of an item.

Table 5-533 ItemSuppliers - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	Number (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.

Table 5-534 ItemUdas - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number identifying the User-Defined Attribute of type List of Value
udaValue	No	String (30)	This field contains value of the Used Defined attribute for the item.

Sample Input Message

```
{
  "items": [
    {
      "item": "100100076",
      "itemParent": null,
      "itemLevel": "1",
      "diff1": "BLACK",
      "dept": 3041,
      "class": 1,
      "subclass": 1,
      "itemDescription": "Placeholder item",
      "itemSuppliers": [
        {
          "supplier": 2400
        }
      ],
      "itemUdas": [
        {
          "udaId": 25021,
          "udaValue": "10"
        }
      ],
      "itemAggregateInd": "Y",
      "diff1AggregateInd": "Y",
      "brand": null
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",

```

```

        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Item Sourcing Upload Service

The following services are included in this functional area:

- [Create Item Supplier Countries](#)
- [Create Item Supplier Countries of Manufacture](#)
- [Create Item Supplier Country Dimensions](#)
- [Create Item Supplier Country Locations](#)
- [Create Item Supplier Translations](#)
- [Create Item Suppliers](#)
- [Delete Item Supplier Countries](#)
- [Delete Item Supplier Countries of Manufacture](#)
- [Delete Item Supplier Country Dimensions](#)
- [Delete Item Supplier Country Locations](#)
- [Delete Item Supplier Translations](#)
- [Delete Item Suppliers](#)
- [Update Item Supplier Countries](#)
- [Update Item Supplier Countries of Manufacture](#)
- [Update Item Supplier Country Dimensions](#)
- [Update Item Supplier Country Locations](#)
- [Update Item Supplier Translations](#)
- [Update Item Suppliers](#)

Create Item Supplier Countries

Functional Area

Items - Item Definition

Business Overview

This service is used to add country information to existing item supplier in Merchandising. The item supplier must already exist in Merchandising, else an error will be returned. Item supplier country can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/supplier/countries/create

Input Payload Details

Table 5-535 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-536 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-537 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.

Table 5-538 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
primaryCountryInd	No	String (1)	This indicates if the country is the primary country for the item supplier. Valid values are Y (yes) and N (no).

Table 5-538 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
unitCost	No	Number (20,4)	The current corporate unit cost for the item in the supplier's currency. This field will only be used to establish the item supplier country unit cost; it cannot be modified.
leadTime	No	Number (4)	This field contains the number of days that will elapse between the date an order is written and the delivery to the store or warehouse from the supplier. This field is defaulted from the default lead time set at the supplier level.
pickupLeadTime	No	Number (4)	Contains time it takes to get the item from the Supplier to the Initial Receiving Location. This value will be defaulted to the item_supp_country_loc pickup lead time field. The ordering dialog will reference the item/supplier/country/location pickup lead time as the value may vary by location.
mininumOrderQuantity	No	Number (12,4)	This field contains the minimum quantity that can be ordered at one time from the supplier for the item.
maximumOrderQuantity	No	Number (12,4)	This field contains the maximum quantity that can be ordered at one time from the supplier for the item.
supplierHierarchyLevel1	No	String (10)	Highest level of supplier hierarchy (e. g. manufacturer). This will be used for assigning rebates that come from a level in the supply chain that is higher than the supplier. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
supplierHierarchyLevel2	No	String (10)	Second highest level of supplier hierarchy. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
defaultUop	No	String (6)	Contains the default unit of purchase for the item/supplier/country. Valid values include: Standard Units of Measure C for Case P for Pallet
supplierPackSize	No	Number (12,4)	Contains the quantity that orders must be placed in multiples of for the supplier for the item.
innerPackSize	No	Number (12,4)	This field contains the break pack size for this item from the supplier.

Table 5-538 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ti	No	Number (12)	Number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of units (cases) for a pallet.
hi	No	Number (12)	Number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of units (cases) for a pallet.
costUom	No	String (4)	Unit of measure associated with the item cost. It is held to allow costs to be managed in a separate UOM from the standard UOM.
toleranceType	No	String (6)	The unit of the tolerances for catch weight simple packs: A - actual, P - percent.
minimumTolerance	No	Number (12,4)	The minimum tolerance value for a catch weight simple pack.
maximumTolerance	No	Number (12,4)	The maximum tolerance value for a catch weight simple pack.
supplierHierarchyType1	No	String (6)	Identifies partner type of supplier hierarchy level 1 for the foreign key to the partner table.
supplierHierarchyType2	No	String (6)	Identifies partner type of supplier hierarchy level 2 for the foreign key to the partner table.
supplierHierarchyType3	No	String (6)	Identifies partner type of supplier hierarchy level 3 for the foreign key to the partner table.
roundLevel	No	String (6)	This column will be used to determine how order quantities will be rounded to Case, Layer and Pallet.
roundToInnerPercentage	No	Number (12,4)	The Inner Rounding Threshold value. Used to determine whether to round partial Inner quantities up or down.
roundToCasePercentage	No	Number (12,4)	The Case Rounding Threshold value. Used to determine whether to round partial Case quantities up or down.
roundToLayerPercentage	No	Number (12,4)	The Layer Rounding Threshold value. Used to determine whether to round partial Layer quantities up or down.
roundToPalletPercentage	No	Number (12,4)	The Pallet Rounding Threshold value. Used to determine whether to round partial Pallet quantities up or down.
packingMethod	No	String (6)	This field indicates whether the packing method of the item in the container is Flat or Hanging. Values for this field are defined in PKMT code type. Examples are: FLAT - Flat, HANG - Hanging.

Table 5-538 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
defaultExpenseProfilesInd	No	String (1)	This field indicates whether expense profiles will be defaulted to new item-supplier-country relationship.
purchaseType	No	String (1)	This contains a code to indicate whether the item is normal merchandise (i. e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This indicates if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.
pricingControl	No	String (1)	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: R – Retailer S – Supplier If no value is provided for consignment/concession items, this will be defaulted based on the default pricing control at supplier site level.
dimension	No	Collection of Object	References a collection of item supplier country dimensions.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-539 Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	Yes	String (6)	Specific object whose dimensions are specified in this record (e. g. case, pallet, each). Valid codes are defined in the DIMO code type. Examples are: CA - Case, EA - Each.
tareWeight	No	Number (12,4)	Amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dimensionObject.
tareType	No	String (6)	Indicates if tare weight for this dimensionObject is wet or dry. Valid values are defined in the TARE code type. Examples are: W - Wet tare weight, D - Dry tare weight.
lwhUom	No	String (4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
length	No	Number (12,4)	Length of dim_object measured in units specified in lwh_uom.
width	No	Number (12,4)	Width of dim_object measured in units specified in lwh_uom.
height	No	Number (12,4)	Height of dim_object measured in units specified in lwh_uom.
liquidVolume	No	Number (12,4)	Liquid volume, or capacity, of dim_object measured in units specified in volume_uom. Liquid volumes are only convertible to other liquid volumes.
liquidVolumeUom	No	String (4)	Unit of measurement for liquid_volume (e. g. ounces, liters). Liquid volumes are only convertible to other liquid volumes. Valid values for this field are contained in uom field on uom_class table where uom_class field = LVOL.
statisticalCase	No	String (12)	Statistical value of the dim_objects dimensions to be used for loading purposes.
weightUom	No	String (4)	Unit of measurement for weight (e. g. pounds, kilograms). Valid values for this field are contained in uom field on uom_class table where uom_class field = MASS.
weight	No	Number (12,4)	Weight of dim_object measured in units specified in weight_uom.

Table 5-539 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
netWeight	No	Number (12,4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
presentationMethod	No	String (6)	Describes the packaging (if any) being taken into consideration in the specified dimensions. Valid codes are defined in the PCKT code type. Examples are: JHOOK - Peggable Packaging, STACK - Stackable Packaging.

Table 5-540 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-541 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-542 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",

```

```

"supplier": [
  {
    "supplier": "2400",
    "countryOfSourcing": [
      {
        "originCountry": "US",
        "primaryCountryInd": "Y",
        "unitCost": 9.65,
        "leadTime": 1,
        "pickupLeadTime": 1,
        "mininumOrderQuantity": 1,
        "maximumOrderQuantity": 1,
        "supplierHierarchyLevel1": null,
        "supplierHierarchyLevel2": null,
        "supplierHierarchyLevel3": null,
        "defaultUop": null,
        "supplierPackSize": 1,
        "innerPackSize": 1,
        "ti": 1,
        "hi": 1,
        "costUom": "EA",
        "toleranceType": null,
        "minimumTolerance": 76.00,
        "maximumTolerance": 69.00,
        "supplierHierarchyType1": null,
        "supplierHierarchyType2": null,
        "supplierHierarchyType3": null,
        "roundLevel": "C",
        "roundToInnerPercentage": 50,
        "roundToCasePercentage": 50,
        "roundToLayerPercentage": 50,
        "roundToPalletPercentage": 50,
        "packingMethod": "HANG",
        "defaultExpenseProfilesInd": "Y",
        "purchaseType": "0",
        "calculationBasis": null,
        "purchaseRate": 18.00,
        "pricingControl": "R",
        "dimension": [
          {
            "dimensionObject": "CA",
            "tareWeight": 3,
            "tareType": "D",
            "lwhUom": "IN",
            "length": 3,
            "width": 3,
            "height": 3,
            "liquidVolume": 3,
            "liquidVolumeUom": "EA",
            "statisticalCase": null,
            "weightUom": "EA",
            "weight": 3,
            "netWeight": 11.00,
            "presentationMethod": null
          }
        ]
      }
    ]
  },
],

```

```

        "customFlexAttribute": [
          {
            "name": "REVIEW_DATE",
            "value": null,
            "valueDate": "2001-12-31"
          }
        ],
        "localizationExtensions": [
          {
            "country": "BR",
            "attributes": [
              {
                "name": "SPECIAL_CATEGORY_CODE",
                "value": "WS_PRC_08",
                "valueDate": "2001-12-31"
              }
            ]
          }
        ]
      }
    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item Supplier Countries of Manufacture

Functional Area

Items - Item Definition

Business Overview

This service is used to add item supplier manufacturing country to an existing item supplier in Merchandising. The item supplier must already exist in Merchandising, else an error will be returned. Item supplier manufacturing country can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/supplier/countriesOfManufacture/create

Input Payload Details

Table 5-543 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-544 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.

Table 5-544 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-545 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfManufacture	No	Collection of Object	References a collection of country of manufacture of the item.

Table 5-546 CountryOfManufacture - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manufacturerCountry	Yes	String (23)	Country ID of manufacturer.
primaryManufacturerCountryInd	No	String (1)	Indicates if it's the primary country of manufacturer.

Table 5-547 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-548 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100300001",
```

```

    "dataLoadingDestination": "RMS",
    "supplier": [
      {
        "supplier": "2400",
        "countryOfManufacture": [
          {
            "manufacturerCountry": "BR",
            "primaryManufacturerCountryInd": "N"
          }
        ]
      }
    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Item Supplier Country Dimensions

Functional Area

Items - Item Definition

Business Overview

This service is used to add dimensions to an existing item supplier country in Merchandising. Dimensions are not required for an item. However, if non-standard units of measure will be used for the item there must be a case-type dimension provided. For example, for a selling unit retail to be defined in ounces there must be a dimension defined containing the liquid and volume to be used when converting between a unit and an ounce. The item supplier country must already exist in Merchandising, else an error will be returned. Item supplier country dimension can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/supplier/country/dimensions/create

Input Payload Details

Table 5-549 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-550 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-551 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.

Table 5-552 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
dimension	No	Collection of Object	References a collection of item supplier country dimensions.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-553 Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	Yes	String (6)	Specific object whose dimensions are specified in this record (e. g. case, pallet, each). Valid codes are defined in the DIMO code type. Examples are: CA - Case, EA - Each.
tareWeight	No	Number (12,4)	Amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dimensionObject.
tareType	No	String (6)	Indicates if tare weight for this dimensionObject is wet or dry. Valid values are defined in the TARE code type. Examples are: W - Wet tare weight, D - Dry tare weight.
lwhUom	No	String (4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
length	No	Number (12,4)	Length of dim_object measured in units specified in lwh_uom.
width	No	Number (12,4)	Width of dim_object measured in units specified in lwh_uom.
height	No	Number (12,4)	Height of dim_object measured in units specified in lwh_uom.

Table 5-553 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
liquidVolume	No	Number (12,4)	Liquid volume, or capacity, of dim_object measured in units specified in volume_uom. Liquid volumes are only convertible to other liquid volumes.
liquidVolumeUom	No	String (4)	Unit of measurement for liquid_volume (e. g. ounces, liters). Liquid volumes are only convertible to other liquid volumes. Valid values for this field are contained in uom field on uom_class table where uom_class field = LVOL.
statisticalCase	No	String (12)	Statistical value of the dim_objects dimensions to be used for loading purposes.
weightUom	No	String (4)	Unit of measurement for weight (e. g. pounds, kilograms). Valid values for this field are contained in uom field on uom_class table where uom_class field = MASS.
weight	No	Number (12,4)	Weight of dim_object measured in units specified in weight_uom.
netWeight	No	Number (12,4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
presentationMethod	No	String (6)	Describes the packaging (if any) being taken into consideration in the specified dimensions. Valid codes are defined in the PKCT code type. Examples are: JHOOK - Peggable Packaging, STACK - Stackable Packaging.

Table 5-554 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-555 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.

Table 5-555 (Cont.) Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "item": "102250001",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US",
              "dimension": [
                {
                  "dimensionObject": "CA",
                  "tareWeight": 3,
                  "tareType": "D",
                  "lwhUom": "IN",
                  "length": 3,
                  "width": 3,
                  "height": 3,
                  "liquidVolume": 3,
                  "liquidVolumeUom": "EA",
                  "statisticalCase": null,
                  "weightUom": "EA",
                  "weight": 3,
                  "netWeight": 11.00,
                  "presentationMethod": null
                }
              ]
            }
          ],
          "localizationExtensions": [
            {
              "country": "BR",
              "attributes": [
                {
                  "name": "SPECIAL_CATEGORY_CODE",
                  "value": "WS_PRC_08",
                  "valueDate": "2001-12-31"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}

```

```

    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Item Supplier Country Locations

Functional Area

Items - Item Definition

Business Overview

This service is used to add location information to an existing item supplier country in Merchandising. Records are not required at this level for an item however, if provided the values override those defined at the item/supplier/country level for the specified location(s). The item supplier country must already exist in Merchandising, else an error will be returned.

Item supplier country location can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/supplier/country/locations/create

Input Payload Details

Table 5-556 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-557 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
itemSuppCountryLocHierarchyLevel	No	String (2)	The level of the organizational hierarchy used to create item supplier country location relationships. Valid values are: 'AS' - All Store, 'S' - Store, 'DW' - Default Warehouse, 'AW' - All Warehouses, 'W' - Warehouse, 'PW' - Physical Warehouse, 'D' - District, 'R' - Region, 'A' - Area, 'T' - Transfer zone, 'L' - Location trait.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-558 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.

Table 5-558 (Cont.) Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.

Table 5-559 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
location	No	Collection of Object	References a collection of item supplier country locations.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-560 Location - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyId	No	Number (10)	The specific location id for this Item Supplier Country Loc message. Corresponds with the value used in the isclloc_hier_level field.
unitCost	No	Number (20,4)	The unit cost for the item at the item supplier country loc level in the supplier's currency. This field will be used for creating new records, not updating existing records.
negotiatedItemCost	No	Number (20,4)	This will hold the supplier negotiated item cost.
pickupLeadTime	No	Number (4)	Contains time it takes to get the item from the Supplier to the Initial Receiving Location. This value will be defaulted to the item_supp_country_loc pickup lead time field. The ordering dialog will reference the item/supplier/country/location pickup lead time as the value may vary by location.
roundLevel	No	String (6)	This column will be used to determine how order quantities will be rounded to Case, Layer and Pallet.
roundToCasePercentage	No	Number (12,4)	The Case Rounding Threshold value. Used to determine whether to round partial case quantities up or down.
roundToLayerPercentage	No	Number (12,4)	The Layer Rounding Threshold value. Used to determine whether to round partial Layer quantities up or down.

Table 5-560 (Cont.) Location - Object. See list of elements for detail

Element Name	Required	Data Type	Description
roundToPalletPercentage	No	Number (12,4)	The Pallet Rounding Threshold value. Used to determine whether to round partial Pallet quantities up or down.
roundToInnerPercentage	No	Number (12,4)	The Inner Rounding Threshold value. Used to determine whether to round partial Inner quantities up or down.
supplierHierarchyLevel1	No	String (10)	Highest level of supplier hierarchy (e. g. manufacturer). This will be used for assigning rebates that come from a level in the supply chain that is higher than the supplier.
supplierHierarchyLevel2	No	String (10)	Second highest level of supplier hierarchy. Rebates at this level will include all eligible supplier/item/country/loc records assigned to this supplier hierarchy level.
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy, used for assigning rebates by a level other than supplier.
costUom	No	String (4)	Unit of measure associated with the item cost. It is held to allow costs to be managed in a separate UOM from the standard UOM.
purchaseType	No	String (1)	This contains a code to indicate whether the item is normal merchandise (i. e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This indicates if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.

Table 5-560 (Cont.) Location - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pricingControl	No	String (1)	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: • R – Retailer • S – Supplier If no value is provided for consignment/concession items, this will be defaulted based on the default pricing control at supplier site level.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named, flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-561 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-562 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-563 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100041",
      "itemSuppCountryLocHierarchyLevel": "S",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US",
              "location": [
                {
                  "hierarchyId": 1521,
                  "unitCost": 9.65,
                  "negotiatedItemCost": 9.65,
                  "pickupLeadTime": 1,
                  "roundLevel": "C",
                  "roundToCasePercentage": 50,
                  "roundToLayerPercentage": 50,
                  "roundToPalletPercentage": 50,
                  "roundToInnerPercentage": 50,
                  "supplierHierarchyLevel1": null,
                  "supplierHierarchyLevel2": null,
                  "supplierHierarchyLevel3": null,
                  "costUom": "EA",
                  "purchaseType": "0",
                  "calculationBasis": null,
                  "purchaseRate": 3,
                  "pricingControl": "R",
                  "customFlexAttribute": [
                    {
                      "name": "REVIEW_DATE",
                      "value": null,
                      "valueDate": "2001-12-31"
                    }
                  ]
                }
              ]
            }
          ]
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}

```

```

    }
  ]
}
],
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]
}
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Item Supplier Translations

Functional Area

Items - Item Definition

Business Overview

If translations have been defined for items, at the supplier level, then they can be integrated as part of this service. This service will accept the language and the description specified in the language to add supplier level translations to an existing item in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/item/supplier/translations/create

Input Payload Details

Table 5-564 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-565 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-566 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
translation	No	Collection of Object	References a collection of translations of item supplier description fields.

Table 5-567 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This is the language ID.

Table 5-567 (Cont.) Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierLabel	No	String (15)	The supplier label for the item.

Table 5-568 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-569 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "translation": [
```

```

        {
          "language": 13,
          "supplierDiff1": "1",
          "supplierDiff2": "2",
          "supplierDiff3": "3",
          "supplierDiff4": "4",
          "supplierLabel": ""
        }
      ]
    },
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Item Suppliers

Functional Area

Items - Item Definition

Business Overview

This service is used to add item suppliers to an existing item in Merchandising. The item must already exist in Merchandising, else an error will be returned. Item supplier can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/suppliers/create

Input Payload Details

Table 5-570 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-571 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-572 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
primarySupplierInd	No	String (3)	Indicates if the supplier is the primary supplier for the item. Valid values are Y (yes) and N (no).
vpn	No	String (30)	The vendor product number associated with the item.
supplierLabel	No	String (15)	The supplier label for the item.
consignmentRate	No	Number (12,4)	This field contains the consignment rate for this item for the supplier.
supplierDiscontinueDate	No	date	Date which the supplier discontinues an item. The retailer should be aware that the supplier is able to reuse a UPC after 30 months and should work to ensure that no data exists in RMS for a UPC 30 months after it has been discontinued.
directShipInd	No	String (1)	Contains a value of Yes to indicate that any item associated with this supplier is eligible for a direct shipment from the supplier to the customer.
palletName	No	String (6)	Code referencing the name used to refer to the pallet. Valid codes are defined in the PALN code type. Examples are FLA - Flat, PAL - Pallet.

Table 5-572 (Cont.) Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
caseName	No	String (6)	Code referencing the name used to refer to the case. Valid codes are defined in the CASN code type. Examples are PACK - Pack, BX - Box, BG - Bag.
innerName	No	String (6)	Code referencing the name used to refer to the inner. Valid codes are defined in the INRN code type. Examples are SCS - Sub-Case, SPACK - Sub-Pack.
primaryCaseSize	No	String (6)	Used only if AIP is interfaced. Indicates the primary case size for the item supplier when an orderable item is configured for informal case types.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description. This field may only contain a value for items with an item_parent.

Table 5-572 (Cont.) Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
concessionRate	No	Number (12,4)	The concession rate is the margin that a particular supplier receives for the sale of a concession item.
defaultExpenseProfiles Ind	No	String (1)	This field indicates whether expense profiles will be defaulted to new item-supplier relationship. Valid values are Y (yes) and N (no).
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
translation	No	Collection of Object	References a collection of translations of item supplier description fields.
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.
countryOfManufactur e	No	Collection of Object	References a collection of country of manufacture of the item.
uom	No	Collection of Object	Child node holding miscellaneous units of measure (UOM) for an item/supplier combination.

Table 5-573 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
primaryCountryInd	No	String (1)	This indicates if the country is the primary country for the item supplier. Valid values are Y (yes) and N (no).
unitCost	No	Number (20,4)	The current corporate unit cost for the item in the supplier's currency. This field will only be used to establish the item supplier country unit cost; it cannot be modified.
leadTime	No	Number (4)	This field contains the number of days that will elapse between the date an order is written and the delivery to the store or warehouse from the supplier. This field is defaulted from the default lead time set at the supplier level.
pickupLeadTime	No	Number (4)	Contains time it takes to get the item from the Supplier to the Initial Receiving Location. This value will be defaulted to the item_supp_country_loc pickup lead time field. The ordering dialog will reference the item/supplier/country/location pickup lead time as the value may vary by location.
mininumOrderQuantity	No	Number (12,4)	This field contains the minimum quantity that can be ordered at one time from the supplier for the item.

Table 5-573 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
maximumOrderQuantity	No	Number (12,4)	This field contains the maximum quantity that can be ordered at one time from the supplier for the item.
supplierHierarchyLevel1	No	String (10)	Highest level of supplier hierarchy (e. g. manufacturer). This will be used for assigning rebates that come from a level in the supply chain that is higher than the supplier. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
supplierHierarchyLevel2	No	String (10)	Second highest level of supplier hierarchy. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
defaultUop	No	String (6)	Contains the default unit of purchase for the item/supplier/country. Valid values include: Standard Units of Measure C for Case P for Pallet
supplierPackSize	No	Number (12,4)	Contains the quantity that orders must be placed in multiples of for the supplier for the item.
innerPackSize	No	Number (12,4)	This field contains the break pack size for this item from the supplier.

Table 5-573 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ti	No	Number (12)	Number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of units (cases) for a pallet.
hi	No	Number (12)	Number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of units (cases) for a pallet.
costUom	No	String (4)	Unit of measure associated with the item cost. It is held to allow costs to be managed in a separate UOM from the standard UOM.
toleranceType	No	String (6)	The unit of the tolerances for catch weight simple packs: A - actual, P - percent.
minimumTolerance	No	Number (12,4)	The minimum tolerance value for a catch weight simple pack.
maximumTolerance	No	Number (12,4)	The maximum tolerance value for a catch weight simple pack.
supplierHierarchyType1	No	String (6)	Identifies partner type of supplier hierarchy level 1 for the foreign key to the partner table.
supplierHierarchyType2	No	String (6)	Identifies partner type of supplier hierarchy level 2 for the foreign key to the partner table.
supplierHierarchyType3	No	String (6)	Identifies partner type of supplier hierarchy level 3 for the foreign key to the partner table.
roundLevel	No	String (6)	This column will be used to determine how order quantities will be rounded to Case, Layer and Pallet.

Table 5-573 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
roundToInnerPercentage	No	Number (12,4)	The Inner Rounding Threshold value. Used to determine whether to round partial Inner quantities up or down.
roundToCasePercentage	No	Number (12,4)	The Case Rounding Threshold value. Used to determine whether to round partial Case quantities up or down.
roundToLayerPercentage	No	Number (12,4)	The Layer Rounding Threshold value. Used to determine whether to round partial Layer quantities up or down.
roundToPalletPercentage	No	Number (12,4)	The Pallet Rounding Threshold value. Used to determine whether to round partial Pallet quantities up or down.
packingMethod	No	String (6)	This field indicates whether the packing method of the item in the container is Flat or Hanging. Values for this field are defined in PKMT code type. Examples are: FLAT - Flat, HANG - Hanging.
defaultExpenseProfilesInd	No	String (1)	This field indicates whether expense profiles will be defaulted to new item-supplier-country relationship.
purchaseType	No	String (1)	This contains a code to indicate whether the item is normal merchandise (i. e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.

Table 5-573 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
calculationBasis	No	String (1)	This indicates if the cost for the consignment/ concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.
pricingControl	No	String (1)	This indicates whether the price for the consignment/ concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/ Concession ownership model is used within the Merchandising suite of solutions. Valid values are: R - Retailer, S - Supplier. If no value is provided for consignment/ concession items, this will be defaulted based on the default pricing control at supplier site level.
dimension	No	Collection of Object	References a collection of item supplier country dimensions.

Table 5-573 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-574 Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	Yes	String (6)	Specific object whose dimensions are specified in this record (e. g. case, pallet, each). Valid codes are defined in the DIMO code type. Examples are: CA - Case, EA - Each.
tareWeight	No	Number (12,4)	Amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dimensionObject.
tareType	No	String (6)	Indicates if tare weight for this dimensionObject is wet or dry. Valid values are defined in the TARE code type. Examples are: W - Wet tare weight, D - Dry tare weight.

Table 5-574 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lwhUom	No	String (4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
length	No	Number (12,4)	Length of dim_object measured in units specified in lwh_uom.
width	No	Number (12,4)	Width of dim_object measured in units specified in lwh_uom.
height	No	Number (12,4)	Height of dim_object measured in units specified in lwh_uom.
liquidVolume	No	Number (12,4)	Liquid volume, or capacity, of dim_object measured in units specified in volume_uom. Liquid volumes are only convertible to other liquid volumes.
liquidVolumeUom	No	String (4)	Unit of measurement for liquid_volume (e. g. ounces, liters). Liquid volumes are only convertible to other liquid volumes. Valid values for this field are contained in uom field on uom_class table where uom_class field = LVOL.
statisticalCase	No	String (12)	Statistical value of the dim_objects dimensions to be used for loading purposes.
weightUom	No	String (4)	Unit of measurement for weight (e. g. pounds, kilograms). Valid values for this field are contained in uom field on uom_class table where uom_class field = MASS.

Table 5-574 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
weight	No	Number (12,4)	Weight of dim_object measured in units specified in weight_uom.
netWeight	No	Number (12,4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
presentationMethod	No	String (6)	Describes the packaging (if any) being taken into consideration in the specified dimensions. Valid codes are defined in the PCKT code type. Examples are: JHOOK - Peggable Packaging, STACK - Stackable Packaging.

Table 5-575 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-576 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-577 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-578 CountryOfManufacture - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manufacturerCountry	Yes	String (23)	Country ID of manufacturer.
primaryManufacturerCountryInd	No	String (1)	Indicates if it's the primary country of manufacturer.

Table 5-579 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This is the language ID.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description. This field may only contain a value for items with an item_parent.

Table 5-579 (Cont.) Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierLabel	No	String (15)	The supplier label for the item.

Table 5-580 Uom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uom	Yes	String (4)	Contains the unit of measure that the item is being measured in.
value	Yes	Number (20,4)	Contains the equivalent value of the Item/Suppliers shipping carton in the associated unit of measure.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "primarySupplierInd": "Y",
          "vpn": null,
          "supplierLabel": null,
          "consignmentRate": 3,
          "supplierDiscontinueDate": "2001-12-31",
          "directShipInd": "Y",
          "palletName": "PAL",
          "caseName": "CS",
          "innerName": "INR",
          "primaryCaseSize": null,
          "supplierDiff1": null,
          "supplierDiff2": null,
          "supplierDiff3": null,
          "supplierDiff4": null,
          "concessionRate": 1,
          "defaultExpenseProfilesInd": "Y",

```

```

"customFlexAttribute": [
{
"name": "REVIEW_DATE",
"value": null,
"valueDate": "2001-12-31"
}
],
"translation": [
{
"language": 2,
"supplierDiff1": null,
"supplierDiff2": null,
"supplierDiff3": null,
"supplierDiff4": null,
"supplierLabel": null
}
],
"countryOfSourcing": [
{
"originCountry": "US",
"primaryCountryInd": "Y",
"unitCost": 9.65,
"leadTime": 1,
"pickupLeadTime": 1,
"mininumOrderQuantity": 1,
"maximumOrderQuantity": 1,
"supplierHierarchyLevel1": null,
"supplierHierarchyLevel2": null,
"supplierHierarchyLevel3": null,
"defaultUop": "EA",
"supplierPackSize": 1,
"innerPackSize": 1,
"ti": 1,
"hi": 1,
"costUom": "EA",
"toleranceType": null,
"minimumTolerance": 76.00,
"maximumTolerance": 69.00,
"supplierHierarchyType1": null,
"supplierHierarchyType2": null,
"supplierHierarchyType3": null,
"roundLevel": "C",
"roundToInnerPercentage": 50,
"roundToCasePercentage": 50,
"roundToLayerPercentage": 50,
"roundToPalletPercentage": 50,
"packingMethod": "HANG",
"defaultExpenseProfilesInd": "Y",
"purchaseType": "0",
"calculationBasis": null,
"purchaseRate": 18.00,
"pricingControl": "R",
"dimension": [
{
"dimensionObject": "PA",
"tareWeight": 17.00,

```

```

    "tareType": "D",
    "lwhUom": "IN",
    "length": 3,
    "width": 3,
    "height": 3,
    "liquidVolume": 3,
    "liquidVolumeUom": null,
    "statisticalCase": null,
    "weightUom": "LBS",
    "weight": 3,
    "netWeight": 11.00,
    "presentationMethod": null
  },
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": "2001-12-31"
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ],
  "countryOfManufacture": [
    {
      "manufacturerCountry": "BR",
      "primaryManufacturerCountryInd": "N"
    }
  ],
  "uom": [
    {
      "uom": "STK",
      "value": 10
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",

```

```
"valueDate": "2001-12-31"
}
]
}
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item Supplier UOM

Functional Area

Items - Item Definition

Business Overview

Service to create miscellaneous units of measure (UOM) for an item/supplier combination. Supplier UOM can also be added together with the creation of a new item. This UOM indicates the equivalent of one unit of the item in the specified UOM.

For example, if the item is a single package of cigars, and the single package sold in stores contains 10 cigar 'sticks'. The standard UOM of the item will be EA, and one Each of the item is a package of 10 cigar sticks. The retailer can create a MISC UOM of 'Sticks' and in Item Supplier UOM, the UOM would be set to 'Sticks' and a Value = 10 because there are 10 sticks in a single unit of the item. This definition will allow expense, assessment and/or up charge components to be created to calculate a tax. So, if the tax is 0.13 USD per stick, then the system will use the item supplier UOM record for Sticks to determine that 1 unit of the item is 10 sticks and will therefore calculate the cost component as 0.13 x 10 sticks = 1.30 USD for one package of cigars.

Service Type
POST

ReST URL

MerchIntegrations/services/item/supplier/uom/create

Input Payload Details

Table 5-581 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-582 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.

Table 5-583 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
uom	No	Collection of Object	Child node holding miscellaneous units of measure (UOM) for an item/supplier combination.

Table 5-584 Uom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uom	Yes	String (4)	Contains the unit of measure that the item is being measured in.
value	Yes	Number (20,4)	Contains the equivalent value of the Item/Suppliers shipping carton in the associated unit of measure.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "uom": [
            {
              "uom": "STK",
              "value": 10
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
```



```

"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
]
}

```

Delete Item Supplier Countries

Functional Area

Items - Item Definition

Business Overview

This service is used to delete existing item/supplier/country level information from Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/supplier/countries/delete

Input Payload Details

Table 5-585 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-586 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
supplier	No	Collection of Object	References a collection of item supplier.

Table 5-587 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.
countryOfSourcing	No	Collection of Object	References a collection of item supplier country information.

Table 5-588 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The supplier country affected by the deletion activity.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

```
} ]
```

Delete Item Supplier Countries of Manufacture

Functional Area

Items - Item Definition

Business Overview

This service is used to delete an existing item supplier country of manufacturing information from Merchandising. In order to delete an item supplier manufacturing country, the supplier manufacturing country must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/supplier/countriesOfManufacture/delete

Input Payload Details

Table 5-589 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-590 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
supplier	No	Collection of Object	References a collection of item supplier.

Table 5-591 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.
countryOfManufacture	No	Collection of Object	References a collection of country of manufacture of the item.

Table 5-592 CountryOfManufacture - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manufacturerCountry	Yes	String (23)	Country ID of manufacturer.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100000059",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfManufacture": [
            {
              "manufacturerCountry": "BR"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Supplier Country Dimensions

Functional Area

Items - Item Definition

Business Overview

This service is used to delete item supplier country dimensions in Merchandising. In order to delete an item supplier country dimension, the supplier country dimension must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/supplier/country/dimensions/delete

Input Payload Details

Table 5-593 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-594 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
supplier	No	Collection of Object	References a collection of item supplier.

Table 5-595 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.
countryOfSourcing	No	Collection of Object	References a collection of item supplier country information.

Table 5-596 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The supplier country affected by the deletion activity.
dimension	No	Collection of Object	References a collection of item supplier country dimension.

Table 5-597 Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	Yes	String (6)	The dimension object.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US",
              "dimension": [
                {
                  "dimensionObject": "CA"
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Supplier Country Locations

Functional Area

Items - Item Definition

Business Overview

This service is used to delete item supplier country locations from an existing item supplier country in Merchandising. To delete an item supplier country location, the supplier country location must exist for the item in Merchandising; otherwise, an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/supplier/country/locations/delete

Input Payload Details

Table 5-598 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-599 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
hierarchyLevel	No	String (2)	The level of the organizational hierarchy used to delete item supplier country location relationships. Valid values are: 'AS' - All Store, 'S' - Store, 'DW' - Default Warehouse, 'AW' - All Warehouses, 'W' - Warehouse, 'PW' - Physical Warehouse, 'AI' - All internal finishers, 'I' - Internal finisher, 'D' - District, 'R' - Region, 'A' - Area, 'T' - Transfer zone, 'L' - Location trait.
supplier	No	Collection of Object	References a collection of item supplier.

Table 5-600 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.
countryOfSourcing	No	Collection of Object	References a collection of item supplier country information.

Table 5-601 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The supplier country affected by the deletion activity.
location	No	Collection of Object	References a collection of item supplier country location.

Table 5-602 Location - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyId	Yes	Number (10)	The id of the location hierarchy level affected by the message.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100068",
      "hierarchyLevel": "S",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US",
              "location": [
                {
                  "hierarchyId": 1111
                }
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Supplier Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to delete supplier level translations from an existing item in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/supplier/translations/delete

Input Payload Details

Table 5-603 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-604 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
supplier	No	Collection of Object	References a collection of item supplier.

Table 5-605 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.
translation	No	Collection of Object	References a collection of translations of item supplier description fields.

Table 5-606 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	The language ID.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "supplier": [
        {
          "supplier": "2400",
          "translation": [
            {
              "language": 2
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
```

```

    "message": "Error found in validation of input payload",
    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}

```

Delete Item Suppliers

Functional Area

Items - Item Definition

Business Overview

This service is used to delete supplier details from existing items within Merchandising. In order to delete an item supplier, the specified supplier must exist for the item in Merchandising, else an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/suppliers/delete

Input Payload Details

Table 5-607 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-608 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
supplier	No	Collection of Object	References a collection of item supplier.

Table 5-609 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "supplier": [
        {
          "supplier": "2400"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Supplier UOM

Functional Area

Items - Item Definition

Business Overview

Service to delete miscellaneous units of measure (UOM) for an item/supplier combination.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/supplier/uom/delete

Input Payload Details

Table 5-610 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-611 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
supplier	Yes	Collection of Object	References a collection of item supplier.

Table 5-612 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	Item Supplier affected by the deletion activity.
uom	Yes	Collection of Object	Child node.

Table 5-613 Uom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uom	Yes	String (4)	Contains the unit of measure that the item is being measured in.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "supplier": [
        {
          "supplier": "2400",
          "uom": [
            {
              "uom": "STK"
            }
          ]
        }
      ]
    }
  ]
}
```

```
]
}
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Supplier Countries

Functional Area

Items - Item Definition

Business Overview

This service is used to modify existing item supplier country information in Merchandising. To update item supplier country information, the supplier country must exist for the item in Merchandising; otherwise, an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/supplier/countries/update

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-614 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-615 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-616 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.

Table 5-617 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
primaryCountryInd	No	String (1)	This indicates if the country is the primary country for the item supplier. Valid values are Y (yes) and N (no).
unitCost	No	Number (20,4)	The current corporate unit cost for the item in the supplier's currency. This field will only be used to establish the item supplier country unit cost; it cannot be modified.

Table 5-617 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
leadTime	No	Number (4)	This field contains the number of days that will elapse between the date an order is written and the delivery to the store or warehouse from the supplier. This field is defaulted from the default lead time set at the supplier level.
pickupLeadTime	No	Number (4)	Contains time it takes to get the item from the Supplier to the Initial Receiving Location. This value will be defaulted to the item_supp_country_loc pickup lead time field. The ordering dialog will reference the item/supplier/country/location pickup lead time as the value may vary by location.
mininumOrderQuantity	No	Number (12,4)	This field contains the minimum quantity that can be ordered at one time from the supplier for the item.
maximumOrderQuantity	No	Number (12,4)	This field contains the maximum quantity that can be ordered at one time from the supplier for the item.
supplierHierarchyLevel1	No	String (10)	Highest level of supplier hierarchy (e. g. manufacturer). This will be used for assigning rebates that come from a level in the supply chain that is higher than the supplier. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
supplierHierarchyLevel2	No	String (10)	Second highest level of supplier hierarchy. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy. This information is stored on item_supp_country for defaulting into item_supp_country_loc.
defaultUop	No	String (6)	Contains the default unit of purchase for the item/supplier/country. Valid values include: Standard Units of Measure C for Case P for Pallet
supplierPackSize	No	Number (12,4)	Contains the quantity that orders must be placed in multiples of for the supplier for the item.
innerPackSize	No	Number (12,4)	This field contains the break pack size for this item from the supplier.
ti	No	Number (12)	Number of shipping units (cases) that make up one tier of a pallet. Multiply TI x HI to get total number of units (cases) for a pallet.

Table 5-617 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hi	No	Number (12)	Number of tiers that make up a complete pallet (height). Multiply TI x HI to get total number of units (cases) for a pallet.
costUom	No	String (4)	Unit of measure associated with the item cost. It is held to allow costs to be managed in a separate UOM from the standard UOM.
toleranceType	No	String (6)	The unit of the tolerances for catch weight simple packs: A - actual, P - percent.
minimumTolerance	No	Number (12,4)	The minimum tolerance value for a catch weight simple pack.
maximumTolerance	No	Number (12,4)	The maximum tolerance value for a catch weight simple pack.
supplierHierarchyType1	No	String (6)	Identifies partner type of supplier hierarchy level 1 for the foreign key to the partner table.
supplierHierarchyType2	No	String (6)	Identifies partner type of supplier hierarchy level 2 for the foreign key to the partner table.
supplierHierarchyType3	No	String (6)	Identifies partner type of supplier hierarchy level 3 for the foreign key to the partner table.
roundLevel	No	String (6)	This column will be used to determine how order quantities will be rounded to Case, Layer and Pallet.
roundToInnerPercentage	No	Number (12,4)	The Inner Rounding Threshold value. Used to determine whether to round partial Inner quantities up or down.
roundToCasePercentage	No	Number (12,4)	The Case Rounding Threshold value. Used to determine whether to round partial Case quantities up or down.
roundToLayerPercentage	No	Number (12,4)	The Layer Rounding Threshold value. Used to determine whether to round partial Layer quantities up or down.
roundToPalletPercentage	No	Number (12,4)	The Pallet Rounding Threshold value. Used to determine whether to round partial Pallet quantities up or down.
packingMethod	No	String (6)	This field indicates whether the packing method of the item in the container is Flat or Hanging. Values for this field are defined in PKMT code type. Examples are: FLAT - Flat, HANG - Hanging.
defaultExpenseProfilesInd	No	String (1)	This field indicates whether expense profiles will be defaulted to new item-supplier-country relationship.

Table 5-617 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	String (1)	This contains a code to indicate whether the item is normal merchandise (i. e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This indicates if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.
pricingControl	No	String (1)	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: R – Retailer S – Supplier. If no value is provided for consignment/concession items, this will be defaulted based on the default pricing control at supplier site level.
dimension	No	Collection of Object	References a collection of item supplier country dimensions.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-618 Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	Yes	String (6)	Specific object whose dimensions are specified in this record (e. g. case, pallet, each). Valid codes are defined in the DIMO code type. Examples are: CA - Case, EA - Each.

Table 5-618 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
tareWeight	No	Number (12,4)	Amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dimensionObject.
tareType	No	String (6)	Indicates if tare weight for this dimensionObject is wet or dry. Valid values are defined in the TARE code type. Examples are: W - Wet tare weight, D - Dry tare weight.
lwhUom	No	String (4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
length	No	Number (12,4)	Length of dim_object measured in units specified in lwh_uom.
width	No	Number (12,4)	Width of dim_object measured in units specified in lwh_uom.
height	No	Number (12,4)	Height of dim_object measured in units specified in lwh_uom.
liquidVolume	No	Number (12,4)	Liquid volume, or capacity, of dim_object measured in units specified in volume_uom. Liquid volumes are only convertible to other liquid volumes.
liquidVolumeUom	No	String (4)	Unit of measurement for liquid_volume (e. g. ounces, liters). Liquid volumes are only convertible to other liquid volumes. Valid values for this field are contained in uom field on uom_class table where uom_class field = LVOL.
statisticalCase	No	String (12)	Statistical value of the dim_objects dimensions to be used for loading purposes.
weightUom	No	String (4)	Unit of measurement for weight (e. g. pounds, kilograms). Valid values for this field are contained in uom field on uom_class table where uom_class field = MASS.
weight	No	Number (12,4)	Weight of dim_object measured in units specified in weight_uom.
netWeight	No	Number (12,4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.

Table 5-618 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
presentationMethod	No	String (6)	Describes the packaging (if any) being taken into consideration in the specified dimensions. Valid codes are defined in the PCKT code type. Examples are: JHOOK - Peggable Packaging, STACK - Stackable Packaging.

Table 5-619 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-620 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-621 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US",
```

```

"primaryCountryInd": "Y",
"unitCost": 9.65,
"leadTime": 1,
"pickupLeadTime": 1,
"mininumOrderQuantity": 1,
"maximumOrderQuantity": 1,
"supplierHierarchyLevel1": null,
"supplierHierarchyLevel2": null,
"supplierHierarchyLevel3": null,
"defaultUop": null,
"supplierPackSize": 1,
"innerPackSize": 1,
"ti": 1,
"hi": 1,
"costUom": "EA",
"toleranceType": null,
"minimumTolerance": 76.00,
"maximumTolerance": 69.00,
"supplierHierarchyType1": null,
"supplierHierarchyType2": null,
"supplierHierarchyType3": null,
"roundLevel": "C",
"roundToInnerPercentage": 50,
"roundToCasePercentage": 50,
"roundToLayerPercentage": 50,
"roundToPalletPercentage": 50,
"packingMethod": "HANG",
"defaultExpenseProfilesInd": "Y",
"purchaseType": "0",
"calculationBasis": null,
"purchaseRate": 18.00,
"pricingControl": "R",
"dimension": [
  {
    "dimensionObject": "CA",
    "tareWeight": 3,
    "tareType": "D",
    "lwhUom": "IN",
    "length": 3,
    "width": 3,
    "height": 3,
    "liquidVolume": 3,
    "liquidVolumeUom": "EA",
    "statisticalCase": null,
    "weightUom": "EA",
    "weight": 3,
    "netWeight": 11.00,
    "presentationMethod": null
  }
],
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
]

```

```

    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
],
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]
}
]
}
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
    }
  ]
}

```

```

        "inputValue": "X"
      }
    ]
  }

```

Update Item Supplier Countries of Manufacture

Functional Area

Items - Item Definition

Business Overview

This service is used to modify manufacturing country of an existing item supplier in Merchandising. To update item supplier country manufacturing information, the supplier manufacturing country must exist for the item in Merchandising; otherwise, an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/supplier/countriesOfManufacture/update

Input Payload Details

Table 5-622 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-623 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-624 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfManufacture	No	Collection of Object	References a collection of country of manufacture of the item.

Table 5-625 CountryOfManufacture - Object. See list of elements for detail

Element Name	Required	Data Type	Description
manufacturerCountry	Yes	String (23)	Country ID of manufacturer.
primaryManufacturerCountryInd	No	String (1)	Indicates if it's the primary country of manufacturer.

Table 5-626 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-627 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100600020",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfManufacture": [
            {
              "manufacturerCountry": "BR",
              "primaryManufacturerCountryInd": "N"
            }
          ]
        }
      ]
    }
  ]
}
```



```

    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Update Item Supplier Country Dimensions

Functional Area

Items - Item Definition

Business Overview

This service is used to modify dimensions for an existing item supplier country in Merchandising. To update item supplier country dimension information, the supplier country dimension must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/supplier/country/dimensions/update

Input Payload Details

Table 5-628 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-629 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-630 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.

Table 5-631 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
dimension	No	Collection of Object	References a collection of item supplier country dimensions.

Table 5-631 (Cont.) CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-632 Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dimensionObject	Yes	String (6)	Specific object whose dimensions are specified in this record (e. g. case, pallet, each). Valid codes are defined in the DIMO code type. Examples are: CA - Case, EA - Each.
tareWeight	No	Number (12,4)	Amount of weight to be subtracted for packaging materials. Used to calculate the true net weight of the dimensionObject.
tareType	No	String (6)	Indicates if tare weight for this dimensionObject is wet or dry. Valid values are defined in the TARE code type. Examples are: W - Wet tare weight, D - Dry tare weight.
lwhUom	No	String (4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
length	No	Number (12,4)	Length of dim_object measured in units specified in lwh_uom.
width	No	Number (12,4)	Width of dim_object measured in units specified in lwh_uom.
height	No	Number (12,4)	Height of dim_object measured in units specified in lwh_uom.
liquidVolume	No	Number (12,4)	Liquid volume, or capacity, of dim_object measured in units specified in volume_uom. Liquid volumes are only convertible to other liquid volumes.
liquidVolumeUom	No	String (4)	Unit of measurement for liquid_volume (e. g. ounces, liters). Liquid volumes are only convertible to other liquid volumes. Valid values for this field are contained in uom field on uom_class table where uom_class field = LVOL.
statisticalCase	No	String (12)	Statistical value of the dim_objects dimensions to be used for loading purposes.

Table 5-632 (Cont.) Dimension - Object. See list of elements for detail

Element Name	Required	Data Type	Description
weightUom	No	String (4)	Unit of measurement for weight (e. g. pounds, kilograms). Valid values for this field are contained in uom field on uom_class table where uom_class field = MASS.
weight	No	Number (12,4)	Weight of dim_object measured in units specified in weight_uom.
netWeight	No	Number (12,4)	Unit of measurement for length, width, and height (e. g. inches, centimeters, feet). Valid values for this field are contained in uom field on uom_class table where uom_class field = DIMEN.
presentationMethod	No	String (6)	Describes the packaging (if any) being taken into consideration in the specified dimensions. Valid codes are defined in the PCKT code type. Examples are: JHOOK - Peggable Packaging, STACK - Stackable Packaging.

Table 5-633 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-634 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
```

```

"countryOfSourcing": [
  {
    "originCountry": "US",
    "dimension": [
      {
        "dimensionObject": "CA",
        "tareWeight": 3,
        "tareType": "D",
        "lwhUom": "IN",
        "length": 3,
        "width": 3,
        "height": 3,
        "liquidVolume": 3,
        "liquidVolumeUom": "EA",
        "statisticalCase": null,
        "weightUom": "EA",
        "weight": 3,
        "netWeight": 11.00,
        "presentationMethod": null
      }
    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
],
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Supplier Country Locations

Functional Area

Items - Item Definition

Business Overview

This service is used to modify locations of an existing item supplier country in Merchandising. To update item supplier country location information, the supplier country location must exist for the item in Merchandising; otherwise, an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/supplier/country/locations/update

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-635 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-635 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-636 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
itemSuppCountryLocHierarchyLevel	No	String (2)	The level of the organizational hierarchy used to update item supplier country location relationships. Valid values are: 'S' - Store and 'W' - Warehouse.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-637 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
countryOfSourcing	No	Collection of Object	References a collection of country of sourcing of the item.

Table 5-638 CountryOfSourcing - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountry	Yes	String (3)	The identifier for the country where the item is sourced. This value must be predefined on the COUNTRY table.
location	No	Collection of Object	References a collection of item supplier country locations.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-639 Location - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyId	Yes	Number (10)	The specific location id for this Item Supplier Country Loc message. Corresponds with the value used in the iscloc_hier_level field.
unitCost	No	Number (20,4)	The unit cost for the item at the item supplier country loc level in the supplier's currency. This field will be used for creating new records, not updating existing records.
negotiatedItemCost	No	Number (20,4)	This will hold the supplier negotiated item cost.
pickupLeadTime	No	Number (4)	Contains time it takes to get the item from the Supplier to the Initial Receiving Location. This value will be defaulted to the item_supp_country_loc pickup lead time field. The ordering dialog will reference the item/supplier/country/location pickup lead time as the value may vary by location.
roundLevel	No	String (6)	This column will be used to determine how order quantities will be rounded to Case, Layer and Pallet.
roundToCasePercentage	No	Number (12,4)	The Case Rounding Threshold value. Used to determine whether to round partial case quantities up or down.
roundToLayerPercentage	No	Number (12,4)	The Layer Rounding Threshold value. Used to determine whether to round partial Layer quantities up or down.
roundToPalletPercentage	No	Number (12,4)	The Pallet Rounding Threshold value. Used to determine whether to round partial Pallet quantities up or down.
roundToInnerPercentage	No	Number (12,4)	The Inner Rounding Threshold value. Used to determine whether to round partial Inner quantities up or down.
supplierHierarchyLevel1	No	String (10)	Highest level of supplier hierarchy (e. g. manufacturer). This will be used for assigning rebates that come from a level in the supply chain that is higher than the supplier.
supplierHierarchyLevel2	No	String (10)	Second highest level of supplier hierarchy. Rebates at this level will include all eligible supplier/item/country/loc records assigned to this supplier hierarchy level.
supplierHierarchyLevel3	No	String (10)	Third highest level of supplier hierarchy, used for assigning rebates by a level other than supplier.

Table 5-639 (Cont.) Location - Object. See list of elements for detail

Element Name	Required	Data Type	Description
costUom	No	String (4)	Unit of measure associated with the item cost. It is held to allow costs to be managed in a separate UOM from the standard UOM.
purchaseType	No	String (1)	This contains a code to indicate whether the item is normal merchandise (i. e. owned by the retailer), consignment stock or a concession item. Valid values are 0 - Owned, 1 - Consignment, 2 - Concession.
calculationBasis	No	String (1)	This indicates if the cost for the consignment/concession item will be managed either based on cost per unit or as a percentage of retail. Valid values are C - Cost per Unit, P - Purchase Rate.
purchaseRate	No	Number (12,4)	This column contains the percentage of the retail price which will determine the cost paid to the supplier for a consignment or concession item.
pricingControl	No	String (1)	This indicates whether the price for the consignment/concession item will be managed by supplier or by retailer when supplier-controlled pricing for Consignment/Concession ownership model is used within the Merchandising suite of solutions. Valid values are: R – Retailer S – Supplier. If no value is provided for consignment/concession purchase types, this will be defaulted from item/supplier/country of sourcing if purchase type is Consignment/Concession at item/supplier/country of sourcing level. If purchase type is Owned at item/supplier/country of sourcing level, this will be defaulted based on Default Pricing Control at supplier site level.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-640 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-641 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-642 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100068",
      "itemSuppCountryLocHierarchyLevel": "S",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "countryOfSourcing": [
            {
              "originCountry": "US",
              "location": [
                {
                  "hierarchyId": 1111,
                  "unitCost": 9.65,
                  "negotiatedItemCost": 9.65,
                  "pickupLeadTime": 1000,
                  "roundLevel": "C",
                  "roundToCasePercentage": 50,
                  "roundToLayerPercentage": 50,
                  "roundToPalletPercentage": 50,
                  "roundToInnerPercentage": 50,

```

```

        "supplierHierarchyLevel1": null,
        "supplierHierarchyLevel2": null,
        "supplierHierarchyLevel3": null,
        "costUom": "EA",
        "purchaseType": "0",
        "calculationBasis": null,
        "purchaseRate": 3,
        "pricingControl": "R",
        "customFlexAttribute": [
            {
                "name": "REVIEW_DATE",
                "value": null,
                "valueDate": "2001-12-31"
            }
        ]
    },
    "localizationExtensions": [
        {
            "country": "BR",
            "attributes": [
                {
                    "name": "SPECIAL_CATEGORY_CODE",
                    "value": "WS_PRC_08",
                    "valueDate": "2001-12-31"
                }
            ]
        }
    ]
},
"localizationExtensions": [
    {
        "country": "BR",
        "attributes": [
            {
                "name": "SPECIAL_CATEGORY_CODE",
                "value": "WS_PRC_08",
                "valueDate": "2001-12-31"
            }
        ]
    }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Supplier Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to modify supplier level translations of an existing item within Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/supplier/translations/update

Input Payload Details

Table 5-643 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-644 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-645 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
translation	No	Collection of Object	References a collection of translations of item supplier description fields.

Table 5-646 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This is the language ID.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierLabel	No	String (15)	The supplier label for the item.

Table 5-647 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-648 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "106150023",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "translation": [
            {
              "language": 2,
              "supplierDiff1": "Lieferantenunterschied 1",
              "supplierDiff2": "Lieferant Diff 2 Modifiziert",
              "supplierDiff3": "Lieferantenunterschied 3",
              "supplierDiff4": "Lieferantenunterschied 4",
              "supplierLabel": "Lieferantenetikett Geändert"
            }
          ]
        }
      ]
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

```
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Suppliers

Functional Area

Items - Item Definition

Business Overview

This service is used to update supplier level information linked with an item within Merchandising. The specified supplier must already exist for the item in Merchandising, else an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/suppliers/update

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-649 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-650 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-651 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
primarySupplierInd	No	String (3)	Indicates if the supplier is the primary supplier for the item. Valid values are Y (yes) and N (no).
vpn	No	String (30)	The vendor product number associated with the item.
supplierLabel	No	String (15)	The supplier label for the item.
consignmentRate	No	Number (12,4)	This field contains the consignment rate for this item for the supplier.
supplierDiscontinueDate	No	date	Date which the supplier discontinues an item. The retailer should be aware that the supplier is able to reuse a UPC after 30 months and should work to ensure that no data exists in RMS for a UPC 30 months after it has been discontinued.
directShipInd	No	String (1)	Contains a value of Yes to indicate that any item associated with this supplier is eligible for a direct shipment from the supplier to the customer.

Table 5-651 (Cont.) Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
palletName	No	String (6)	Code referencing the name used to refer to the pallet. Valid codes are defined in the PALN code type. Examples are FLA - Flat, PAL - Pallet.
caseName	No	String (6)	Code referencing the name used to refer to the case. Valid codes are defined in the CASN code type. Examples are PACK - Pack, BX - Box, BG - Bag.
innerName	No	String (6)	Code referencing the name used to refer to the inner. Valid codes are defined in the INRN code type. Examples are SCS - Sub-Case, SPACK - Sub-Pack.
primaryCaseSize	No	String (6)	Used only if AIP is interfaced. Indicates the primary case size for the item supplier when an orderable item is configured for informal case types.
supplierDiff1	No	String (120)	This field contains the first supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff2	No	String (120)	This field contains the second supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff3	No	String (120)	This field contains the third supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
supplierDiff4	No	String (120)	This field contains the fourth supplier differentiator and/or description. This field may only contain a value for items with an item_parent.
concessionRate	No	Number (12,4)	The concession rate is the margin that a particular supplier receives for the sale of a concession item.
defaultExpenseProfilesInd	No	String (1)	This field indicates whether expense profiles will be defaulted to new item-supplier relationship. Valid values are Y (yes) and N (no).
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-652 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-653 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-654 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "primarySupplierInd": "Y",
          "vpn": null,
          "supplierLabel": null,
          "consignmentRate": 3,
          "supplierDiscontinueDate": "2001-12-31",
          "directShipInd": "Y",
          "palletName": "PAL",
          "caseName": "CS",
          "innerName": "INR",
          "primaryCaseSize": null,
          "supplierDiff1": null,
          "supplierDiff2": null,
          "supplierDiff3": null,
          "supplierDiff4": null,
          "concessionRate": 1,

```

```

        "defaultExpenseProfilesInd": "Y",
        "customFlexAttribute": [
          {
            "name": "REVIEW_DATE",
            "value": null,
            "valueDate": "2001-12-31"
          }
        ]
      },
    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Update Item Supplier UOM

Functional Area

Items - Item Definition

Business Overview

Service to update miscellaneous units of measure (UOM) for an item/supplier combination. This UOM indicates the equivalent of one unit of the item in the specified UOM.

For example, if the item is a single package of cigars, and the single package sold in stores contains 10 cigar 'sticks'. The standard UOM of the item will be EA, and one Each of the item is a package of 10 cigar sticks. The retailer can create a MISC UOM of 'Sticks' and in Item Supplier UOM, the UOM would be set to 'Sticks' and a Value = 10 because there are 10 sticks in a single unit of the item. This definition will allow expense, assessment and/or up charge components to be created to calculate a tax. So, if the tax is 0.13 USD per stick, then the system will use the item supplier UOM record for Sticks to determine that 1 unit of the item is 10 sticks and will therefore calculate the cost component as 0.13 x 10 sticks = 1.30 USD for one package of cigars.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/supplier/uom/update

Input Payload Details

Table 5-655 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-656 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
supplier	No	Collection of Object	References a collection of suppliers associated to the item.

Table 5-657 Supplier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	Yes	String (10)	The identifier of the supplier who sources the item. This value must be predefined on the SUPS table.
uom	No	Collection of Object	Child node holding miscellaneous units of measure (UOM) for an item/supplier combination.

Table 5-658 Uom - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uom	Yes	String (4)	Contains the unit of measure that the item is being measured in.
value	Yes	Number (20,4)	Contains the equivalent value of the Item/Suppliers shipping carton in the associated unit of measure.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "supplier": [
        {
          "supplier": "2400",
          "uom": [
            {
              "uom": "STK",
              "value": 10
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Item Taxes Upload Service

The following services are included in this functional area:

- [Create Item Taxes](#)
- [Delete Item Taxes](#)

Create Item Taxes

Functional Area

Items - Item Definition

Business Overview

This service is used to add Value Added Tax (VAT) information to an existing item in Merchandising when the tax solution for the system is Simple VAT (SVAT). The item must already exist in Merchandising, else an error will be returned. Item Tax information can also be provided during the creation of a new item. For more details on item creation, see the Item Upload Service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/taxes/create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-659 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-660 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
itemLevel	No	Number (1)	The level of the item. Valid values are 1 (single level or pack item), 2 (child item), and 3 (grandchild item).
tranLevel	No	Number (1)	The transaction level of the item. Valid values are 1, 2, and 3. If this is a transaction item this will equal the item level. If the item is a pack this will equal 1. This value cannot be modified.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
vat	No	Collection of Object	References a collection of vat information.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-661 Vat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatType	Yes	String (1)	Indicates if the VAT rate is used for purchasing or selling. Valid values are C (cost), R (retail) and B (both).
vatRegion	Yes	Number (6)	The identifier for the VAT region. This value must be predefined on the VAT_REGION table. This node is optional, if it is included this field is required.
vatCode	Yes	String (6)	The identifier for the VAT code. This node is optional, if it is included in the message this field is required.
activeDate	Yes	date	The date that the VAT rate became active for the item/VAT region combination.

Table 5-661 (Cont.) Vat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reverseVatInd	No	String (1)	Indicates if the item is subject to reverse charge VAT at the vat region. Valid values are Y (yes) and N (no).

Table 5-662 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-663 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "itemLevel": 1,
      "tranLevel": 1,
      "dataLoadingDestination": "RMS",
      "vat": [
        {
          "vatType": "B",
          "vatRegion": 1000,
          "vatCode": "S",
          "activeDate": null,
          "reverseVatInd": "N"
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```



```
    ]
  }
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Taxes

Functional Area

Items - Item Definition

Business Overview

This service is used to delete VAT (value added tax) information of an existing item in Merchandising when default tax type for the system is SVAT (Simple VAT). In order to delete an item tax, the tax information must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/taxes/delete

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-664 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-665 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
vat	No	Collection of Object	References a collection of item VAT.

Table 5-666 Vat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatType	Yes	String (1)	The vat type of the vat item info to be deleted. Valid values are C (cost), R (retail) and B (both).
vatRegion	Yes	Number (6)	The vat region of the vat item info to be deleted.
activeDate	Yes	date	The active date of the vat item info to be deleted.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "vat": [
        {
          "vatType": "R",
          "vatRegion": 1000,
          "activeDate": null
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Item Up Charge Download Service

The following services are included in this functional area:

- [Get Item Up Charges for Future Date](#)

Get Item Up Charges for Future Date

Functional Area

Items - Item Definition

Business Overview

This service can be used to retrieve the complete list of up charges and the calculated up charge value that is expected to be a part of the future cost for the item, location and future date combination provided in the service call.

The From Location input is optional, and if one is not provided, the system will use the source/default location of the provided To Location as the From Location in order to determine the list of up charges to be calculated and passed out of the service.

Service Type

POST

ReST URL

MerchIntegrations/services/item/upCharges/futureDate/get

Input Payload Details

Table 5-667 Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection. The maximum number of the items allowed in a collection is 9999.
items	Yes	Collection of Object	References a collection of From Location, To Location, Future Date and Items.

Table 5-668 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	No	Number (10)	The source location to fetch the Item Up Charges. This does not need to be provided if the To Location is a franchise location.
toLocation	Yes	Number (10)	The destination location to fetch the Item Up Charges.
futureDate	Yes	date	The future date to fetch the details of the Future Cost for the given Item/ToLocation combination. The expected date format is YYYY-MM-DD.
itemList	Yes	Array of String	The item/items for which the Up Charge details are requested. The maximum number of the items allowed is 9999.

Table 5-669 GetError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
errors	Yes	Collection of Object	

Table 5-670 Errors - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	Yes	Number (10)	The source location to fetch the Item Up Charges.

Table 5-670 (Cont.) Errors - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	Yes	Number (10)	The destination location to fetch the Item Up Charges.
futureDate	Yes	date	The future date to fetch the details of the Future Cost for the given Item/ToLocation combination. The expected date format is YYYY-MM-DD.
errors	Yes	Array of String	List of errors.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "fromLocation": 10003,
      "toLocation": 1521,
      "futureDate": "2001-12-31",
      "itemList": [
        "108200021"
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-671 GetResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of From Location, To Location, Item and Upcharge details returned as response.

Table 5-672 GetResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	Yes	Number (10)	The source location to fetch the Item Up Charges.
toLocation	Yes	Number (10)	The destination location to fetch the Item Up Charges.

Table 5-672 (Cont.) GetResponse.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
futureDate	Yes	date	The future date to fetch the details of the Future Cost for the given Item/ToLocation combination. The expected date format is YYYY-MM-DD.
item	Yes	String (25)	The item for which the Up Charge details are requested.
totalUpChargeAmount	Yes	Number (20,4)	The Total Up Charge Amount for the given Item/FromLocation/ToLocation/FutureDate combination.
currencyCode	Yes	String (3)	The currency of the ToLocation in which the Total Up Charge Amount is calculated.
upCharges	Yes	Collection of Object	Details of the individual Item Up Charge Components.

Table 5-673 UpCharges - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	Yes	String (10)	Up Charge component.
componentDescription	Yes	String (250)	Description of the Up Charge component.
upChargeGroup	Yes	String (6)	Group to which the Up Charge component belongs. Valid values are defined in code type UCHG (Up Charge Group) with values: A-Admin. Fee, F-Freight, K-Special K Fees, M-Miscellaneous, T-Taxes, W-Franchise.
calculationBasis	Yes	String (6)	Indicates whether the component is calculated on a per-unit (Specific) or ad valorem (Value) basis. Valid values are Value (V) or Specific (S).

Table 5-673 (Cont.) UpCharges - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentRate	Yes	Number (20,4)	The rate to be charged against the cost of the Item/To Location combinations. The value in this field will be a percentage if the calculation basis is V (Value), and a monetary amount if the calculation basis is S (Specific).
costBasis	Yes	String (1)	The basis, the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W) and Supplier Cost (S) and Off-Invoice Net Cost (N).
computationValueBase	Yes	String (10)	The Computation Value Basis for the Up Charge component.
perCount	Yes	Number (12,4)	The amount of the Per Count Unit of Measure to which the rate applies. This will only contain a value if the calculation basis is S (Specific).
perCountUom	Yes	String (4)	The unit of measure in which the Per Count is specified. This column will only contain a value if the calculation basis is S (Specific).
componentCurrency	Yes	String (3)	Currency of the Up Charge component.
upChargeAmount	Yes	Number (20,4)	The calculated Up Charge Amount for the specific Up Charge Component. This will be in the currency of the ToLocation.

Table 5-673 (Cont.) UpCharges - Object. See list of elements for detail

Element Name	Required	Data Type	Description
upChargeLevel	No	String (1)	This field contains the level at which the up charge is defined. Valid values are: 'D' (up charge is derived from the department level), 'I' (up charge is derived from the item level), and 'F' (up charge is tied to a Franchise store).

Sample Response Message

```

{
  "items": [
    {
      "fromLocation": 10003,
      "toLocation": 1521,
      "futureDate": "2001-12-31",
      "item": "108200021",
      "totalUpChargeAmount": 1.5,
      "currencyCode": "USD",
      "upCharges": [
        {
          "componentId": "TSFINSUR",
          "componentDescription": "Transfer Insurance",
          "upChargeGroup": "M",
          "calculationBasis": "V",
          "componentRate": 1.5,
          "costBasis": "W",
          "computationValueBase": null,
          "perCount": null,
          "perCountUom": null,
          "componentCurrency": "USD",
          "upChargeAmount": 1.5,
          "upChargeLevel": "I"
        }
      ]
    }
  ]
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-674 GetError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
errors	Yes	Collection of Object	

Table 5-675 Errors - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	Yes	Number (10)	The source location to fetch the Item Up Charges.
toLocation	Yes	Number (10)	The destination location to fetch the Item Up Charges.
futureDate	Yes	date	The future date to fetch the details of the Future Cost for the given Item/ToLocation combination. The expected date format is YYYY-MM-DD.
errors	Yes	Array of String	List of errors.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "errors": [
        {
          "fromLocation": 10003,
          "toLocation": 1521,
          "futureDate": "2001-12-31",
          "errors": [
            "Invalid Item."
          ]
        }
      ]
    }
  ]
}
```

Item Upload Service

The following services are included in this functional area:

- [Create Item Description Translations](#)
- [Create Item Image Translations](#)
- [Create Item Images](#)
- [Create Item Seasons](#)
- [Create Item Tickets](#)
- [Create Item UDA Free Form Translations](#)
- [Create Item UDAs](#)
- [Create Item Up Charges](#)
- [Create Items](#)
- [Create Required Documents](#)
- [Delete Item Description Translations](#)
- [Delete Item Image Translations](#)
- [Delete Item Images](#)
- [Delete Item Seasons](#)
- [Delete Item Tickets](#)
- [Delete Item UDA Free Form Translations](#)
- [Delete Item UDAs](#)
- [Delete Item Up Charges](#)
- [Delete Items](#)
- [Delete Required Documents](#)
- [Update Item Attributes Customized](#)
- [Update Item Description Translations](#)
- [Update Item Image Translations](#)
- [Update Item Images](#)
- [Update Item Tickets](#)
- [Update Item UDA Free Form Translations](#)
- [Update Item UDAs](#)
- [Update Item Up Charge Details](#)
- [Update Items](#)

Create Item Description Translations

Functional Area

Items - Item Definition

Business Overview

If translations have been defined for items, at item header level, then they can be integrated as part of this service. This service will accept the language and the description specified in the language to add item description translations to an existing item in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/item/description/translations/create

Input Payload Details

Table 5-676 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-677 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
translation	No	Collection of Object	References a collection of translations of the item description field.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-678 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field will hold the language in which the translated text is maintained.
shortDescription	No	String (120)	This field will hold the translated text of the shortened description of the item.
itemDescription	Yes	String (250)	This field will hold the translated text of the primary description of the item.
itemDescriptionSecondary	No	String (250)	This field will hold the translated text of the secondary description of the item.

Table 5-678 (Cont.) Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
longDesc	No	String (2000)	Long description of the item.

Table 5-679 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-680 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "106150023",
      "dataLoadingDestination": "RMS",
      "translation": [
        {
          "language": 5,
          "shortDescription": "登録項目",
          "itemDescription": "レギュラーアイテム 106150023",
          "itemDescriptionSecondary": "レギュラーアイテム二次 106150023",
          "longDesc": "100100076 Item Regular"
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

```
    ]
  }
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item Image Translations

Functional Area

Items - Item Definition

Business Overview

If translations have been defined for items, at the image level, then they can be integrated as part of this service. The service will accept the language and the description specified in the language to add image level translations to an existing item in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/item/image/translations/create

Input Payload Details

Table 5-681 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-681 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-682 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
image	No	Collection of Object	References a collection of item images.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-683 Image - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	Yes	String (120)	The name of the image.
translation	No	Collection of Object	References a collection of translations of item image description field.

Table 5-684 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This is the language ID.
imageDescription	Yes	String (40)	The description associated with the image.

Table 5-685 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-686 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100083",
      "dataLoadingDestination": "RMS",
      "image": [
        {
          "imageName": "imageName1",
          "translation": [
            {
              "language": 8,
              "imageDescription": "供☒商形象☒先"
            }
          ]
        }
      ]
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item Images

Functional Area

Items - Item Definition

Business Overview

This service is used to add images to an existing item in Merchandising. If translations have been defined for the item image, then these can be integrated as part of this service as well. The supporting node of this integration will accept the language and the related description specified in the language.

Service Type

POST

ReST URL

MerchIntegrations/services/item/images/create

Input Payload Details

Table 5-687 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-688 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.

Table 5-688 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
image	No	Collection of Object	References a collection of item images.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-689 Image - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	Yes	String (120)	The name of the image.
imageAddress	No	String (255)	Path where the file of the image is stored.
imageDescription	No	String (40)	The description associated with the image.
imageType	No	String (6)	Field indicating the type of the image of the item. Can be H(igh), M(edium), L(ow) and T(humbnail).
primaryImageInd	No	String (1)	Field indicating whether the record is a primary image or not.
displayPriority	No	Number (4)	Field to specify the display sequence order of images associated to the item per priority.
translation	No	Collection of Object	References a collection of translations of item image description field.

Table 5-690 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This is the language ID.
imageDescription	Yes	String (40)	The description associated with the image.

Table 5-691 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-692 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "image": [
        {
          "imageName": "imageName1",
          "imageAddress": "http://www.oracle.com/",
          "imageDescription": "imageDesc1",
          "imageType": "T",
          "primaryImageInd": "Y",
          "displayPriority": 1,
          "translation": [
            {
              "language": 2,
              "imageDescription": "Item Image"
            }
          ]
        }
      ]
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Item Seasons

Functional Area

Items - Item Definition

Business Overview

This service is used to add seasons to an existing item in Merchandising. Item Seasons can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/seasons/create

Input Payload Details

Table 5-693 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-694 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
season	No	Collection of Object	References a collection of item seasons.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-695 Season - Object. See list of elements for detail

Element Name	Required	Data Type	Description
seasonId	Yes	Number (3)	The season identifier.
phaseId	Yes	Number (3)	The phase identifier.
sequenceNo	No	Number (4)	This field contains a sequence number that combined with the item number will make each record unique.
diffId	No	String (10)	This field will hold a value for group/differentiator combination records only.

Table 5-696 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-697 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "season": [
        {
          "seasonId": 1,
          "phaseId": 1,
          "sequenceNo": 1,
          "diffId": null
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
    }
  ]
}
```

```

        "inputValue": "X"
      }
    ]
  }

```

Create Item Tickets

Functional Area

Items - Item Definition

Business Overview

This service is used to add tickets to an existing item in Merchandising. Tickets may be created together with the creation of a new item as well. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/tickets/create

Input Payload Details

Table 5-698 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-699 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
ticket	No	Collection of Object	References a collection of item ticket.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-700 Ticket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketTypeId	Yes	String (4)	This unique identifier for the ticket type which is associated with the item.
poPrintType	No	String (1)	This field indicates when the ticket type for the given item should be printed by default, upon the approval (A) or receipt (R) of the purchase order.
printOnPriceChangeInd	No	String (1)	This field indicates whether or not this type of ticket should be printed for this item when a permanent price change goes into effect. Valid values are Y (yes) and N (no).
ticketOverPrintPercentage	No	Number (12,4)	This field contains a percent which indicates the amount of tickets which should be printed for a given event.

Table 5-701 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-702 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "102350095",
      "dataLoadingDestination": "RMS",
      "ticket": [
        {
          "ticketTypeId": "TKT2",
          "poPrintType": "R",
          "printOnPriceChangeInd": "Y",
          "ticketOverPrintPercentage": 99
        }
      ]
    }
  ]
}
```

```

    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Item UDA Free Form Translations

Functional Area

Items - Item Definition

Business Overview

If translations have been defined for items at the freeform UDA level, they can be included as part of this service. The service will accept both the language and the description specified in the language to add freeform UDA translations to an existing item in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/item/uda/translations/create

Input Payload Details

Table 5-703 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-704 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
uda	No	Collection of Object	References a collection of item UDAs.
localizationExtensions	No	Collection of Object	Holds localization attributes, including Brazil fiscal attributes.

Table 5-705 Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number identifying the User-Defined Attribute.
displayType	Yes	String (2)	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT - Date, FF - Free From, LV - List of values.

Table 5-705 (Cont.) Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaText	No	String (250)	This field contains the text value of the Used Defined attribute for the item.
translation	No	Collection of Object	Child node holding translated UDA text values by language for the item.

Table 5-706 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field represents the language in which the translated text is maintained. The language should be from the installed language pack.
udaTextDesc	Yes	String (250)	This contains the translated value for the UDA text in the selected language.

Table 5-707 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-708 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "uda": [
        {
          "udaId": 10000,
          "displayType": "FF",
          "udaText": null,
          "translation": [
            {
              "language": 2,
              "udaTextDesc": null
            }
          ]
        }
      ],
      "localizationExtensions": [
        {
          "countryId": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
```

```

"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
]
}

```

Create Item UDAs

Functional Area

Items - Item Definition

Business Overview

This service is used to add user defined attributes (UDAs) of type date, freeform text, or list of values to an existing item in Merchandising. UDAs can also be added to a item as the item is being created. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/uda/create

Input Payload Details

Table 5-709 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-710 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
uda	No	Collection of Object	References a collection of item UDAs.

Table 5-710 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-711 Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number identifying the User-Defined Attribute.
displayType	Yes	String (2)	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT - Date, FF - Free From, LV - List of values.
udaDate	No	date	This field contains the text value of the Used Defined attribute for the item.
udaValue	No	String (30)	This field contains value of the Used Defined attribute for the item.
udaText	No	String (250)	This field contains the text value of the Used Defined attribute for the item.
translation	No	Collection of Object	Child node holding translated UDA text values by language for the item.

Table 5-712 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field represents the language in which the translated text is maintained. The language should be from the installed language pack.

Table 5-712 (Cont.) Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaTextDesc	Yes	String (250)	This contains the translated value for the UDA text in the selected language.

Table 5-713 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-714 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100087",
      "dataLoadingDestination": "RMS",
      "uda": [
        {
          "udaId": 201,
          "displayType": "FF",
          "udaDate": "2001-12-31",
          "udaValue": null,
          "udaText": "UDA FF Text 100100087",
          "translation": [
            {
              "language": 2,
              "udaTextDesc": null
            }
          ]
        }
      ]
    }
  ]
}
```

```

    ],
    "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
      ]
    }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
  {
    "error": "must be one of Y, N",
    "field": "createRecord.arg0.approveInd",
    "inputValue": "X"
  }
  ]
}

```

Create Item Up Charges

Functional Area

Items - Item Definition

Business Overview

This service is used to add upcharges to an existing item in Merchandising. Upcharges can also be created together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/upCharges/create

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Upcharge - Object. See list of elements for detail

Details - Object. See list of elements for detail

LocalizationExtensions - Object. See list of elements for detail

Attributes - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Create Items

Functional Area

Items - Item Definition

Business Overview

This service is used by an external system to create items in Merchandising.

Item request payload include the detail nodes the suppliers and supplier/countries. If the item is not a non-sellable pack, the item/zone/price node is also required. Optional nodes can be included in the message for supplier/countries, supplier/country/locations, pack components, item/VAT, dimensions, UDAs, seasons, images, translations, HTS, assessments, expenses, and up charges.

This service supports the insertion of data into the Item Induction staging tables without requiring that the data sent to be functionally complete. If a complete set of item data is sent by the external system through this service, then it is possible to automatically trigger the upload of the data directly into Merchandising by setting a flag in the input payload, such that item information is uploaded without further user interaction.

Items must be created and maintained following a logical hierarchy as outlined by the referential integrity of the item database tables: Item parents before child items; item components before items that are packs; items before item-suppliers; item/suppliers before item/supplier/countries; items before item/locations (a separate service), and so on. Failing to do so results in message failure.

The auto-creation of item children using differentiator records attached to an item parent, as currently occurs using Merchandising online processes, is not supported in this service.

When a new item is created, this service will first validate that all required fields are present in the message. After that, business level validation on the input information will be performed. If all the validations are met, then the item is created with the status defined in the message. If

the status in the message is approved, then the item is also subjected to a series of approval checks. For an item to be successfully approved, mandatory information, such as supplier, supplier country, component item information (if item is a pack), is required to be passed as part of the item message. If the item cannot be approved, it is not created. The following information can be included for the item: Expenses, HTS and Assessments, Tickets, UDAs, VAT (for SVAT tax type), Up Charges, Images, Seasons, Item Supplier Country Locations, Item Supplier Countries of Manufacture, and Item Supplier Country Dimensions. If included, these are created simultaneously with the creation of the item.

If item VAT is not populated, item VAT defaults to the corresponding department's VAT information.

Service Type

POST

ReST URL

MerchIntegrations/services/items/create

Input Payload Details

Create - Object. See list of elements for detail

Items - Object. See list of elements for detail

Supplier - Object. See list of elements for detail

CountryOfSourcing - Object. See list of elements for detail

Location - Object. See list of elements for detail

CustomFlexAttribute - Object. See list of elements for detail

Dimension - Object. See list of elements for detail

LocalizationExtensions - Object. See list of elements for detail

Attributes - Object. See list of elements for detail

CountryOfManufacture - Object. See list of elements for detail

Translation - Object. See list of elements for detail

Uom - Object. See list of elements for detail

PackDetail - Object. See list of elements for detail

Vat - Object. See list of elements for detail

RetailByZone - Object. See list of elements for detail

Uda - Object. See list of elements for detail

Uda.Translation - Object. See list of elements for detail

Season - Object. See list of elements for detail

Image - Object. See list of elements for detail

Image.Translation - Object. See list of elements for detail

Items.Translation - Object. See list of elements for detail

Hts - Object. See list of elements for detail

- Assessments - Object. See list of elements for detail
- Expense - Object. See list of elements for detail
- Ticket - Object. See list of elements for detail
- Upcharge - Object. See list of elements for detail
- Details - Object. See list of elements for detail
- ConditionalTariffTreatment - Object. See list of elements for detail
- RelatedItems - Object. See list of elements for detail
- RelatedItems.Translation - Object. See list of elements for detail
- RelatedItems.Details - Object. See list of elements for detail
- RequiredDocuments - Object. See list of elements for detail
- ItemImportAttributes - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)
In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Create Item Import Attributes

Functional Area

Items - Item Definition

Business Overview

New service endpoints to create the import specific attributes associated with a given item.

Service Type

POST

ReST URL

MerchIntegrations/services/item/importAttributes/create

Input Payload Details

Table 5-715 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-716 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
itemImportAttributes	No	Collection of Object	Child node holding import specific attributes associated with a given item.
localizationExtensions	No	Collection of Object	Holds localization attributes, including Brazil fiscal attributes.

Table 5-717 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-718 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-719 ItemImportAttributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importDescription	No	String (2000)	Contains description of the item used for importing purposes.

Table 5-719 (Cont.) ItemImportAttributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
commodity	No	String (6)	Contains the code used to classify the item by their type of merchandise.
tooling	No	Number (20,4)	Represents the cost associated with a specific Tooling/Assist charge. The value will be provided in the system's primary currency.
firstOrder	No	String (1)	Used to determine if duty is to be paid on the total value of the tooling expense the first time the item is shipped.
amortizeBase	No	Number (12,4)	Contains the number of units over which the tooling/assist cost will be amortized.
openBalance	No	Number (12,4)	Contains the open balance of the number of units yet to be received over which the tooling costs will be amortized.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "itemImportAttributes": [
        {
          "importDescription": "import desc",
          "commodity": "A123",
          "tooling": 12,
          "firstOrder": "Y",
          "amortizeBase": 12.4568,
          "openBalance": 12.4568
        }
      ],
      "localizationExtensions": [
        {
          "countryId": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",

```

```
"value": "WS_PRC_08",  
"valueDate": "2001-12-31"  
}  
]  
}  
]  
}  
]  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Create Required Documents

Functional Area

Items - Item Definition

Business Overview

This service is used to add required documents to an existing item in Merchandising. Required documents may be added together with the creation of a new item as well. For more details on item creation, see the [Create Items](#) service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/requiredDocuments/create

Input Payload Details

Table 5-720 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-721 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
requiredDocuments	No	Collection of Object	Child node holding documents information of the item.
localizationExtensions	No	Collection of Object	Holds localization attributes, including Brazil fiscal attributes.

Table 5-722 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	Yes	String (3)	ID of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-723 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-724 RequiredDocuments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
documentId	Yes	Number (6)	Unique number used to distinguish the different documents.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
```

```

    "item": "100100076",
    "dataLoadingDestination": "RMS",
    "requiredDocuments": [
      {
        "documentId": 1
      }
    ],
    "localizationExtensions": [
      {
        "countryId": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Delete Item Description Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to delete header level translations of an existing item in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/description/translations/delete

Input Payload Details

Table 5-725 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-726 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
translation	No	Collection of Object	References a collection of translations of item description field.

Table 5-727 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	The language ID.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "translation": [
        {
          "language": 5
        }
      ]
    }
  ]
}
```


Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Image Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to delete image level translations from an existing item within Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/image/translations/delete

Input Payload Details

Table 5-728 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-729 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
image	No	Collection of Object	References a collection of item image.

Table 5-730 Image - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	Yes	String (120)	The name of the image.
translation	No	Collection of Object	References a collection of translations of item image description field.

Table 5-731 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	The language ID.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100000041",
      "image": [
        {
          "imageName": "Omni_Int_Delta_Change",
          "translation": [
            {
              "language": 38
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Images

Functional Area

Items - Item Definition

Business Overview

This service is used to delete an existing item image information from Merchandising. In order to perform this deletion, the image must exist for the item in Merchandising, else an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/images/delete

Input Payload Details

Table 5-732 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-733 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
image	No	Collection of Object	References a collection of item image.

Table 5-734 Image - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	Yes	String (120)	The name of the image.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "image": [
        {
          "imageName": "imageName"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Seasons**Functional Area**

Items - Item Definition

Business Overview

This service is used to delete season data from an existing item in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/seasons/delete

Input Payload Details

Table 5-735 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-736 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
season	No	Collection of Object	References a collection of item season.

Table 5-737 Season - Object. See list of elements for detail

Element Name	Required	Data Type	Description
seasonId	Yes	Number (3)	The season identifier.
phaseId	Yes	Number (3)	The phase identifier
diffId	No	String (10)	This field will hold a value for group/differentiator combination records only.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "season": [
        {
          "seasonId": 1,
          "phaseId": 1,
          "diffId": null
        }
      ]
    }
  ]
}
```

```
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Tickets

Functional Area

Items - Item Definition

Business Overview

This service is used to delete existing item tickets from Merchandising. In order to perform this deletion, the item/ticket type ID must exist within Merchandising, else an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/tickets/delete

Input Payload Details

Table 5-738 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-738 (Cont.) Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-739 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
ticket	No	Collection of Object	References a collection of item ticket.

Table 5-740 Ticket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketTypeId	Yes	String (4)	Id of the ticket type associated with the item.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "ticket": [
        {
          "ticketTypeId": "TKT1"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item UDA Free Form Translations**Functional Area**

Items - Item Definition

Business Overview

This service is used to delete freeform UDA translation from an existing item in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/uda/translations/delete

Input Payload Details**Table 5-741 Delete - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-742 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
uda	No	Collection of Object	References a collection of item uda.

Table 5-743 Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number uniquely identifying the User-Defined Attribute.
displayType	Yes	String (2)	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT - Date, FF - Free From, LV - List of values.
udaValue	No	String (250)	This field contains value of the Used Defined attribute for the item
translation	No	Collection of Object	Child node.

Table 5-744 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field represents the language for which the translation needs to be deleted.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "uda": [
        {
          "udaId": 201,
          "displayType": "FF",
          "udaValue": null,
          "translation": [
            {
              "language": 2
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item UDAs

Functional Area

Items - Item Definition

Business Overview

This service is used to delete item UDAs in Merchandising. In order to delete an item UDA, the UDA must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/uda/delete

Input Payload Details

Table 5-745 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-746 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
uda	No	Collection of Object	References a collection of item uda.

Table 5-747 Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number uniquely identifying the User-Defined Attribute.
displayType	Yes	String (2)	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT - Date, FF - Free From, LV - List of values.
udaDateValueText	No	String (250)	This field contains value of the Used Defined attribute for the item

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "uda": [
        {
          "udaId": 201,
          "displayType": "FF",
          "udaDateValueText": "FF14St"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Up Charges

Functional Area

Items - Item Definition

Business Overview

This service is used to delete existing item upcharges from Merchandising. In order to delete item upcharges, the upcharge or upcharge/component id must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/upCharges/delete

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-748 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-748 (Cont.) Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-749 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
upcharge	No	Collection of Object	References a collection of item upcharge.

Table 5-750 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocationType	Yes	String (6)	Location type of the from_loc. This can be S - Store,W - Warehouse,A - Area,R - Region,C - Country,AS - All Stores,AW - All Warehouses,PW - Physical Warehouse
fromLocation	No	String (10)	Depending on the from_loc_type, this contains the organization hierarchy value of store, warehouse, area, region, or country from which transferring goods will incur upcharges. It will not contain a value when from_loc_type is AS (all stores) or AW (all warehouses).
toLocationType	Yes	String (6)	Location_type of the to_loc. This can be S - Store,W - Warehouse,A - Area,R - Region,C - Country,AS - All Stores,AW - All Warehouses,PW - Physical Warehouse
toLocation	No	String (10)	Depending on the to_loc_type, this contains the organization hierarchy value of store, warehouse, area, region, or country from which transferring goods will incur upcharges. It will not contain a value when to_loc_type is AS (all stores) or AW (all warehouses).
details	No	Collection of Object	References a collection of item upcharge detail records.

Table 5-751 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	Yes	String (10)	This field contains the unique identifier of the Up Charge component.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "104600060",
      "upcharge": [
        {
          "fromLocationType": "S",
          "fromLocation": "1211",
          "toLocationType": "S",
          "toLocation": "1111",
          "details": [
            {
              "componentId": "WHFEE"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Items

Functional Area

Items - Item Definition

Business Overview

This service is used to delete existing items within Merchandising based on the item number present in the message.

If you are deleting an item, the service will first validate that the item number provided is valid. If the item does not exist in Merchandising, an error message will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/items/delete

Input Payload Details

Table 5-752 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-753 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Required Documents**Functional Area**

Items - Item Definition

Business Overview

This service is used to delete required document data from an existing item in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/requiredDocuments/delete

Input Payload Details

Table 5-754 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-755 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
requiredDocuments	No	Collection of Object	Child node.

Table 5-756 RequiredDocuments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
documentId	Yes	Number (6)	Unique number used to distinguish the different documents.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "requiredDocuments": [
        {
          "documentId": 1
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Description Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to modify header level translations of an existing item in Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/description/translations/update

Input Payload Details

Table 5-757 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-758 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
translation	No	Collection of Object	References a collection of translations of the item description field.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-759 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field will hold the language in which the translated text is maintained.
shortDescription	No	String (120)	This field will hold the translated text of the shortened description of the item.
itemDescription	Yes	String (250)	This field will hold the translated text of the primary description of the item.
itemDescriptionSecondary	No	String (250)	This field will hold the translated text of the secondary description of the item.

Table 5-759 (Cont.) Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
longDesc	No	String (2000)	Long description of the item.

Table 5-760 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-761 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "106150023",
      "dataLoadingDestination": "RMS",
      "translation": [
        {
          "language": 5,
          "shortDescription": "アイテム 106150023",
          "itemDescription": "レギュラーアイテム 106150023",
          "itemDescriptionSecondary": "レギュラーアイテム二次 106150023",
          "longDesc": "100100076 Item Regular"
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

```
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Image Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to modify image level translations of an existing item within Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/image/translations/update

Input Payload Details

Table 5-762 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-762 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-763 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
image	No	Collection of Object	References a collection of item images.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-764 Image - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	Yes	String (120)	The name of the image.
translation	No	Collection of Object	References a collection of translations of item image description field.

Table 5-765 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This is the language ID.
imageDescription	Yes	String (40)	The description associated with the image.

Table 5-766 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-767 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "image": [
        {
          "imageName": "imageName1",
          "translation": [
            {
              "language": 20,
              "imageDescription": "Π ρ ο η γ ο ρ ε ν η ε ι κ ο ν α"
            }
          ]
        }
      ]
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Images

Functional Area

Items - Item Definition

Business Overview

This service is used to update an existing item image within Merchandising. In order to update an item image, the specified image name must exist for the item in Merchandising, else an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/images/update

Input Payload Details

Table 5-768 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-769 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.

Table 5-769 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
image	No	Collection of Object	References a collection of item images.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-770 Image - Object. See list of elements for detail

Element Name	Required	Data Type	Description
imageName	Yes	String (120)	The name of the image.
imageAddress	No	String (255)	Path where the file of the image is stored.
imageDescription	No	String (40)	The description associated with the image.
imageType	No	String (6)	Field indicating the type of the image of the item. Can be H(igh), M(edium), L(ow) and T(humbnail).
primaryImageInd	No	String (1)	Field indicating whether the record is a primary image or not.
displayPriority	No	Number (4)	Field to specify the display sequence order of images associated to the item per priority.

Table 5-771 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-772 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "image": [
        {
          "imageName": "imageName1",
          "imageAddress": "http://www.oracle.com/",
          "imageDescription": "imageDesc1",
          "imageType": "T",
          "primaryImageInd": "Y",
          "displayPriority": 1
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
```

```

        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
    }
  ]
}

```

Update Item Tickets

Functional Area

Items - Item Definition

Business Overview

This service is used to modify tickets of existing items in Merchandising. In order to update tickets, the item/ticket type id must exist in Merchandising, otherwise, an error will be returned. All item ticket fields except for ticket type ID are updateable and will go through the same validation as in the creation of item tickets.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/tickets/update

Input Payload Details

Table 5-773 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-774 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
ticket	No	Collection of Object	References a collection of item ticket.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-775 Ticket - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ticketTypeId	Yes	String (4)	This unique identifier for the ticket type which is associated with the item.
poPrintType	No	String (1)	This field indicates when the ticket type for the given item should be printed by default, upon the approval (A) or receipt (R) of the purchase order.
printOnPriceChangeInd	No	String (1)	This field indicates whether or not this type of ticket should be printed for this item when a permanent price change goes into effect. Valid values are Y (yes) and N (no).
ticketOverPrintPercentage	No	Number (12,4)	This field contains a percent which indicates the amount of tickets which should be printed for a given event.

Table 5-776 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-777 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "102350132",
      "dataLoadingDestination": "RMS",
      "ticket": [
        {
          "ticketTypeId": "TKT1",
          "poPrintType": "R",
          "printOnPriceChangeInd": "N",
          "ticketOverPrintPercentage": 3
        }
      ]
    }
  ]
}
```

```

    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Update Item UDA Free Form Translations

Functional Area

Items - Item Definition

Business Overview

This service is used to update freeform item UDA level translations of an existing item within Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/uda/translations/update

Input Payload Details

Table 5-778 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-779 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
uda	No	Collection of Object	References a collection of item UDAs.
localizationExtensions	No	Collection of Object	Holds localization attributes, including Brazil fiscal attributes.

Table 5-780 Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number identifying the User-Defined Attribute.
displayType	Yes	String (2)	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT - Date, FF - Free From, LV - List of values.

Table 5-780 (Cont.) Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaText	No	String (250)	This field contains the text value of the Used Defined attribute for the item.
translation	No	Collection of Object	Child node holding translated UDA text values by language for the item.

Table 5-781 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field represents the language in which the translated text is maintained. The language should be from the installed language pack.
udaTextDesc	Yes	String (250)	This contains the translated value for the UDA text in the selected language.

Table 5-782 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-783 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "uda": [
        {
          "udaId": 10000,
          "displayType": "FF",
          "udaText": null,
          "translation": [
            {
              "language": 2,
              "udaTextDesc": null
            }
          ]
        }
      ],
      "localizationExtensions": [
        {
          "countryId": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
```

```

"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
]
}

```

Update Item UDAs

Functional Area

Items - Item Definition

Business Overview

This service is used to modify UDAs for an existing item in Merchandising. In order to update an item UDA of type date, freeform text, or list of values, the UDA must exist for the item in Merchandising, otherwise, an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/uda/update

Input Payload Details

Table 5-784 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-785 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
uda	No	Collection of Object	References a collection of item UDAs.

Table 5-785 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-786 Uda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaId	Yes	Number (5)	This field contains a number identifying the User-Defined Attribute.
displayType	Yes	String (2)	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT - Date, FF - Free From, LV - List of values.
udaDate	No	date	This field contains the text value of the Used Defined attribute for the item.
udaValue	No	String (30)	This field contains value of the Used Defined attribute for the item.
udaText	No	String (250)	This field contains the text value of the Used Defined attribute for the item.
newUdaDate	No	date	This field contains the value for modification the uda_date of the Used Defined attribute for the item.
newUdaValue	No	String (30)	This field contains the value for modification the uda_value of the Used Defined attribute for the item.
newUdaText	No	String (250)	This field contains the value for modification the uda_date of the Used Defined attribute for the item.

Table 5-787 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-788 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "uda": [
        {
          "udaId": 10000,
          "displayType": "FF",
          "udaDate": "2001-12-31",
          "udaValue": null,
          "udaText": null,
          "newUdaDate": "2001-12-31",
          "newUdaValue": null,
          "newUdaText": null
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}

```

```
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Up Charge Details

Functional Area

Items - Item Definition

Business Overview

This service is used to modify details for existing item upcharges in Merchandising. To update an item upcharge detail, the upcharge/component ID must exist for the item in Merchandising; otherwise, an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/upCharge/details/update

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Table 5-789 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-790 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
upcharge	No	Collection of Object	References a collection of item upcharges.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-791 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocationType	Yes	String (6)	Location type of the from_loc. This can be S - Store,W - Warehouse,A - Area,R - Region,C - Country,AS - All Stores,AW - All Warehouses,PW - Physical Warehouse
fromLocation	No	String (10)	Depending on the from_loc_type, this contains the organization hierarchy value of store, warehouse, area, region, or country from which transferring goods will incur upcharges. It will not contain a value when from_loc_type is AS (all stores) or AW (all warehouses).
toLocationType	Yes	String (6)	Location_type of the to_loc. This can be S - Store,W - Warehouse,A - Area,R - Region,C - Country,AS - All Stores,AW - All Warehouses,PW - Physical Warehouse
toLocation	No	String (10)	Depending on the to_loc_type, this contains the organization hierarchy value of store, warehouse, area, region, or country to which transferring goods will incur upcharges. It will not contain a value when to_loc_type is AS (all stores) or AW (all warehouses).

Table 5-791 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
details	Yes	Collection of Object	References a collection of item upcharge detail records.

Table 5-792 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	Yes	String (10)	Up Charge component.
componentRate	Yes	Number (20,4)	The rate to be charged against the cost of the Item/To Location combinations within the department. The value in this field will be a percentage if the calculation basis is V (Value), and a monetary amount if the calculation basis is S.
perCount	No	Number (12,4)	count indicating the amount of the Per Count Unit of Measure to which the rate applies. This will only contain a value if the calculation basis is S (Specific).
perCountUom	No	String (4)	contains the unit of measure in which the Per Count is specified. This column will only contain a value if the calculation basis is S (Specific).
upChargeGroup	Yes	String (6)	group to which the component ID belongs. This can be A-Admin. Fee,F-Freight,K-Special K Fees,M-Miscellaneous,T-Taxes,W-Franchise.
componentCurrency	Yes	String (3)	Currency of the Up Charge component.
transferAllocationDefaultInd	No	String (1)	Indicates whether all existing transfers and allocations belonging to the hierarchy should be updated or not.
computationValueBase	No	String (10)	Contains the Computation Value Basis for the Up Charge component.
costBasis	No	String (6)	Contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W) and Supplier Cost (S) and Off-Invoice Net Cost (N).
includeInTotalUpchargeInd	No	String (1)	Used to determine if the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N).

Table 5-793 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-794 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100500029",
      "dataLoadingDestination": "RMS",
      "upcharge": [
        {
          "fromLocationType": "W",
          "fromLocation": "10001",
          "toLocationType": "S",
          "toLocation": "1231",
          "details": [
            {
              "componentId": "WHFEE",
              "componentRate": 3.55,
              "perCount": 1,
              "perCountUom": "EA",
              "upChargeGroup": "A",
              "componentCurrency": "USD",
              "transferAllocationDefaultInd": "Y",
              "computationValueBase": null,
              "costBasis": null,
              "includeInTotalUpchargeInd": "Y"
            }
          ]
        }
      ]
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
```

```
        "value": "WS_PRC_08",  
        "valueDate": "2001-12-31"  
      }  
    ]  
  }  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Update Items

Functional Area

Items - Item Definition

Business Overview

This service is used to modify header level information of existing items in Merchandising.

Updates can be made to the items that are in Worksheet, Submitted, or Approved status. In order to perform the update, the service will validate that the item number included in the message already exists within Merchandising. Only header level fields need to be provided for header level updates. Any item details included in the message will be ignored for a header level update message. There are certain fields that are not allowed to be updated at the header level depending on the status, and if these are still provided in the message, an appropriate error message will be returned. Any custom flexible attributes (CFAS) defined at the item level, can be updated as part of this service. The flexible attributes are treated as a single group and the complete CFAS payload needs to be included in the message for Update.

If an attribute is omitted from the message or is included with an empty (NULL) value, then the existing value will be cleared out.

This service can also be used to reinstate a deleted item back to its previous status. In such a scenario, the status needs to be passed as 'R' which will solely reinstate the item to its last defined status and ignore any updates to other fields included in the payload.

Service Type

PUT

ReST URL

MerchIntegrations/services/items/update

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-795 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-796 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
status	No	String (1)	Status of the item.
itemDescription	No	String (250)	The long description of the item.
shortDescription	No	String (120)	The short description of the item. If this field is not populated it will default from the long description.
itemDescriptionSecondary	No	String (250)	Secondary descriptions of the item.
longDesc	No	String (2000)	Long description of the item.
standardUom	No	String (4)	The unit of measure at which the item is tracked at the corporate level. This will be defaulted to EA (eaches) if not populated on the message. It cannot be modified.
primaryReferenceItemInd	No	String (1)	This field indicates whether the sub-transaction level item is designated as the primary sub-transaction level item. For transaction level items and above the value in this field will be No.

Table 5-796 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diff1	No	String (10)	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Mens pant sizes) or a value (6 oz).
diff2	No	String (10)	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Mens pant sizes) or a value (6 oz).
diff3	No	String (10)	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Mens pant sizes) or a value (6 oz).
diff4	No	String (10)	Diff_group or diff_id that differentiates the current item from its item_parent. For an item that is a parent, this field may be either a group (i.e. Mens pant sizes) or a value (6 oz).
storeOrderMultiple	No	String (1)	The unit type of merchandise shipped from the warehouse to the store for this item. Valid values are C (cases), I (innners), and E (eaches).
forecastInd	No	String (1)	Indicates whether the item is forecastable. Valid value are Y (yes) and N (no). This value should be N if the item is a pack.
uomConversionFactor	No	Number (20,10)	Conversion factor between an Each and the standard_uom when the standard_uom is not in the quantity class (e. g. if standard_uom = lb and 1 lb = 10 eaches, this factor will be 10). This factor will be used to convert sales and stock data when an item is retailed in eaches but does not have eaches as its standard unit of measure.
packageSize	No	Number (12,4)	Holds the size of the product printed on any packaging (i. e. 24 ounces). This field is used for reporting purposes, as well as by Oracle Retail Price Mangement to determine same sized and different sized items.
handlingTemperature	No	String (6)	Holds the temperature information associated with the item. Valid codes are defined in the HTMP code type. Examples are: FRIDGE - Keep Refrigerated, FROZEN - Keep Frozen.

Table 5-796 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
handlingSensitivity	No	String (6)	Holds the sensitivity information associated with the item. Valid codes are defined in the HSEN code type. Examples are: EXPLOD - Explosive, FRAG - Fragile.
manufacturerRecommended Retail	No	Number (20,4)	Manufacturers recommended retail price for the item in primary currency. Used for informational purposes only.
wasteType	No	String (6)	Identifies the wastage type as either sales or spoilage wastage. Valid values are: SP - Spoilage, SL - Sales Wastage is not applicable to pack items.
averageWastePercentage	No	Number (12,4)	Average percent of wastage for the item over its shelf life. Used in inflating the retail price for wastage items.
catchWeightInd	No	String (1)	Indicates whether the item should be weighed when it arrives at a location. Valid values for this field are Y and N.
containerItem	No	String (25)	This holds the container item number for a content item.
packageUom	No	String (4)	Holds the unit of measure associated with the package size. This field is used for reporting purposes and by Retek Price Management to determine same sized and different sized items.
brandName	No	String (120)	This is added to capture the brand information of an item.
productClassification	No	String (6)	This defines which items can be combined (packed or boxed) together. Valid codes are defined in the PCLA code type. Examples are: APP - Apparel, GRC - Grocery.
originalRetail	No	Number (20,4)	The original retail price of the item per unit. This field is stored in the primary currency.
retailLabelType	No	String (6)	This field indicates any special label type associated with an item. Valid codes are defined in the RTLTL code type. Examples are: COFF - Cents Off, EXTRA - Extra Product, PREP - Pre-Priced.
retailLabelValue	No	Number (20,4)	This field represents the value associated with the retail label type.
defaultWastePercentage	No	Number (12,4)	Default daily wastage percent for spoilage type wastage items.

Table 5-796 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
itemServiceLevel	No	String (6)	Holds a value that restricts the type of shipment methods that RCOM can select for an item.
depositInPricePerUom	No	String (6)	This field indicates if the deposit amount is included in the price per UOM calculation for a contents item ticket. Valid codes are defined in the DITM code type. Examples are: E - Exclusive of Deposit Amount, I - Inclusive of Deposit Amount.
constantDimensionInd	No	String (1)	Indicates that the dimensions of the product are always the same, regardless of the supplier. If this field is Y, the dimensions for all suppliers will be defaulted to the primary supplier/primary country dimensions. Editing of dimensions for the item will only be allowed for the primary supplier/primary country.
giftWrapInd	No	String (1)	This field will contain a value of 'Y' if the item is eligible to be gift wrapped.
shipAloneInd	No	String (1)	This field will contain a value of Y if the item should be shipped to the customer is a separate package versus being grouped together in a box.
aipCaseType	No	String (6)	Only used if AIP is integrated. Determines which case sizes to extract against an item in the AIP interface. Applicable only to non-pack orderable items. Valid values are defined in the AIPT code type. Examples are: F - Formal and I - Informal.
transferAllocUpChargeSource eType	No	String (6)	This field impacts how up charges default to transfers and allocations when an item is added. When set to 'D', department-level up charges are applied if item-level up charges are not defined for the item/from-location/to-location combination. When set to 'I', only item-level up charges are applied; department-level up charges are not inherited.
comments	No	String (2000)	Comments associated with the item.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.

Table 5-796 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-797 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-798 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-799 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "status": "W",
      "itemDescription": "Regular Item 100100076",
      "shortDescription": "100100076 Item",

```

```

"itemDescriptionSecondary": "100100076 Item Regular",
"longDesc": "100100076 Item Regular",
"standardUom": "EA",
"primaryReferenceItemInd": "N",
"diff1": null,
"diff2": null,
"diff3": null,
"diff4": null,
"storeOrderMultiple": "E",
"forecastInd": null,
"uomConversionFactor": 3,
"packageSize": 90.00,
"handlingTemperature": null,
"handlingSensitivity": null,
"manufacturerRecommendedRetail": 12.99,
"wasteType": null,
"averageWastePercentage": 18.00,
"catchWeightInd": "N",
"containerItem": null,
"packageUom": null,
"brandName": null,
"productClassification": null,
"originalRetail": 12.99,
"retailLabelType": null,
"retailLabelValue": null,
"defaultWastePercentage": null,
"itemServiceLevel": null,
"depositInPricePerUom": "I",
"constantDimensionInd": "Y",
"giftWrapInd": "Y",
"shipAloneInd": "Y",
"aipCaseType": null,
"transferAllocUpChargeSourceType": "D",
"comments": null,
"dataLoadingDestination": "RMS",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"localizationExtensions": [
  {
    "country": "BR",
    "attributes": [
      {
        "name": "SPECIAL_CATEGORY_CODE",
        "value": "WS_PRC_08",
        "valueDate": "2001-12-31"
      }
    ]
  }
]
}

```

```
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Item Import Attributes

Functional Area

Items - Item Definition

Business Overview

New service endpoints will be added to modify the import specific attributes associated with a given item.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/importAttributes/update

Input Payload Details

Table 5-800 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.

Table 5-800 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of items.

Table 5-801 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	No	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
itemImportAttributes	No	Collection of Object	Child node holding import specific attributes associated with a given item.
localizationExtensions	No	Collection of Object	Holds localization attributes, including Brazil fiscal attributes.

Table 5-802 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryId	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-803 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-804 ItemImportAttributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importDescription	No	String (2000)	Contains description of the item used for importing purposes.
commodity	No	String (6)	Contains the code used to classify the item by their type of merchandise.
tooling	No	Number (20,4)	Represents the cost associated with a specific Tooling/Assist charge. The value will be provided in the system's primary currency.
firstOrder	No	String (1)	Used to determine if duty is to be paid on the total value of the tooling expense the first time the item is shipped.
amortizeBase	No	Number (12,4)	Contains the number of units over which the tooling/assist cost will be amortized.
openBalance	No	Number (12,4)	Contains the open balance of the number of units yet to be received over which the tooling costs will be amortized.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "itemImportAttributes": [
        {
          "importDescription": "import desc",
          "commodity": "A123",
          "tooling": 12,
          "firstOrder": "Y",
          "amortizeBase": 12.4568,
          "openBalance": 12.4568
        }
      ],
      "localizationExtensions": [
        {
```



```
"countryId": "BR",
"attributes": [
{
"name": "SPECIAL_CATEGORY_CODE",
"value": "WS_PRC_08",
"valueDate": "2001-12-31"
}
]
}
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Manage Substitute Items

Functional Area

Items - Item Definition

Business Overview

Service endpoint to create, update or delete substitute items for an item.

Service Type

POST

ReST URL

MerchIntegrations/services/item/location/substituteItems/manage

Input Payload Details

Table 5-805 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	
items	Yes	Collection of Object	

Table 5-806 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Describes type of action.
item	Yes	String (25)	Alphanumeric identifier for the main item. This item must exist as a main sku on the sub_items_head table.
locationHierarchyLevel	Yes	String (5)	The corresponding ID based on the specified level of the organizational hierarchy. Valied values are codes from code_type 'GRTA'.
hierarchy	No	String (20)	The level of the organizational hierarchy used for item-location relationships to add substitute item.
fillPriority	No	String (1)	This contains the fill priority for the main item. This field applies only to Warehouses.
useSales	No	String (1)	This indicates whether substitute items will be used in calculating the maximum stock for the floating-point replenishment method.

Table 5-806 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
useStock	No	String (1)	This indicates whether the substitute items will be used in calculating the net inventory at a location when determining the locations need for the main item. When set Yes ('Y'), the substitute item stock can be used to fulfil a locations warehouse demands for the main item.
useForecastSales	No	String (1)	This indicates whether the substitute item be used to determine the required stock at a given location when a forecasting replenishment method is used for the main item.
details	No	Collection of Object	Child node containing the details of substitute items.

Table 5-807 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
substituteItem	Yes	String (25)	This contains unique identifier for the substitute item.
primaryReplenishment Pack	No	String (25)	This contains the replenishment pack, which will be used to fulfill the demand of the associated item.
pickPriority	No	Number (3)	Contains the pick priority for the substitute item. If there are multiple substitute items for a main item, then the pick priority will determine the order in which the substitute items will fulfill the store's demand.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "item": "100100050",
      "locationHierarchyLevel": "S",
      "hierarchy": "1010",
      "fillPriority": "M",
      "useSales": "Y",
      "useStock": "Y",
      "useForecastSales": "Y",
      "details": [
        {
          "substituteItem": "100100068",
          "primaryReplenishmentPack": "100100092",
          "pickPriority": 1
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Related Items Upload Service

The following services are included in this functional area:

- [Create Related Items](#)
- [Create Related Item Translations](#)

- [Delete Related Items](#)
- [Delete Related Item Translations](#)
- [Update Related Items](#)
- [Update Related Item Translations](#)

Create Related Items

Functional Area

Items - Related Items

Business Overview

This service is used to add related items to an existing item in Merchandising. Related items can also be added together with the creation of a new item. For more details on item creation, see the Create Items service description.

Service Type

POST

ReST URL

MerchIntegrations/services/item/relatedItems/create

Input Payload Details

Table 5-808 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-809 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	Yes	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
relatedItems	Yes	Collection of Object	Child node holding related-item information for the item.
localizationExtensions	No	Collection of Object	Holds localization attributes, including Brazil fiscal attributes.

Table 5-810 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-811 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-812 RelatedItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	No	Number (20)	Unique identifier for each relationship header.
relationshipName	Yes	String (255)	Name given to the relationship.
relationshipType	Yes	String (6)	Describes the type of relationship. Values are configured in code_detail table under code_type IREL.
mandatoryInd	No	String (1)	Indicates whether the relationship is mandatory.
details	Yes	Collection of Object	Child node containing the details for related item.

Table 5-813 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relatedItem	Yes	String (25)	Item id of the related item.

Table 5-813 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
priority	No	Number (4)	Applicable only in case of relationship type SUBS. In case of multiple related substitute items, this column could be used (optional) to define relative priority.
startDate	No	date	From this date related item can be used on transactions.
endDate	No	date	Till this date related item can be used on transactions. A value of null means that it is effective forever.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "relatedItems": [
        {
          "relationshipId": 10001,
          "relationshipName": "Related Item Test",
          "relationshipType": "SUBS",
          "mandatoryInd": "Y",
          "details": [
            {
              "relatedItem": "100100141",
              "priority": 1,
              "startDate": "2001-12-31",
              "endDate": "2001-12-31"
            }
          ]
        }
      ],
      "localizationExtensions": [
        {
          "country": "BR",
          "attributes": [
            {
              "name": "SPECIAL_CATEGORY_CODE",
              "value": "WS_PRC_08",
              "valueDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}

```

```
}  
]  
}  
]  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Create Related Item Translations

Functional Area

Items - Related Items

Business Overview

If translations have been defined for items, at related item level, then they can be integrated as part of this service. The service will accept the language and the description specified in the language to add related item translations to an existing item in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/item/relatedItem/translations/create

Input Payload Details

Table 5-814 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-815 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	Yes	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
relateditem	Yes	Collection of Object	Child node holding related-item information for the item.

Table 5-816 Relateditem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	Yes	Number (20)	Unique identifier for each relationship header.
translation	Yes	Collection of Object	Child node containing the translated value of the relationship name.

Table 5-817 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lang	Yes	Number (6)	This field contains the language in which the translated text is maintained.
relationshipName	Yes	String (255)	The translated text of the relationship name.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "relateditem": [
        {
          "relationshipId": 10041,
          "translation": [
            {
              "lang": 2,
              "relationshipName": "Descripción de la relación"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Related Items

Functional Area

Items - Related Items

Business Overview

This service is used to delete existing item relationship from Merchandising. In order to perform this deletion, the relationship ID must exist within Merchandising, else an error will be returned.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/relatedItems/delete

Input Payload Details

Table 5-818 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-819 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
relatedItem	Yes	Collection of Object	Child node.

Table 5-820 RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	Yes	Number (20)	Unique identifier for the relationship.
details	No	Collection of Object	Child node.

Table 5-821 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relatedItem	No	String (25)	Item id of the related item.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
```

```

    "relatedItem": [
      {
        "relationshipId": 10001,
        "details": [
          {
            "relatedItem": "100100141"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Delete Related Item Translations

Functional Area

Items - Related Items

Business Overview

This service is used to delete an existing translation record of a related item relationship in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/relatedItem/translations/delete

Input Payload Details

Table 5-822 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-823 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier for the item.
relatedItem	No	Collection of Object	Child node.

Table 5-824 RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	Yes	Number (20)	Unique identifier for the relationship.
translation	No	Collection of Object	Child node.

Table 5-825 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lang	Yes	Number (6)	The language ID.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "relatedItem": [
        {
          "relationshipId": 10001,
          "translation": [
            {
              "lang": 2
            }
          ]
        }
      ]
    }
  ]
}
```

```
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Related Items

Functional Area

Items - Related Items

Business Overview

This service is used to update item relationships of an existing item within Merchandising. In order to update an item relationship, the specified relationship ID must exist for the item in Merchandising, else an error will be returned.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/relatedItems/update

Input Payload Details

Table 5-826 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-827 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	Yes	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
relatedItems	Yes	Collection of Object	Child node holding related-item information for the item.

Table 5-828 RelatedItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	Yes	Number (20)	Unique identifier for each relationship header.
relationshipName	Yes	String (255)	Name given to the relationship.
relationshipType	Yes	String (6)	Describes the type of relationship. Values are configured in code_detail table under code_type IREL.
mandatoryInd	Yes	String (1)	Indicates whether the relationship is mandatory.
details	No	Collection of Object	Child node containing the details for related item.

Table 5-829 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relatedItem	No	String (25)	Item id of the related item.
priority	No	Number (4)	Applicable only in case of relationship type SUBS. In case of multiple related substitute items, this column could be used (optional) to define relative priority.
startDate	No	date	From this date related item can be used on transactions.
endDate	No	date	Till this date related item can be used on transactions. A value of null means that it is effective forever.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "relatedItems": [
        {
          "relationshipId": 10001,
          "relationshipName": "Related Item Test - Updated",
          "relationshipType": "SUBS",
          "mandatoryInd": "N",
          "details": [
            {
              "relatedItem": "100100141",
              "priority": 2,
              "startDate": "2001-12-31",
              "endDate": "2001-12-31"
            }
          ]
        }
      ]
    }
  ]
}
```


Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Related Item Translations

Functional Area

Items - Related Items

Business Overview

This service is used to update related item level translations of an existing item within Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/relatedItem/translations/update

Input Payload Details

Table 5-830 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of items.

Table 5-831 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The unique identifier of the main item on the message.
dataLoadingDestination	Yes	String (6)	This field indicates if the item will be created in Merchandising or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
relatedItem	Yes	Collection of Object	Child node holding related-item information for the item.

Table 5-832 RelatedItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
relationshipId	Yes	Number (20)	Unique identifier for each relationship header.
translation	Yes	Collection of Object	Child node containing the translated value of the relationship name.

Table 5-833 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lang	Yes	Number (6)	This field contains the language in which the translated text is maintained.
relationshipName	Yes	String (255)	The translated text of the relationship name.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "100100076",
      "dataLoadingDestination": "RMS",
      "relatedItem": [
        {
          "relationshipId": 10041,
          "translation": [
            {
```

```

    "lang": 20,
    "relationshipName": "ε ν η μ ε ρ ω μ θ η π ε ρ ι γ ρ α φ ρ τ η ς σ χ θ η ς"
  }
]
}
]
}
]
}
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Item Foundation

The following services are included in this functional area:

- [Brand Download Service](#)
 - [Get Brands](#)
- [Brand Upload Service](#)
 - [Manage Brands](#)
- [Differentiator Download Service](#)
 - [Get Differentiator Details](#)
 - [Get Differentiator Details for Provided Differentiator](#)
 - [Get Differentiator Details for Single Differentiator](#)
- [Differentiator Group Download Service](#)

- [Get Differentiator Groups](#)
- [Get Differentiators for Provided Group](#)
- [Differentiator Group Upload Service](#)
 - [Create Differentiator Group Details](#)
 - [Create Differentiator Groups](#)
 - [Delete Differentiator Group Details](#)
 - [Delete Differentiator Groups](#)
 - [Update Differentiator Group Details](#)
 - [Update Differentiator Groups](#)
- [Differentiator Type Download Service](#)
 - [Get Differentiator Type Details for Provided Type](#)
 - [Get Differentiator Types](#)
- [Differentiator Upload Service](#)
 - [Create Differentiators](#)
 - [Delete Differentiators](#)
 - [Update Differentiators](#)
- [Seasons and Phases Download Service](#)
 - [Get Seasons and Phases Service](#)
- [User Defined Attribute \(UDA\) Download Service](#)
 - [REST Endpoint for UDA details](#)
 - [REST Endpoint to Fetch UDA by UDA ID](#)

Brand Download Service

The following services are included in this functional area:

- [Get Brands](#)

Get Brands

Endpoints

/MerchIntegrations/services/item/brands

Functional Area

Items - Item Foundation

Business Overview

This service retrieves all Brand records. This service supports pagination and the page size of each response is controlled by the query parameter 'limit'. This service takes an optional query parameter of brand name to retrieve the specified brand.

Service Type

GET

ReST URL

MerchIntegrations/services/item/brands

Input Parameters

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key (Brand Name)
brandName	No	String	Brand Name
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Response Code: 200 (Success)

Table 5-834 PageResultItemsBrand - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-835 Brand - Object. See list of elements for detail

Element Name	Required	Data Type	Description
brandName	Yes	String (30)	This field contains the brand name.
brandDescription	Yes	String (120)	This field contains the brand description.

Table 5-835 (Cont.) Brand - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDatetime	Yes	dateTime	This field contains the date-time in UTC when this record was created.
lastUpdateDatetime	Yes	dateTime	This field contains the date-time in UTC when this record was updated.

Table 5-836 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "brandName": "ORCL",
      "brandDescription": "Oracle",
      "createDatetime": "2001-12-31T23:59:59.000Z",
      "lastUpdateDatetime": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?limit=1000",
      "rel": "self"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Brand Upload Service

The following services are included in this functional area:

- [Manage Brands](#)

Manage Brands

Functional Area

Items - Item Foundation

Business Overview

This service allows external system to manage brand data. It supports creating, updating and deleting brand records as well as managing translated brand descriptions.

This service accepts brand data as input. If the delete indicator is set to 'N' (or is empty) and the input brand name does not exists in the system, a new brand record will be created. If the brand name already exists, it will update the brand description (if provided) and the provided translation entries.

A value of 'Y' in the delete indicator will delete the brand record if the brand is not attached with any item.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/brands/manage

Input Payload Details

Table 5-837 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of Brand Data.

Table 5-838 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
brandName	Yes	String (30)	This field contains the brand name. This is maintained in upper case.
brandDescription	No	String (120)	This field contains the description of the brand. This field is optional and if provided, brand description will be updated for an existing record. This field is required for new brand creation and any update request where translation records are not being maintained.
deleteInd	No	String (1)	This optional field allows for delete of existing brand record provided the brand is not attached to any item. If this is set to Y (Yes), input brand name must exist in the system. Default value is 'N'
translation	No	Collection of Object	References a collection of translations of the brand description field.

Table 5-839 Translation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
language	Yes	Number (6)	This field contains the language ID. This value must be predefined on the LANG table. This should not be the system primary language id.
brandDescription	No	String (120)	This field contains the translated description of the brand in the provided language id. This field is optional for delete of an existing transaction entry.
deleteInd	No	String (1)	This optional field allows for delete of existing translation entry for brand. If this is set to Y (Yes), the translation entry must exist for the input language id. Default value is 'N'

Table 5-840 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
brandName	Yes	String (30)	This field contains the input brand name for which validation error happened
errorMessage	Yes	String (2000)	This field contains the error message.

Sample Input Message

```
{
  "collectionSize": 1,
```



```
"items": [
  {
    "brandName": "NEW",
    "brandDescription": "New Brand",
    "deleteInd": null,
    "translation": [
      {
        "language": 3,
        "brandDescription": "Nouvelle marque",
        "deleteInd": null
      }
    ]
  }
]
```

Response Code: 200 (Success)
Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-841 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
brandName	Yes	String (30)	This field contains the input brand name for which validation error happened
errorMessage	Yes	String (2000)	This field contains the error message.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "brandName": "NEW",
      "errorMessage": "Delete failed because brand is associated with one or more items"
    }
  ]
}
```

Differentiator Download Service

The following services are included in this functional area:

- [Get Differentiator Details](#)
- [Get Differentiator Details for Provided Differentiator](#)
- [Get Differentiator Details for Single Differentiator](#)

Get Differentiator Details

Endpoints

MerchIntegrations/services/foundation/diffid

MerchIntegrations/services/foundation/diffid/{diffId}

Functional Area

Items – Item Foundation

Business Overview

This service publishes differentiator master data to downstream consuming systems. It publishes only differentiators whose DIFF_TYPE still resolves to a valid row in DIFF_TYPE.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIFF_IDS

Additional business tables: DIFF_TYPE

JSON cache table: MERCHAPI_EXT_DIFF_IDS

JSON generation view: V_MERCHAPI_EXT_DIFF_IDS_JSON

Builds the payload from DIFF_IDS and joins DIFF_TYPE to add the differentiator type description.

Functionally, that means only differentiators whose DIFF_TYPE still resolves to a valid row in DIFF_TYPE are published.

Initial refresh supports both truncate-and-load and rebuild behavior, and rebuild marks cache rows deleted when the differentiator no longer exists in V_MERCHAPI_EXT_DIFF_IDS_JSON.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_DIFF_IDS_JSON and apply only DIFF_IDS rows with a matching DIFF_TYPE row are published.

Mark cache rows deleted when the differentiator no longer exists in V_MERCHAPI_EXT_DIFF_IDS_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIFF_IDS (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on DIFF_IDS.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_DIFF_ID

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_DIFF_IDS_ADHOC_PROCESS ->
MERCHAPI_EXT_DIFF_IDS_ADHOC_JOB

Webhook configuration api name: foundation/diffid

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/diffid

MerchIntegrations/services/foundation/diffid/{diffid}

Input Parameters for MerchIntegrations/services/foundation/diffid

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Diff ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-842 Input Parameter for "MerchIntegrations/services/foundation/diffid/{diffid}"

Parameter Name	Required	Data Type	Description
diffid	Yes	String	Differentiator Id

Table 5-842 (Cont.) Input Parameter for "MerchIntegrations/services/foundation/diffId/{diffId}"

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/diffId/{diffId}

Parameter Name	Required	Data Type	Description
differentiatorId	Yes	String	This field contains a unique identifier of the differentiator
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output "Diff ID"

Table 5-843 PageResultsItemsMerchApiDiffId - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-844 MerchApiDiffId - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for differentiator ID details.
diffId	No	String (10)	This field contains a unique identifier of the differentiator.
diffDescription	No	String (120)	This field contains the description of the differentiator.
diffType	No	String (6)	This field contains a code indicating the type of differentiator, for example C for Color.
diffTypeDescription	No	String (120)	This field contains the description of the differentiator type.
industryCode	No	String (10)	This field is used to hold the unique code used by industry standards to identify the differentiator.
industrySubgroup	No	String (10)	This field is used to hold a sub-grouping code used by industry standards to further identify the differentiator.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
cacheTimestamp	No	dateTime	This field specifies the date and time when the diff ID record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-845 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "diffId": "XYZ01",
      "diffDescription": "diffDesc",
      "diffType": "C",
      "diffTypeDescription": null,
      "industryCode": null,
      "industrySubgroup": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.

Elements	Description
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

```
{
  "items": [
    {
      "action": "INSERT",
      "diffId": "1",
      "diffDescription": "Pot Size",
      "diffType": "POTSZ",
      "industryCode": null,
      "industrySubgroup": null,
      "createDateTime": "2005-02-01T00:00:00.000Z",
      "updateDateTime": "2005-02-01T00:00:00.000Z",
      "cacheTimestamp": "2021-08-31T07:55:38.434Z"
    }
  ],
  "hasMore": false,
  "limit": 1000,
  "count": 1,
  "links": [
    {
      "href": "https://<hostname>:<port>/rgbu-rex-rgbu-dev1-mfcs19-1/
MerchIntegrations/services/foundation/diffid?
limit=10000&since=1970-01-01T00:00:00.001Z&before=2022-03-25T09:17:01.879357+0
0:00",
      "rel": "self"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_IDS	Yes	No	Yes	Yes
MERCHAPI_EXT_DIFF_IDS	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_DIFF_IDS_JSON	Yes	No	No	No

Get Differentiator Details for Provided Differentiator

Endpoints

MerchIntegrations/services/foundation/diffid
 MerchIntegrations/services/foundation/diffid/{diffId}

Functional Area

Items - Item Foundation

Business Overview

This service can be used by external applications to get all or selected diff IDs and its details from Merchandising.

Service Type

GET

ReST URL

/MerchIntegrations/services/foundation/diffid?
 since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
 /MerchIntegrations/services/foundation/diffId/{diffId}

Input Parameters for MerchIntegrations/services/foundation/diffId

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node ID.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.

Input Parameters for MerchIntegrations/services/foundation/diffId/{diffId}

Parameter Name	Required	Data Type	Description
differentiatorId	Yes	String	This field contains a unique identifier of the differentiator

Output - Diff ID

Name	Data Type	Description
action	String	Action for Item details. Valid Values are INSERT, UPDATE and DELETE.
diffId	String	This field contains a unique identifier of the differentiator.
diffDescription	String	This field contains the description of the differentiator.
diffType	String	This field contains a code indicating the type of differentiator, for example C for Color.
industryCode	String	This field is used to hold the unique code used by industry standards to identify the differentiator
industrySubgroup	String	This field is used to hold a sub-grouping code used by industry standards to further identify the differentiator
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when the Diff ID record was last maintained.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```
{
  "items": [
    {
      "action": "INSERT",
      "diffId": "1",
      "diffDescription": "Pot Size",
      "diffType": "POTSZ",
```

```

        "industryCode": null,
        "industrySubgroup": null,
        "createDateTime": "2005-02-01T00:00:00.000Z",
        "updateDateTime": "2005-02-01T00:00:00.000Z",
        "cacheTimestamp": "2021-08-31T07:55:38.434Z"
    }
],
"hasMore": false,
"limit": 1000,
"count": 1,
"links": [
    {
        "href": "https://<hostname>:<port>/rgbu-rex-rgbu-dev1-mfcs19-1/
MerchIntegrations/services/foundation/diffid?
limit=100000&since=1970-01-01T00:00:00.001Z&before=2022-03-25T09:17:01.879357+0
0:00",
        "rel": "self"
    }
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_IDS	Yes	No	Yes	Yes
MERCHAPI_EXT_DIFF_IDS	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_DIFF_IDS_JSON	Yes	No	No	No

Get Differentiator Details for Single Differentiator

This section describes the Diff Detail service.

Business Overview

Diff Detail service allows user to retrieve Diff description for a selected Diff Id.

Service Type

Get

ReST URL

DiffIds/diffIdDetail?diffId={diffId}

Input Parameters

Parameter Name	Required	Description
Diff_Id	Yes	Diff ID

Output

RestDiffIdsRecRDO

Parameter Name	Data Type
industrySubgroup	String
diffGroupDesc	String
diffType	String
diffDesc	String
industryCode	String
diffGroupId	String
diffTypeDesc	String

JSON Structure

```

{
  "industrySubgroup": null,
  "diffGroupDesc": null,
  "diffType": null,
  "diffDesc": "null",
  "industryCode": null,
  "diffGroupId": null,
  "diffTypeDesc": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
DIFF_IDS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Differentiator Group Download Service

The following services are included in this functional area:

- [Get Differentiator Groups](#)
- [Get Differentiators for Provided Group](#)

Get Differentiator Groups

Endpoints

MerchIntegrations/services/foundation/diffgroup

MerchIntegrations/services/foundation/diffgroup/{diffGroupId}

Functional Area

Items – Item Foundation

Business Overview

This service publishes differentiator group master data to downstream consuming systems. The payload includes the differentiator group definition together with its member differentiator IDs and display sequence. This service only publishes differentiator groups that currently have at least one detail row.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIFF_GROUP_HEAD, DIFF_GROUP_DETAIL

Additional business tables: none

JSON cache table: MERCHAPI_EXT_DIFF_GROUP

JSON generation view: V_MERCHAPI_EXT_DIFF_GROUP_JSON

Builds the header from DIFF_GROUP_HEAD and the detail array from DIFF_GROUP_DETAIL.

The view uses an inner join between the header and aggregated detail rows, so only differentiator groups that currently have at least one detail row are published.

Detail-level changes from DIFF_GROUP_DETAIL are treated as updates to the parent differentiator group, so adds, updates, and deletes on detail rows republish the full group payload.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild apply the same publication rule used by V_MERCHAPI_EXT_DIFF_GROUP_JSON: groups publish only when DIFF_GROUP_HEAD has at least one matching DIFF_GROUP_DETAIL row.

Mark cache rows deleted when the diff group no longer exists in V_MERCHAPI_EXT_DIFF_GROUP_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIFF_GROUP (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on DIFF_GROUP_HEAD, DIFF_GROUP_DETAIL.

DIFF_GROUP_DETAIL inserts, updates, and deletes are treated as updates to the parent diff group.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_DIFF_GROUP

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_DIFF_GROUP_ADHOC_PROCESS ->
MERCHAPI_EXT_DIFF_GROUP_ADHOC_JOB

Webhook configuration api name: foundation/diffgroup

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/diffgroup

MerchIntegrations/services/foundation/diffgroup/{diffGroupId}

Input Parameters for MerchIntegrations/services/foundation/diffgroup

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Diff Group ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-846 Input Parameter for "MerchIntegrations/services/foundation/diffgroup/{diffGroupId}"

Parameter Name	Required	Data Type	Description
diffGroupId	Yes	String	Differentiator Group Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/diffgroup/{diffGroupId}

Parameter Name	Required	Data Type	Description
diffGroupId	Yes	String	Differentiator Group ID
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output “ Diff Group Head

Table 5-847 PageResultsItemsMerchApiDiffGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-848 MerchApiDiffGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for differentiator group details.
diffGroupId	No	String (10)	This field contains the ID of the diff group being added or updated.
diffGroupDescription	No	String (120)	This field contains the description of the diff group.

Table 5-848 (Cont.) MerchApiDiffGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffType	No	String (6)	This field contains the code for the type of diff contained in this group, such as S for size.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
details	No	Collection of Object	This list holds attributes for diff group details.
cacheTimestamp	No	dateTime	This field specifies the date and time when the diff group record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-849 DiffGroupDetail - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	No	String (10)	This field contains the ID of the diff that is included in the group.
displaySequence	No	Number(4,0)	This field contains the display sequence for the diff in the group.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).

Table 5-850 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-850 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "diffGroupId": "Y21 Colors",
      "diffGroupDescription": "Year 2021 Colors",
      "diffType": "C",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "details": [
        {
          "diffId": "BLACK",
          "displaySequence": 1,
          "createDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    },
    {
      "hasMore": true,
      "limit": 1000,
      "count": 1000,
      "links": [
        {
          "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
          "rel": "next"
        }
      ]
    }
  ]
}
```

Output – Diff Group Details

Name	Data Type	Description
diffId	String	This field contains the ID of the diff that is included in the group.

Name	Data Type	Description
displaySequence	Number	This field contains the display sequence for the diff in the group.
createDateTime	String	This field holds the record creation date.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

```
{
  "items": [
    {
      "action": "UPDATE",
      "diffGroupId": "BASIC CLRS",
      "diffGroupDescription": "Basic Color Pallet",
      "diffType": "C",
      "createDateTime": "2004-10-12T08:52:47.000Z",
      "updateDateTime": "2004-10-12T08:52:47.000Z",
      "details": [
        {
          "diffId": "BLACK",
          "displaySequence": null,
          "createDateTime": "2004-09-27T00:00:00.000Z"
        },
        {
          "diffId": "WHITE",
          "displaySequence": null,
          "createDateTime": "2005-01-29T00:00:00.000Z"
        }
      ],
      "cacheTimestamp": "2022-07-02T00:15:20.548Z"
    }
  ],
  "hasMore": true,
```

```

    "limit": 1,
    "count": 1,
    "links": [
      {
        "href": "http://<hostname>:<port>/MerchIntegrations/services/
foundation/diffgroup?
limit=1&since=1970-01-01T00:00:00.001Z&before=2022-07-18T23:39:19.734588+00:00
",
        "rel": "self"
      },
      {
        "href": "http://<hostname>:<port>/MerchIntegrations/services/
foundation/diffgroup?
offsetkey=BASIC+CLRS&limit=1&since=1970-01-01T00:00:00.001Z&before=2022-07-18T
23:39:19.734588+00:00",
        "rel": "next"
      }
    ]
  }
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_GROUP	Yes	No	Yes	Yes
MERCHAPI_EXT_DIFF_GROUP	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_DIFF_GROUP_JSON	Yes	No	No	No

Get Differentiators for Provided Group

Endpoints

MerchIntegrations/services/foundation/diffgroup
 MerchIntegrations/services/foundation/diffgroup/{diffGroupId}

Functional Area

Items - Item Foundation

Business Overview

This service can be used by external applications to get all or selected diff groups and its details from Merchandising.

Service Type

GET

ReST URL

```
/MerchIntegrations/services/foundation/diffgroup?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}

/MerchIntegrations/services/foundation/diffgroup/{diffGroupId}
```

Input Parameters for MerchIntegrations/services/foundation/diffgroup

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node ID.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.

Input Parameters for MerchIntegrations/services/foundation/diffgroup/{diffGroupId}

Parameter Name	Required	Data Type	Description
diffGroupId	Yes	String	Differentiator Group ID

Output – Diff Group Head

Name	Data Type	Description
action	String	Action for Item details. Valid Values are INSERT, UPDATE and DELETE.
diffGroupId	String	This field contains the ID of the diff group being added or updated.
diffGroupDescription	String	This field contains the description of the diff group.
diffType	String	This field contains the code for the types of diffs contained in this group, such as S for size.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
details		This List holds list attributes for Diff Group Details
cacheTimestamp	String	This field specifies date and time when Diff Group record was last maintained.

Output – Diff Group Details

Name	Data Type	Description
diffId	String	This field contains the ID of the diff that is included in the group.
displaySequence	Number	This field contains the display sequence for the diff in the group.
createDateTime	String	This field holds the record creation date.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON output:

```
{
  "items": [
    {
      "action": "UPDATE",
      "diffGroupId": "BASIC CLRS",
      "diffGroupDescription": "Basic Color Pallet",
      "diffType": "C",
      "createDateTime": "2004-10-12T08:52:47.000Z",
      "updateDateTime": "2004-10-12T08:52:47.000Z",
      "details": [
        {
          "diffId": "BLACK",
          "displaySequence": null,
          "createDateTime": "2004-09-27T00:00:00.000Z"
        },
        {
          "diffId": "WHITE",
          "displaySequence": null,
          "createDateTime": "2005-01-29T00:00:00.000Z"
        }
      ]
    },
    {
      "cacheTimestamp": "2022-07-02T00:15:20.548Z"
    }
  ]
}
```

```
    ],
    "hasMore": true,
    "limit": 1,
    "count": 1,
    "links": [
      {
        "href": "http://<hostname>:<port>/MerchIntegrations/services/
foundation/diffgroup?
limit=1&since=1970-01-01T00:00:00.001Z&before=2022-07-18T23:39:19.734588+00:00
",
        "rel": "self"
      },
      {
        "href": "http://<hostname>:<port>/MerchIntegrations/services/
foundation/diffgroup?
offsetkey=BASIC+CLRS&limit=1&since=1970-01-01T00:00:00.001Z&before=2022-07-18T
23:39:19.734588+00:00",
        "rel": "next"
      }
    ]
  }
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_GROUP	Yes	No	Yes	Yes
MERCHAPI_EXT_DIFF_GROUP	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_DIFF_GROUP_JSON	Yes	No	No	No

Differentiator Group Upload Service

The following services are included in this functional area:

- [Create Differentiator Group Details](#)
- [Create Differentiator Groups](#)
- [Delete Differentiator Group Details](#)
- [Delete Differentiator Groups](#)
- [Update Differentiator Group Details](#)
- [Update Differentiator Groups](#)

Create Differentiator Group Details

Functional Area

Items - Item Foundation

Business Overview

This service is used to create Differentiator Group details for existing Diff Groups in Merchandising. Successful creation of a Differentiator Group detail depends on if all required fields are present in the message and field specific business validations.

Service Type

POST

ReST URL

MerchIntegrations/services/item/differentiator/group/details/create

Input Payload Details

Table 5-851 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-852 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffGroupId	Yes	String (10)	A unique differentiator group identifier.
details	No	Collection of Object	Description is not available.

Table 5-853 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The identifier of the differentiator contained within the differentiator group. This id must be unique within the diff group. This must be populated when a diff group is being created since a diff group needs at least one diff id. Value must be predefined in the DIFF_IDS table.
displaySequence	No	Number (4)	The order in which the differentiators within the differentiator group are displayed on-line.

Sample Input Message

```
{
  "items": [
    {
      "diffGroupId": "Y21 Colors",
      "details": [
        {
```

```

        "diffId": "BLACK",
        "displaySequence": 1
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Create Differentiator Groups

Functional Area

Items - Item Foundation

Business Overview

This service is used to create Differentiator Groups in Merchandising. Successful creation of a Differentiator Group depends on if all required fields are present in the message and field specific business validations.

Service Type

POST

ReST URL

MerchIntegrations/services/item/differentiator/groups/create

Input Payload Details

Table 5-854 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of differentiator groups.

Table 5-855 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffGroupId	Yes	String (10)	A unique differentiator group identifier.
diffType	Yes	String (6)	A differentiator type which is predefined on the DIFF_TYPE table. Each diff group is composed of one diff type.
diffGroupDescription	Yes	String (120)	The description of the differentiator group.
details	No	Collection of Object	References a collection of differentiators within the differentiator group.

Table 5-856 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The identifier of the differentiator contained within the differentiator group. This id must be unique within the diff group. This must be populated when a diff group is being created since a diff group needs at least one diff id. Value must be predefined in the DIFF_IDS table.
displaySequence	No	Number (4)	The order in which the differentiators within the differentiator group are displayed on-line.

Sample Input Message

```
{
  "items": [
    {
      "diffGroupId": "Y21 Colors",
      "diffType": "C",
      "diffGroupDescription": "Year 2021 Colors",
      "details": [
```



```

        {
            "diffId": "BLACK",
            "displaySequence": 1
        }
    ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Delete Differentiator Group Details

Functional Area

Items - Item Foundation

Business Overview

This service is used to delete existing differentiator group details from Merchandising. For successful deletion, validation is performed to ensure that the provided Differentiator is associated with the Differentiator Group.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/differentiator/group/details/delete

Input Payload Details

Table 5-857 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-858 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffGroupId	Yes	String (10)	A unique differentiator group identifier.
details	No	Collection of Object	Description is not available.

Table 5-859 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The identifier of the differentiator contained within the differentiator group. This id must be unique within the diff group. Value must be predefined in the DIFF_IDS table.

Sample Input Message

```
{
  "items": [
    {
      "diffGroupId": "Y21 Colors",
      "details": [
        {
          "diffId": "BLACK"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
```

```
"message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Delete Differentiator Groups**Functional Area**

Items - Differentiators

Business Overview

This service is used to delete existing Differentiator Groups from Merchandising. For successful deletion, this API will validate that the Differentiator Group is valid and that it is not associated with any items or diff ranges.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/differentiator/groups/delete

Input Payload Details**Delete** - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	References a collection of differentiator groups.

Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffGroupId	Yes	String (10)	A unique differentiator group identifier.

Sample Input Message

```
{
  "items": [
    {
      "diffGroupId": "Y21 Colors"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` is present when the input payload or input parameters do not match the schema definition for this service. The element `businessError` is present when the payload passes the schema validation but an exception is caught while processing the business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Update Differentiator Group Details**Functional Area**

Items - Item Foundation

Business Overview

This service is used to update existing Differentiator Group details in Merchandising. Successful update of a Differentiator Group detail depends on if all required fields are present in the message. Only field that can be updated using this API is Display Sequence.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/differentiator/group/details/update

Input Payload Details

Table 5-860 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-861 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffGroupId	Yes	String (10)	A unique differentiator group identifier.
items	No	Collection of Object	Description is not available.

Table 5-862 Items.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The identifier of the differentiator contained within the differentiator group. This id must be unique within the diff group. This must be populated when a diff group is being created since a diff group needs at least one diff id. Value must be predefined in the DIFF_IDS table.
displaySequence	No	Number (4)	The order in which the differentiators within the differentiator group are displayed on-line.

Sample Input Message

```
{
  "items": [
    {
      "diffGroupId": "Y21 Colors",
      "items": [
        {
          "diffId": "BLACK",
```

```

        "displaySequence": 1
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Update Differentiator Groups

Functional Area

Items - Item Foundation

Business Overview

This service is used to update existing Differentiator Groups in Merchandising. When updating a differentiator group, the group ID must already be present in the Merchandising. For a successful update, all of the required header level information needs to be included in the update, similar to that for creating a new Differentiator Group. However, the diff details should not be included in a header only update. Fields that can be updated using this API include:

- Differentiator type
- Differentiator group description

Service Type

PUT

ReST URL

MerchIntegrations/services/item/differentiator/groups/update

Input Payload Details

Table 5-863 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of differentiator groups.

Table 5-864 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffGroupId	Yes	String (10)	A unique differentiator group identifier.
diffType	Yes	String (6)	A differentiator type which is predefined on the DIFF_TYPE table. Each diff group is composed of one diff type.
diffGroupDescription	Yes	String (120)	The description of the differentiator group.

Sample Input Message

```
{
  "items": [
    {
      "diffGroupId": "Y21 Colors",
      "diffType": "C",
      "diffGroupDescription": "Year 2021 Colors"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Differentiator Type Download Service

The following services are included in this functional area:

- [Get Differentiator Type Details for Provided Type](#)
- [Get Differentiator Types](#)

Get Differentiator Type Details for Provided Type**Endpoints**

MerchIntegrations/services/foundation/difftype

MerchIntegrations/services/foundation/difftype/{diffType}

Functional Area

Items – Item Foundation

Business Overview

This service publishes all differentiator type master data to downstream consuming systems. Diff Types are used to qualify the diff IDs included in other integration. Examples of diff types are size, color, flavor, etc.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIFF_TYPE

Additional business tables: none

JSON cache table: MERCHAPI_EXT_DIFF_TYPE

JSON generation view: V_MERCHAPI_EXT_DIFF_TYPE_JSON

The JSON view is intentionally simple: it reads directly from DIFF_TYPE and publishes the differentiator type code, description, and audit timestamps.

There is no additional functional filter in the view, so every row in DIFF_TYPE is eligible for publication.

Initial refresh reads the full V_MERCHAPI_EXT_DIFF_TYPE_JSON view, supports both truncate-and-load and rebuild modes, and marks cache rows deleted when a diff type no longer exists in the view.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_DIFF_TYPE_JSON and apply none; all DIFF_TYPE rows are eligible.

Mark cache rows deleted when the diff type no longer exists in V_MERCHAPI_EXT_DIFF_TYPE_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIFF_TYPE (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on DIFF_TYPE.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_DIFF_TYPE

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_DIFF_TYPE_ADHOC_PROCESS ->
MERCHAPI_EXT_DIFF_TYPE_ADHOC_JOB

Webhook configuration api name: foundation/difftype

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/difftype

MerchIntegrations/services/foundation/difftype/{diffType}

Input Parameters for Merchintegrations/services/foundation/difftype

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Diff Type)
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-865 Input Parameter for "MerchIntegrations/services/foundation/diffType/{diffType}"

Parameter Name	Required	Data Type	Description
diffType	Yes	String	Differentiator Type
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for Merchintegrations/services/foundation/diffType/{diffType}

Parameter Name	Required	Data Type	Description
diffType	Yes	String	Differentiator Type
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output "Diff ID"

Table 5-866 PageResultsItemsMerchApiDiffType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-866 (Cont.) PageResultsItemsMerchApiDiffType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-867 MerchApiDiffType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for differentiator type details.
diffType	No	String (6)	This field contains a code indicating the type of differentiator, for example C for Color.
diffTypeDescription	No	String (120)	This field holds the description of the diff type (e.g., Color, Size, and so on).
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
cacheTimestamp	No	dateTime	This field specifies the date and time when the record was fetched into the cache for web service publication (UTC).

Table 5-867 (Cont.) MerchApiDiffType - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-868 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "diffType": "C",
      "diffTypeDescription": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	“self” : This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Example JSON Output

```
{
  "items": [
    {
      "action": "INSERT",
      "diffType": "POTSZ",
      "diffTypeDescription": "Pot Size",
      "createDateTime": "2020-11-23T00:00:00.000Z",
      "updateDateTime": "2021-07-13T00:00:00.000Z",
      "cacheTimestamp": "2021-08-31T07:51:43.095Z"
    }
  ],
  "hasMore": false,
  "limit": 10000,
  "count": 1,
  "links": [
    {
      "href": "https://rex.retail.us-phoenix-1.ocs.oc-test.com/rgbu-rex-rgbu-dev1-mfcs19-1/MerchIntegrations/services/foundation/diffType?limit=10000&since=1970-01-01T00:00:00.001Z&before=2022-03-25T09:27:21.798652+00:00",
      "rel": "self"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_TYPE	Yes	No	Yes	Yes
MERCHAPI_EXT_DIFF_TYPE	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_DIFF_TYPE_JSON	Yes	No	No	No

Get Differentiator Types

Endpoints

MerchIntegrations/services/foundation/difftype

MerchIntegrations/services/foundation/difftype/{diffType}

Functional Area

Items – Item Foundation

Business Overview

This service can be used by external applications to get all or selected diff types and its details from Merchandising.

Service Type

GET

ReST URL

/MerchIntegrations/services/foundation/difftype?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}

/MerchIntegrations/services/foundation/difftype/{diffType}

Input Parameters for Merchintegrations/services/foundation/difftype

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for Merchintegrations/services/foundation/difftype/{diffType}

Parameter Name	Required	Data Type	Description
diffType	Yes	String	Differentiator Type
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output – Diff ID

Name	Data Type	Description
action	String	Action for Item details. Valid Values are INSERT, UPDATE and DELETE.
diffType	string	This field contains a code indicating the type of differentiator, for example C for Color.
diffTypeDescription	string	This field holds the description of the diff type (e.g., Color, Size, and so on).
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when Diff Type record was last maintained.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.

Elements	Description
links	“self” : This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Example JSON output:

```
{
  "items": [
    {
      "action": "INSERT",
      "diffType": "POTSZ",
      "diffTypeDescription": "Pot Size",
      "createDateTime": "2020-11-23T00:00:00.000Z",
      "updateDateTime": "2021-07-13T00:00:00.000Z",
      "cacheTimestamp": "2021-08-31T07:51:43.095Z"
    }
  ],
  "hasMore": false,
  "limit": 10000,
  "count": 1,
  "links": [
    {
      "href": "https://rex.retail.us-phoenix-1.ocs.oc-test.com/rgbu-rex-rgbu-dev1-mfcs19-1/MerchIntegrations/services/foundation/difftype?limit=10000&since=1970-01-01T00:00:00.001Z&before=2022-03-25T09:27:21.798652+00:00",
      "rel": "self"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIFF_TYPE	Yes	No	Yes	Yes
MERCHAPI_EXT_DIFF_TYPE	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_DIFF_TYPE_JSON	Yes	No	No	No

Differentiator Upload Service

The following services are included in this functional area:

- [Create Differentiators](#)
- [Delete Differentiators](#)

- [Update Differentiators](#)

Create Differentiators

Functional Area

Items - Item Foundation

Business Overview

This service is used to create Differentiators in Merchandising from an external system. For successful creation of a new differentiator, this service will first validate that all required fields are present in the payload. After that, the following business level validation on the input information are performed before creating the differentiator.

- Verifies if the diff id does not contain white space or underscores
- Verifies if diff id is not already present as a diff id or diff group id
- Verifies if the diff type is a valid value in the DIFF_TYPE table

Service Type

POST

ReST URL

MerchIntegrations/services/item/differentiators/create

Input Payload Details

Table 5-869 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of differentiators.

Table 5-870 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The unique identifier of the differentiator.
diffType	Yes	String (6)	The identifier of the differentiator type. This value must be predefined on the DIFF_TYPE table.
diffDescription	Yes	String (120)	The description of the differentiator.
industryCode	No	String (10)	The unique reference number which represents all possible combinations of sizes according to the National Retail Federation.

Table 5-870 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
industrySubgroup	No	String (10)	The unique reference number to represent a sub-grouping code used by industry standards to further identify the differentiator. For example, in the US, the National Retail Federation uses a subgroup for colors (e.g., purple is defined as 500; dark purple represents a range from 501 - 509, medium purple represents a range from 510 - 519, bright purple represents a range from 520 - 529, etc.).

Sample Input Message

```
{
  "items": [
    {
      "diffId": "XYZ02",
      "diffType": "C",
      "diffDescription": "XYZ02 diffDesc",
      "industryCode": null,
      "industrySubgroup": null
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",

```

```

        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

Delete Differentiators

Functional Area

Items - Differentiators

Business Overview

This service is used to delete existing Differentiator from Merchandising. For a successful deletion, this service first validates that all required fields are present in the message and if Diff id to be deleted exists in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/item/differentiators/delete

Input Payload Details

Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	References a collection of differentiators.

Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The unique identifier for the differentiator.

Sample Input Message

```

{
  "items": [
    {
      "diffId": "XYZ02"
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` is present when the input payload or input parameters do not match the schema definition for this service. The element `businessError` is present when the payload passes the schema validation but an exception is caught while processing the business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Update Differentiators

Functional Area

Items - Item Foundation

Business Overview

This service is used to update existing Differentiator in Merchandising. For a successful update of an existing differentiator, this service will first validate that all required fields are present in the payload and business level validations are met. These business level validations are similar to those performed at the time of creation. Please refer to Create Differentiators service description for more details.

Service Type

PUT

ReST URL

MerchIntegrations/services/item/differentiators/update

Input Payload Details

Table 5-871 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	References a collection of differentiators.

Table 5-872 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
diffId	Yes	String (10)	The unique identifier of the differentiator.
diffDescription	Yes	String (120)	The description of the differentiator.
industryCode	No	String (10)	The unique reference number which represents all possible combinations of sizes according to the National Retail Federation.
industrySubgroup	No	String (10)	The unique reference number to represent a sub-grouping code used by industry standards to further identify the differentiator. For example, in the US, the National Retail Federation uses a subgroup for colors (e.g., purple is defined as 500; dark purple represents a range from 501 - 509, medium purple represents a range from 510 - 519, bright purple represents a range from 520 - 529, etc.).

Sample Input Message

```
{
  "items": [
    {
      "diffId": "XYZ02",
      "diffDescription": "XYZ02 diffDesc UPdate",
      "industryCode": null,
      "industrySubgroup": null
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Seasons and Phases Download Service

The following services are included in this functional area:

- [Get Seasons and Phases Service](#)

Get Seasons and Phases Service

Endpoints

/MerchIntegrations/services/item/foundation/seasons

Functional Area

Items - Item Foundation

Business Overview

This service returns all seasons and all phases within a season. This service supports pagination, and the page size of each response is controlled by the query parameter 'limit'. This service additionally supports two optional query parameters.

season - Include season id as a query parameter to restrict the response to a specific season.

status - This optional parameter allows filtering the response based on season start and/or end date. The comparison uses current server date. Following values are supported.

OLD - Retrieves seasons having end date less than system date

ACTIVE - Retrieves seasons having system date between start date (inclusive) and end date (inclusive)

FUTURE - Retrieves seasons having start date greater than system date

ACTIVE_FUTURE - Retrieves seasons having end date greater than or equal to system date

OLD_ACTIVE - Retrieves seasons where start date is less than or equal to system date

Service Type

GET

ReST URL

MerchIntegrations/services/item/foundation/seasons

Input Parameters for MerchIntegrations/services/foundation/seasons

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset key
season	No	String (3)	Season Id
status	No	String	Status
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output â€œ Season

Table 5-873 PageResultItemsSeason - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-874 Season - Object. See list of elements for detail

Element Name	Required	Data Type	Description
season	No	Number	This field contains the unique identifier for season
description	No	String	This field contains the description associated with the season
startDate	No	date	This field contains the starting date for the season
endDate	No	date	This field contains the ending date for the season
phases	No	Collection of Object	List of Phases associated with the season

Table 5-875 Phases - Object. See list of elements for detail

Element Name	Required	Data Type	Description
phase	No	Number	This field contains the identifier for phase that is associated to a season. The combination of season id and phase id makes a phase unique.
description	No	String	This field holds the description of the phase
startDate	No	date	This field contains the starting date for the phase
endDate	No	date	This field contains the ending date for the phase

Table 5-876 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-876 (Cont.) LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "season": 10,
      "description": "Spring 2023 Collections",
      "startDate": "2001-12-31",
      "endDate": "2001-12-31",
      "phases": [
        {
          "phase": 2,
          "description": "Spring Break/Easter 2023",
          "startDate": "2001-12-31",
          "endDate": "2001-12-31"
        }
      ]
    },
    {
      "hasMore": true,
      "limit": 1,
      "count": 1,
      "links": [
        {
          "href": "https://<server>:<port>/Application/services/resources?limit=1000",
          "rel": "self"
        }
      ]
    }
  ]
}
```

Output – Phases

Name	Data Type	Description
phase	Number	This field contains the identifier for phase that is associated with a season. The combination of season ID and phase ID makes a phase unique.
description	String	This field holds the description of the phase
startDate	Date	This field contains the starting date for the phase. Format yyyy-MM-dd

Name	Data Type	Description
endDate	Date	This field contains the ending date for the phase. Format yyyy-MM-dd

Elements in JSON Output

Elements	Description
items	The first element is named <code>items</code> and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on the <code>limit</code> value. If the total rows are more than then limit defined, then <code>hasMore</code> is set to <code>true</code> , otherwise it is <code>false</code> .
limit	This is used to limit the rows to control the paging of result data. Based on the <code>limit</code> , <code>next</code> and <code>previous</code> links are displayed. For example: If <code>limit</code> is set to 25, internally it tries to fetch 26 rows. If 26th row exists, then 25 rows are returned along with a <code>next</code> link; otherwise, no links are added to the JSON response.
links	<code>self</code> : This contains a URI pointing to the current page. <code>next</code> : This contains a URI pointing to the next page of results.

Example JSON Output

```
{
  "items": [
    {
      "season": 10,
      "description": "Spring 2023 Collections",
      "startDate": "2023-03-20",
      "endDate": "2023-06-21",
      "phases": [
        {
          "phase": 2,
          "description": "Spring Break/Easter 2023",
          "startDate": "2023-04-03",
          "endDate": "2023-04-16"
        }
      ]
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "https Server:port/MerchIntegrations/services/item/foundation/seasons?limit=1",
      "rel": "self"
    }
  ]
}
```

```
    },
    {
      "href": "https Server:port/MerchIntegrations/services/item/foundation/
seasons?offsetkey=10&limit=1",
      "rel": "next"
    }
  ]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SEASONS	Yes	No	No	No
PHASES	Yes	No	No	No

User Defined Attribute (UDA) Download Service

The following services are included in this functional area:

- [REST Endpoint for UDA details](#)
- [REST Endpoint to Fetch UDA by UDA ID](#)

REST Endpoint for UDA details

Endpoints

MerchIntegrations/services/foundation/uda

MerchIntegrations/services/foundation/uda/{udaId}

Functional Area

Items – Item Foundation

Business Overview

This service publishes user-defined attribute master data to downstream consuming systems. It publishes the UDA definition for date, free-form, and list-of-values UDAs, and list-of-values UDAs also include their configured value list. Functionally, there is no additional filter in the view, so every row in UDA is eligible for publication.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: UDA

Additional business tables: UDA_VALUES

JSON cache table: MERCHAPI_EXT_UDA

JSON generation view: V_MERCHAPI_EXT_UDA_JSON

The JSON view reads the UDA header from UDA and left-joins an aggregated UDA_VALUES array to populate udaListOfValues for list-of-values UDAs.

Functionally, there is no additional filter in the view, so every row in UDA is eligible for publication.

Date, free-form, and list-of-values UDAs are all included; only the LOV-style UDAs contribute rows to the udaListOfValues array.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_UDA_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the UDA no longer exists in V_MERCHAPI_EXT_UDA_JSON.

ICL (Integration Change Log) table: ICL_RMS_UDA (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on UDA.

Queue logic distinguishes UDA versus UDA_VALUES changes through ICL_TABLE_NAME.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_UDA

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job: MERCHAPI_DELTA_EXT_UDA_ADHOC_PROCESS - > MERCHAPI_EXT_UDA_ADHOC_JOB

Webhook configuration api name: foundation/uda

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/uda

MerchIntegrations/services/foundation/uda/{udald}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (UDA Id)
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-877 Input Parameter for "MerchIntegrations/services/foundation/uda/{udaId}"

Parameter Name	Required	Data Type	Description
udaId	Yes	Number	UDA Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-878 PageResultsItemsMerchApiUda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-879 MerchApiUda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for UDA details.
udaId	No	Number(5,0)	This field contains a unique number identifying the user-defined attribute.
udaDescription	No	String (120)	This field contains a description of the user-defined attribute.
displayType	No	String (2)	This field contains the display type, which is how the UDA values will be displayed to the user. Valid values are DT - Date, FF - Free Form, and LV - List of Values.
dataType	No	String (12)	This field contains the data type of any valid values for the UDA. Valid values are Number, Alphanumeric, and Date.
dataLength	No	Number(3,0)	This field contains the data length of any valid values for the UDA.
singleValueInd	No	String (1)	This field indicates whether or not the UDA should be constrained to having at most one value. Valid values are Y and N.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
udaListOfValues	No	Collection of Object	This list holds the UDA list of values.
cacheTimestamp	No	dateTime	This field specifies the date and time when the UDA record was fetched into the cache for web service publication (UTC).

Table 5-879 (Cont.) MerchApiUda - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-880 UdaValues - Object. See list of elements for detail

Element Name	Required	Data Type	Description
udaValue	No	Number(5,0)	This field contains a unique number identifying the user-defined attribute value for the UDA. A UDA can have multiple values.
udaValueDescription	No	String (250)	This field contains a description of the user-defined attribute value.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).

Table 5-881 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
```

```

"udaId": 201,
"udaDescription": null,
"displayType": "FF",
"dataType": null,
"dataLength": 1,
"singleValueInd": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"udaListOfValues": [
{
"udaValue": 1,
"udaValueDescription": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
MERCHAPI_EXT_UDA	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_UDA_JSON	Yes	No	No	No
ICL_RMS_UDA	Yes	No	Yes	Yes

REST Endpoint to Fetch UDA by UDA ID

Endpoints

```

MerchIntegrations/services/foundation/uda
MerchIntegrations/services/foundation/uda/{udaId}

```


Functional Area

Items – Item Foundation

Business Overview

This service can be used by external applications to get all or selected user defined attributes and its details from Merchandising.

Service Type

GET

ReST URL

/MerchIntegrations/services/foundation/uda?
offsetkey={offsetkey}&limit={limit}&since={since}&before={before}

/MerchIntegrations/services/foundation/uda/{udaId}

Input Parameters

Table 5-882 /MerchIntegrations/services/foundation/uda

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is a Uda Id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.

Table 5-883 /MerchIntegrations/services/foundation/uda/{udaId}

Parameter Name	Required	Data Type	Description
udaId	Yes	Number	Number identifying the User Defined Attribute.

Output

Name	Data Type	Description
action	String	Action for UDA details. Valid Values are INSERT, UPDATE and DELETE.
udaId	Number	Number identifying the User Defined Attribute.
udaDescription	String	Description of the User Defined Attribute.
displayType	String	This field contains the display type (how the UDA values will be displayed to the user) for the given UDA. The valid values are DT-Date, FF - Free From, LV - List of values.

Name	Data Type	Description
dataType	String	This field contains the data type of any valid values for the UDA. Valid values are Number, Alphanumeric and Date.
dataLength	Number	This field contains the data length of any valid values for the UDA.
singleValueInd	String	This field indicates whether or not the UDA should be constrained to having at most one value. Valid values are Y and N.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
udaListOfValues		This list holds the UDA list of values.
cacheTimestamp	String	This field specifies date and time when the UDA record was last maintained.

Table 5-884 udaListOfValues

Name	Data Type	Description
action	String	Action for UDA Value details. Valid Values are INSERT, UPDATE and DELETE.
udaValue	Number	This field contains a unique number identifying the User Defined Attribute value for the UDA. A UDA can have multiple values.
udaValueDescription	String	Description of the User Defined Attribute Values.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.

Table 5-885 Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than the limit defined, then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists, then 25 rows are returned along with a "next" link else no links would be added to the JSON response.

Table 5-885 (Cont.) Elements in JSON Output

Elements	Description
links	“self”: This contains URI pointing to current page. “next”: This contains URI pointing to the next page of results.

Example JSON input:

```
{
  "items": [
    {
      "action": null,
      "udaId": 23,
      "udaDescription": "NEW_UDA_MATERIAL",
      "displayType": "LV",
      "dataType": "ALPHA",
      "dataLength": 250,
      "singleValueInd": "N",
      "createDateTime": "2021-10-04T08:40:53.000Z",
      "updateDateTime": "2021-10-04T08:40:53.000Z",
      "udaListOfValues": [
        {
          "udaValue": 3,
          "udaValueDescription": "AUTO_ADMIN_Nylon",
          "createDateTime": "2021-10-04T08:40:53.000Z",
          "updateDateTime": "2021-10-04T08:40:53.000Z"
        },
        {
          "udaValue": 2,
          "udaValueDescription": "AUTO_ADMIN_Silk_Cotton",
          "createDateTime": "2021-10-04T08:40:53.000Z",
          "updateDateTime": "2021-10-04T08:40:53.000Z"
        },
        {
          "udaValue": 1,
          "udaValueDescription": "AUTO_ADMIN_100_Synthetic",
          "createDateTime": "2021-10-04T08:40:53.000Z",
          "updateDateTime": "2021-10-04T08:40:53.000Z"
        }
      ],
      "cacheTimestamp": "2021-10-04T08:50:56.868Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
MERCHAPI_EXT_UDA	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_UDA_JSON	Yes	No	No	No
ICL_RMS_UDA	Yes	No	Yes	Yes

Merchandise Hierarchy

The following services are included in this functional area:

- [Class Download Service](#)
 - [Get Class Details for Provided Class](#)
 - [Get Classes](#)
- [Class Upload Service](#)
 - [Create Class](#)
 - [Delete Class](#)
 - [Update Class](#)
- [Company Upload Service](#)
 - [Update Company](#)
- [Department Download Service](#)
 - [Get Department Details](#)
 - [Get Department Details for Provided Department](#)
 - [Get Department Name](#)
 - [Get Department Tax Details](#)
 - [Get Departments](#)
- [Department Upload Service](#)
 - [Create Department](#)
 - [Create Department Taxes](#)
 - [Create Department Up Charge](#)
 - [Delete Department](#)
 - [Delete Department Up Charge](#)
 - [Update Department](#)
 - [Update Department Taxes](#)
 - [Update Department Up Charge](#)
- [Copy Department Up Charges](#)
- [Division Download Service](#)

- [Get Division Details for Provided Division](#)
 - [Get Divisions](#)
- [Division Upload Service](#)
 - [Create Division](#)
 - [Delete Division](#)
 - [Update Division](#)
- [Group Download Service](#)
 - [Get Group for Provided Group](#)
 - [Get Groups](#)
- [Group Upload Service](#)
 - [Create Group](#)
 - [Delete Group](#)
 - [Update Group](#)
- [Merchandise Hierarchy Download Service](#)
 - [Get Merchandise Hierarchy](#)
- [Subclass Download Service](#)
 - [Get Subclass Details for Provided Subclass](#)
 - [Get Subclasses](#)
- [Subclass Upload Service](#)
 - [Create Subclass](#)
 - [Delete Subclass](#)
 - [Update Subclass](#)
- [Reclassification](#)
 - [Item Reclassification Download Service](#)
 - [Item Reclassification Upload Service](#)
 - [Merchandise Hierarchy Reclassification Upload Service](#)

Copy Department Up Charges

Functional Area

Merchandise Hierarchy

Business Overview

This service allows the user to upload a request to copy the department up charges from one location to another for all departments, all departments in a group or a single department.

Valid Apply To types are defined under the Department Up Charge Apply To (DUAT) code type.

The Copy From and Copy To locations can be a store, virtual warehouse, importer, exporter, internal finisher, or external finisher.

The request to copy up charges will be staged and later processed through asynchronous processing.

Up charges will be copied based on the copy method. Copy methods are defined under the Department Up Charges Copy Methods (DUCM) code type and include:

Full Replace (FR)

For each department specified by the Apply To, this copy method deletes all shipping routes and associated up charges from the Copy To Location, and copies over all routes/up charges from the Copy From Location.

Copy Missing Shipping Routes (CM)

For each department specified by the Apply To, this copy method preserves existing shipping routes and up charges already defined for the Copy To Location, and copies missing shipping routes and associated up charges from the Copy From Location to the Copy To Location.

Merge Shipping Routes / Replace Up Charges (MRRU)

For each department specified by the Apply To, this copy method keeps existing routes on Copy To Location. Where the same route exists for the Copy From Location, the system deletes up charges from the Copy To Location and replaces them with up charges from the Copy From Location for that same route. Lastly, the process copies missing routes and associated up charges from the Copy From Location to the Copy To Location.

Merge Shipping Routes / Add Missing Up Charges (MRAU)

For each department specified by the Apply To, this copy method keeps existing routes defined for Copy To Location and keeps existing up charges, but adds missing charges from the Copy From Location for the same route. Then the process copies missing routes and associated up charges from the Copy From Location to the Copy To Location.

Responses:

Upon successfully staging the copy request, the service will return success status.

If there are validation errors the entire payload will be rejected, the service will return a failure status with the error message.

The service allows to stage a copy request which is duplicate at “copy from location, copy to location and department” level. In this case, the service will return success status with a warning message. The background processing, which processes the data from stage table and copies the up charges to the copy to location, will process the latest request and will ignore the previous duplicate requests.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/upCharge/copy

Availability During Nightly Batch Cycle

This service will not be available when batches that affect costs are in-progress.

Input Payload Details

Copy - Object. See list of elements for detail

Items - Object. See list of elements for detail

CopyError - Object. See list of elements for detail

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "fromLocationType": "S",
      "copyFromLocation": 1521,
      "whereCopyFrom": "E",
      "toLocationType": "S",
      "copyToLocation": 1111,
      "copyMethod": "FR",
      "applyToType": "D",
      "applyTo": 1810
    }
  ]
}
```

Response Code: 200 (Success)

CopyResponse - Object. See list of elements for detail

CopyUpchargesResponse - Object. See list of elements for detail

Sample Response Message

```
{
  "copyUpchargesResponse": [
    {
      "fromLocationType": "S",
      "copyFromLocation": 1521,
      "whereCopyFrom": "E",
      "toLocationType": "S",
      "copyToLocation": 1111,
      "copyMethod": "FR",
      "applyToType": "D",
      "applyTo": 1810,
      "warning": "String"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

CopyError - Object. See list of elements for detail

Sample Error Message

```
{
  "status": "ERROR",
```

```
"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
],
"businessError": [
{
"fromLocationType": "S",
"copyFromLocation": 1521,
"whereCopyFrom": "E",
"toLocationType": "S",
"copyToLocation": 1111,
"copyMethod": "FR",
"applyToType": "D",
"applyTo": 1810,
"errorMessages": [
"String"
]
}
]
}
```

Class Download Service

The following services are included in this functional area:

- [Get Class Details for Provided Class](#)
- [Get Classes](#)

Get Class Details for Provided Class

Endpoints

- MerchIntegrations/services/foundation/merchhier/division/{division}
- MerchIntegrations/services/foundation/merchhier/division
- MerchIntegrations/services/foundation/merchhier/groups/{group}
- MerchIntegrations/services/foundation/merchhier/groups
- MerchIntegrations/services/foundation/merchhier/deps/{dept}
- MerchIntegrations/services/foundation/merchhier/deps
- MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}
- MerchIntegrations/services/foundation/merchhier/class
- MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}
- MerchIntegrations/services/foundation/merchhier/subclass

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy class data to downstream consuming systems. It is the class-level projection of the shared merchandise hierarchy cache and publishes the current class definition together with class name, department, class VAT indicator, inventory-layer identifier type, the unique class identifier, and class custom flex attributes.

The different merchandise level API shares the same underlying technical components. Refer to api documentation of '/foundation/merchhier/deps' for additional details.

Table 5-886 Input Parameters

Parameter Name	Required	Data Type	Description
dept	No	String (4)	Department
class	No	String (4)	Class

Table 5-886 (Cont.) Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-887 Input Parameter for "MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}"

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

```
/MerchIntegrations/services/foundation/merchhier/division?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
```

```
/MerchIntegrations/services/foundation/merchhier/division/{division}
```

ReST URLs for Groups

```
/MerchIntegrations/services/foundation/merchhier/groups?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
```

```
/MerchIntegrations/services/foundation/merchhier/groups/{group}
```

ReST URLs for Department

```
/MerchIntegrations/services/foundation/merchhier/deps?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
```

```
/MerchIntegrations/services/foundation/merchhier/deps/{dept}
```

ReST URLs for Class

```
MerchIntegrations/services/foundation/merchhier/class
```

```
MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}
```

ReST URLs for Subclass

```
/MerchIntegrations/services/foundation/merchhier/subclass?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}&subclass={subclass}
```

```
/MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}
```

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.

Parameter Name	Required	Data Type	Description
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-888 PageResultsItemsMerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-889 MerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for class details.

Table 5-889 (Cont.) MerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
className	No	String (120)	This field contains the name of the class which, along with the class number, identifies the class.
classVatInd	No	String (1)	This field indicates whether retail is displayed and held with or without VAT for items within the class. Valid values are Y and N.
dept	No	Number(4,0)	This field contains the number of the department of which the class is a member.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the class in merchandise hierarchy.

Table 5-889 (Cont.) MerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the class level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-890 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-891 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-891 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```

{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "class": 1,
      "className": "className",
      "classVatInd": "Y",
      "dept": 3041,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "inventoryLayerIdentifierType": null,
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}

```

Output - Groups

Name	Data Type	Description
action	String	Action for group details. Valid Values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
group	Number	This field contains the number which uniquely identifies the group.
groupName	String	This field contains the description which, along with the group number, identifies the group.
Buyer	Number	This field contains the number of the buyer associated with the division.
merchantId	String	This field contains the number of the merchandiser associated with the division.
division	Number	This field contains the number which uniquely identifies the division of the company.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.

Output - Department

Name	Data Type	Description
action	String	Action for department details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
dept	Number	This field contains the number which uniquely identifies the department.
deptName	String	This field contains the description which identifies the department.
buyer	Number	This field contains the number of the buyer associated with the division.
purchaseType	Number	This field Contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment, or concession.
totalMarketAmount	Number	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
merchantId	String	This field contains the number of the merchandiser associated with the division.
group	Number	This field contains the number which uniquely identifies the group.
budgetMarkup	Number	This field contains the number of the group in which the department exists.

Name	Data Type	Description
profitCalcType	Number	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	String	This field contains the code which determines how markup is calculated in this department.
othCalcType	String	This field contains the code which determines how markup is calculated in this department.
budgetedIntakePercent	Number	This field contains maximum average counter will hold the maximum count of days with acceptable data to include in an average for items within the department.
deptVatIncludeInd	String	This field is no longer used by Merchandising and will always be defaulted to Y.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at department level.

Output - Class

Table 5-892 PageResultsItemsMerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-893 MerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for class details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
className	No	String (120)	This field contains the name of the class which, along with the class number, identifies the class.
classVatInd	No	String (1)	This field indicates whether retail is displayed and held with or without VAT for items within the class. Valid values are Y and N.
dept	No	Number(4,0)	This field contains the number of the department of which the class is a member.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.

Table 5-893 (Cont.) MerchApiClass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the class in merchandise hierarchy.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the class level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-894 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-895 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "class": 1,
      "className": "className",
      "classVatInd": "Y",
      "dept": 3041,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "inventoryLayerIdentifierType": null,
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Subclass

Name	Data Type	Description
action	String	Action for subclass details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
subclass	Number	This field Contains the number which uniquely identifies the subclass within the department and class.
subclassName	String	This field Contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	Number	Contains the number which uniquely identifies the class within the department.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueSubclassId	Number	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.

Output - Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.

Name	Data Type	Description
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```


Groups

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Department

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Class

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Subclass

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Get Classes

Endpoints

MerchIntegrations/services/foundation/merchhier/division/{division}

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/groups/{group}

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/deps/{dept}

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

MerchIntegrations/services/foundation/merchhier/class

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

MerchIntegrations/services/foundation/merchhier/subclass

Functional Area

Merchandise Hierarchy

Business Overview

This service can be used by external applications to get all or selected hierarchy levels and its details from Merchandising.

Service Type

GET

ReST URLs for Division

/MerchIntegrations/services/foundation/merchhier/division?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}

/MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

```
/MerchIntegrations/services/foundation/merchhier/groups?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
```

```
/MerchIntegrations/services/foundation/merchhier/groups/{group}
```

ReST URLs for Department

```
/MerchIntegrations/services/foundation/merchhier/deps?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}
```

```
/MerchIntegrations/services/foundation/merchhier/deps/{dept}
```

ReST URLs for Class

```
/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}
```

```
/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}
```

ReST URLs for Subclass

```
/MerchIntegrations/services/foundation/merchhier/subclass?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}&subclass={subclass}
```

```
/MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}
```

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/
{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Name	Data Type	Description
action	String	Action for division details. Valid Values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
division	Number	This field contains the number which uniquely identifies the division of the company.
divisionName	String	This field contains the name which, along with the division number, identifies the division of the company.
buyer	Number	This field contains the number of the buyer associated with the division.
merchantId	String	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	Number	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	Number	This field contains the number which uniquely identifies the Company.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.

Output - Groups

Name	Data Type	Description
action	String	Action for group details. Valid Values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
group	Number	This field contains the number which uniquely identifies the group.

Name	Data Type	Description
groupName	String	This field contains the description which, along with the group number, identifies the group.
Buyer	Number	This field contains the number of the buyer associated with the division.
merchantId	String	This field contains the number of the merchandiser associated with the division.
division	Number	This field contains the number which uniquely identifies the division of the company.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.

Output - Department

Name	Data Type	Description
action	String	Action for department details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
dept	Number	This field contains the number which uniquely identifies the department.
deptName	String	This field contains the description which identifies the department.
buyer	Number	This field contains the number of the buyer associated with the division.
purchaseType	Number	This field Contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment, or concession.
totalMarketAmount	Number	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
merchantId	String	This field contains the number of the merchandiser associated with the division.
group	Number	This field contains the number which uniquely identifies the group.
budgetMarkup	Number	This field contains the number of the group in which the department exists.
profitCalcType	Number	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	String	This field contains the code which determines how markup is calculated in this department.
otbCalcType	String	This field contains the code which determines how markup is calculated in this department.

Name	Data Type	Description
budgetedIntakePercent	Number	This field contains maximum average counter will hold the maximum count of days with acceptable data to include in an average for items within the department.
deptVatIncludeInd	String	This field is no longer used by Merchandising and will always be defaulted to Y.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at department level.

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Name	Data Type	Description
action	String	Action for subclass details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
subclass	Number	This field Contains the number which uniquely identifies the subclass within the department and class.
subclassName	String	This field Contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	Number	Contains the number which uniquely identifies the class within the department.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueSubclassId	Number	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.

Output - Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.

Name	Data Type	Description
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Groups

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Department

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Class

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Subclass

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Class Upload Service

The following services are included in this functional area:

- [Create Class](#)
- [Delete Class](#)
- [Update Class](#)

Create Class

Functional Area

Merchandise Hierarchy

Business Overview

The merchandise hierarchy allows the retailer to create the relationships that are necessary to support the product management structure of a company. This hierarchy reflects a classification of merchandise into multi-level descriptive categorizations to facilitate the planning, tracking, reporting, and management of merchandise within the company. This service may be used to create the class element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information.

The merchandise hierarchy must be created from the highest-level down. This means a department is associated with a group; a class is associated with a department; and a subclass

is associated with department/class combination because classes are not unique across departments.

When a new class is created, this service will first validate that all the required fields are present in the request payload. Business level validation on the input information will be performed, which checks whether the class already exists. If the class does not exist, the class is created. Active custom flex attributes that are associated with a class can also be created through this service.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/class/create

Input Payload Details

Table 5-896 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
class	Yes	Number (4)	The unique number of the class. This field cannot be modified.
className	Yes	String (120)	The name of the class.
dept	Yes	Number (4)	The number of the department which contains the class. This value must be predefined on the DEPS table. This value cannot be modified.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
invLayerIdentifierType	No	String (6)	This field holds the type of identifier that will be used across systems to manage and identify inventory layers at class level.

Table 5-897 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "class": 1,
  "className": "className",
  "dept": 3041,
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": "2001-12-31"
    }
  ],
  "invLayerIdentifierType": "L"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Class

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to delete the class element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a class is deleted, this service will add the class to a purging staging table for processing in the Daily Purge of Foundation Data process.

Note: Associated custom flex attributes are deleted when the class is deleted.

For more details on Class, see the Create Class service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/class/delete

Input Payload Details

Table 5-898 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
class	Yes	Number (4)	The unique number of the class.
dept	Yes	Number (4)	The ID of the department which which contains the class.

Sample Input Message

```
{
  "class": 1,
  "dept": 3041
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Class

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to update the class element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a class is updated, this service first validates that all the required fields are present in the request payload. Business level validation on the input information will be performed, which verifies if the class to be updated already exists. If class already exists, the class details are updated. Active custom flex attributes can also be updated for the class.

For more details on Class, see the Create Class service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/class/update

Input Payload Details

Table 5-899 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
class	Yes	Number (4)	The unique number of the class. This field cannot be modified.
className	Yes	String (120)	The name of the class.
dept	Yes	Number (4)	The number of the department which contains the class. This value must be predefined on the DEPS table. This value cannot be modified.

Table 5-899 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
invLayerIdentifierType	No	String (6)	This field holds the type of identifier that will be used across systems to manage and identify inventory layers at class level.

Table 5-900 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "class": 1,
  "className": "className",
  "dept": 3041,
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": "2001-12-31"
    }
  ],
  "invLayerIdentifierType": "L"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Company Upload Service

The following services are included in this functional area:

- [Update Company](#)

Update Company

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to update the company element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information.

When a company is updated, this service will first validate that all the required fields are present in the request payload. Business level validation on the input information will be performed, which verifies if the company ID to be updated already exists. If the company already exists, the details of the company are updated.

Service Type

PUT

ReST URL

`MerchIntegrations/services/merchandiseHierarchy/company/update`

Input Payload Details

Table 5-901 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
company	Yes	Number (4)	The unique number which identifies the company for which the system is running. This value cannot be modified.
companyName	Yes	String (120)	The name of the company for which the system is running.
add1	Yes	String (240)	The first line of the company headquarters address.
add2	No	String (240)	The second line of the company headquarters address.
add3	No	String (240)	The third line of the company headquarters address.
city	Yes	String (120)	The city the company headquarters.
state	No	String (3)	The abbreviation of the state of the company headquarters. This value must be predefined on the STATE table.
countryCode	Yes	String (3)	The abbreviation of the country of the company headquarters. This value must be predefined on the COUNTRY table.
postalCode	No	String (30)	The postal zip code of the company headquarters.

Sample Input Message

```
{
  "company": 1000,
  "companyName": "companyName",
  "add1": "101 First Street",
  "add2": "201 Second Street",
  "add3": "301 Third Street",
  "city": "Minneapolis",
  "state": "MN",
  "countryCode": "US",
  "postalCode": "55555-1234"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Department Download Service

The following services are included in this functional area:

- [Get Department Details](#)
- [Get Department Details for Provided Department](#)
- [Get Department Name](#)
- [Get Department Tax Details](#)
- [Get Departments](#)

Get Department Details

Business Overview

This service retrieves departments with ID or name matching search string.

Service Type

GET

ReST URL

/Common/departmentSearch?
searchString={searchString}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description	Valid values
searchString	Yes	search string for department Id or Name	

Parameter Name	Required	Description	Valid values
PageSize	No	Maximum number of records to retrieve per page	
PageNumber	No	Result page to retrieve	

Output

Table 5-902 MerchHierDeptRDO

Parameter Name	Data Type
department	BigDecimal
departmentName	String

Table 5-903 PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure:

```
{
  "type": "paginationRDO",
  "totalRecordCount": 3512,
  "hyperMediaContent": {},
  "links" : [],
  "results": [{
    "departmentId": 3252,
    "departmentDescription": "some description"
  }]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_DEPS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Department Details for Provided Department

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table

MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:

MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-904 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-905 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-906 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-907 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-907 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-908 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-908 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-909 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-909 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-910 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-910 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-911 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-912 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-913 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-913 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-913 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-914 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-914 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-915 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,
    }
  ]
}
```

```

"budgetedIntakePercent": 3,
"deptVatIncludeInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"inventoryLayerIdentifierType": null,
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-916 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-917 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-917 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-917 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-918 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-919 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
      limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
    }
  ]
}
```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
      }
    ],
    "hasMore": false,
    "limit": 1,
    "count": 1,
    "links": null
  }

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Get Department Name

Business Overview

This service retrieves departments' name of input IDs.

Service Type

GET

ReST URL

/Common/departmentsLoad?departments={departments}

Input Parameters

Parameter Name	Required	Description	Valid values
departments	Yes	Comma separated values for Departments	NA

Output

Table 5-920 MerchHierDeptRDO

Parameter Name	Data Type
department	BigDecimal
departmentName	String

JSON Structure:

```
{
  "departmentId": 3252,
  "departmentDescription": "some description"
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_DEPS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Department Tax Details

This section describes the VAT Detail service.

Business Overview

VAT Detail service allows user to retrieve VAT information for a selected department.

Service Type

Get

ReST URL

Vat/vatDetail?department={departmentNumber}

Input Parameters

Parameter Name	Required	Description
Department	Yes	Department ID

Output

RestVatRecRDO

Parameter Name	Data Type
vatRegion	BigDecimal
vatRegionName	String
vatRegionType	String
vatType	String
vatCode	String
vatCodeDesc	String
vatRate	BigDecimal

JSON Structure:

```
{
  "vatRegion": null,
  "vatRegionName": null,
  "vatRegionType": null,
  "vatType": null,
  "vatCode": null,
  "vatCodeDesc": null,
  "vatRate": null,
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_DEPS	Yes	No	No	No
VAT_DEPS	Yes	No	No	No
VAT_REGION	Yes	No	No	No
V_VAT_REGION_TL	Yes	No	No	No
V_VAT_CODES_TL	Yes	No	No	No
VAT_CODE_RATES	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Departments

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-921 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-922 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-923 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-924 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-924 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-925 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-925 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-926 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-926 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-927 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-927 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-928 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-929 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-930 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-930 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-930 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-931 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-931 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-932 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,
    }
  ]
}
```

```

"budgetedIntakePercent": 3,
"deptVatIncludeInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"inventoryLayerIdentifierType": null,
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-933 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-934 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-934 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-934 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-935 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-936 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```

{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}

```

Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
    }
  ]
}
```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```



```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
],
"hasMore": false,
"limit": 1,
"count": 1,
"links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Department Upload Service

The following services are included in this functional area:

- [Create Department](#)
- [Create Department Taxes](#)
- [Create Department Up Charge](#)
- [Delete Department](#)
- [Delete Department Up Charge](#)
- [Update Department](#)
- [Update Department Taxes](#)
- [Update Department Up Charge](#)

Create Department

Functional Area

Merchandise Hierarchy

Business Overview

The merchandise hierarchy allows the retailer to create the relationships that are necessary to support the product management structure of a company. This hierarchy reflects a classification of merchandise into multi-level descriptive categorizations to facilitate the planning, tracking, reporting, and management of merchandise within the company. This service may be used to create the department element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information,

When a new department is created, this service will first validate that all required fields are present in the request payload. Business level validation on the input information will be performed to:

- Verify if the department is not already present.
- Verify if total market amount is received. If received then it should be at least 1000.
- Verify the child details, if included, have their required fields present. The child detail contains the VAT and up charge details for a department.

If all the validations are completed successfully, then the department data is created. Active custom flex attributes that are associated with a department can also be created through this service.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/create

Input Payload Details

Table 5-937 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department. This value cannot be modified.
deptName	Yes	String (120)	The name of the department.
buyer	No	Number (4)	The number of the buyer associated to the department. This value must be predefined on the BUYER table.
purchaseType	Yes	Number (1)	The code which indicates whether items in this department are normal merchandise (0) or consignment stock (1).

Table 5-937 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
totalMarketAmount	No	Number (24,4)	The total market amount that is expected for the department. This value cannot be less than 1000.
merchandiser	No	Number (4)	The number of the merchandiser that is associated to the department. This value must be predefined on the MERCHANT table.
groupNo	Yes	Number (4)	The number of the group to which the department belongs. This value must be predefined on the GROUPS table.
budgetedMarkup	No	Number (12,4)	The budgeted markup percentage. The markup percent of cost. If this value is not populated on the message it will be calculated to be the inverse of the budgeted intake percentage. This column will hold 70% as 70, not. 70.
profitCalcType	Yes	Number (1)	The number which indicates whether profit will be calculated by direct cost (1) or retail inventory (2).
markupCalcType	Yes	String (2)	The code letter that indicates how markup is calculated in this department. Valid values are cost (C) and retail (R).
otbCalcType	No	String (1)	The code letter which indicates how OTB is calculated for this department. Valid values are cost (C) and retail (R).
maximumAverageCounter	No	Number (5)	The maximum count of days with acceptable data to include in an average for items within the department.
averageTolerancePercent	No	Number (12,4)	The tolerance percentage value used in averaging for items within this value. This column will hold 70% as 70, not. 70.
budgetedIntakePercent	No	Number (12,4)	The budgeted intake percentage. The percent of the total take that is income. If this field is not populated on the message it will be calculated as the inverse of the budgeted markup percentage. This column will hold 70% as 70, not. 70.
vat	No	Collection of Object	References a collection of department VAT information.

Table 5-937 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
upcharge	No	Collection of Object	References a collection of department upcharges.
invLayerIdentifierType	No	String (6)	This field holds the type of identifier that will be used across systems to manage and identify inventory layers at department level.
applyInvIdentifierType	No	String (1)	This field determines whether the values at the class level should be overwritten with department-level values.

Table 5-938 Vat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatRegion	Yes	Number (4)	The number of the VAT region to which this department is associated. This value must be predefined on the VAT_REGION table.
vatCode	Yes	String (6)	The alphanumeric identifier of the VAT code. This value must be predefined on the VAT_CODES table.
vatType	Yes	String (1)	Indicates if the VAT rate is used for purchasing (C), selling (R), or both (B).
reverseVatInd	No	String (1)	Indicates if items in the department are subject to reverse charge VAT at the vat region. Valid values are Y and N.

Table 5-939 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-940 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses
toLocationType	Yes	String (6)	Contains the type of destination location to which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
details	No	Collection of Object	References a collection of department upcharge detail information.

Table 5-941 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains the unique identifier of the Up Charge component.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the cost of the Item/To Location combinations within the department.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies.

Table 5-941 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified.
upChargeGroup	Yes	String (6)	This field contains the up charge group to which the component ID belongs. Valid values can be found on the codes table with a code type of UCHG. Examples are Admin. Fee (A), Freight (F) and Special K Fees (K).
componentCurrency	Yes	String (3)	This field contains the currency of the Up Charge component.
effectiveDate	No	date	The date from which the new values are effective in the system.
itemDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing items under the department.
transferAllocationDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing transfers and allocations under the department.
computationValueBase	No	String (10)	Contains the Computation Value Basis for the Up Charge component.
costBasis	No	String (6)	Contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
includeInTotalUpChargeInd	No	String (1)	Used to determine if the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level.

Sample Input Message

```
{
  "dept": 3041,
  "deptName": "deptName",
  "buyer": 610,
  "purchaseType": 1,
  "totalMarketAmount": 1010,
  "merchandiser": 810,
  "groupNo": 300,
  "budgetedMarkup": 3,
  "profitCalcType": 1,
  "markupCalcType": "C",
  "otbCalcType": "C",
```

```

"maximumAverageCounter": 42,
"averageTolerancePercent": 3,
"budgetedIntakePercent": 3,
"vat": [
  {
    "vatRegion": 1000,
    "vatCode": "S",
    "vatType": "B",
    "reverseVatInd": "Y"
  }
],
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"upcharge": [
  {
    "fromLocation": "1231",
    "toLocation": "3111",
    "fromLocationType": "S",
    "toLocationType": "S",
    "details": [
      {
        "component": "UC1",
        "componentRate": 3.55,
        "perCount": 3,
        "perCountUom": "EA",
        "upChargeGroup": "A",
        "componentCurrency": "USD",
        "effectiveDate": "2001-12-31",
        "itemDefaultInd": "Y",
        "transferAllocationDefaultInd": "Y",
        "computationValueBase": "C1",
        "costBasis": "S",
        "includeInTotalUpChargeInd": "N"
      }
    ]
  }
],
"invLayerIdentifierType": "L",
"applyInvIdentifierType": "N"
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```


Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Department Taxes

Functional Area

Merchandise Hierarchy

Business Overview

If you are configured to run Merchandising using the Simple VAT (SVAT) setup for your default tax type, then you can set the VAT rates by region for the department using this service. VAT details can be passed in the Department create service payload, or they can be passed in specifically using this service. VAT region and VAT codes records must exist prior to creating department VAT records. Also, when passing in a new VAT region to an existing department with attached items, the VAT information will default to all items.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/tax/create

Input Payload Details

Table 5-942 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department. This value cannot be modified.

Table 5-942 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vat	No	Collection of Object	References a collection of department VAT information.
invLayerIdentifierType	No	String (6)	This field holds type of identifier that will be used across systems to manage and identify inventory layers at department level.
applyInvIdentifierType	No	String (1)	This field will determine if the values at the class level should be overwritten with department level values.

Table 5-943 Vat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatRegion	Yes	Number (4)	The number of the VAT region to which this department is associated. This value must be predefined on the VAT_REGION table.
vatCode	Yes	String (6)	The alphanumeric identifier of the VAT code. This value must be predefined on the VAT_CODES table.
vatType	Yes	String (1)	Indicates if the VAT rate is used for purchasing (C), selling (R), or both (B).
reverseVatInd	No	String (1)	Indicates if items in the department are subject to reverse charge VAT at the vat region. Valid values are Y and N.

Sample Input Message

```
{
  "dept": 3041,
  "vat": [
    {
      "vatRegion": 1000,
      "vatCode": "S",
      "vatType": "C",
      "reverseVatInd": "Y"
    }
  ],
  "invLayerIdentifierType": "L",
  "applyInvIdentifierType": "N"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Department Up Charge

Functional Area

Merchandise Hierarchy

Business Overview

This service is used to create up charges for a department.

This service first validates the set of required fields, including the from-locations and to-locations in the request payload. If no up-charge record is found, this message creates the up-charge for the department and from/to location combination. As part of the addition, you can indicate in the payload if you want to have the up-charges added to existing items or only added for new items. Similarly, there is a flag in the message to indicate whether the new up charges need to be cascaded to transfers and allocations which are unshipped and not in closed or deleted status. The department up charges will be created as soon as the request payload is consumed, but the new up charges will be cascaded to items, transfers, and allocations via batches which runs at the end of every business day.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/upCharge/create

Input Payload Details

Table 5-944 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department. This value cannot be modified.
upcharge	No	Collection of Object	References a collection of department upcharges.
invLayerIdentifierType	No	String (6)	This field holds type of identifier that will be used across systems to manage and identify inventory layers at department level.
applyInvIdentifierType	No	String (1)	This field will determine if the values at the class level should be overwritten with department level values.

Table 5-945 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.

Table 5-945 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses
toLocationType	Yes	String (6)	Contains the type of destination location to which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
details	No	Collection of Object	References a collection of department upcharge detail information.

Table 5-946 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains the unique identifier of the Up Charge component.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the cost of the Item/To Location combinations within the department.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies.
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified.
upChargeGroup	Yes	String (6)	This field contains the up charge group to which the component ID belongs. Valid values can be found on the codes table with a code type of UCHG. Examples are Admin. Fee (A), Freight (F) and Special K Fees (K).
componentCurrency	Yes	String (3)	This field contains the currency of the Up Charge component.
effectiveDate	No	date	The date from which the new values are effective in the system.
itemDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing items under the department.
transferAllocationDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing transfers and allocations under the department.
computationValueBase	No	String (10)	Contains the Computation Value Basis for the Up Charge component.

Table 5-946 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
costBasis	No	String (6)	Contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
includeInTotalUpChargeInd	No	String (1)	Used to determine if the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level.

Sample Input Message

```

{
  "dept": 3041,
  "upcharge": [
    {
      "fromLocation": "1231",
      "toLocation": "6000",
      "fromLocationType": "S",
      "toLocationType": "S",
      "details": [
        {
          "component": "UC1",
          "componentRate": 6.2,
          "perCount": 3,
          "perCountUom": "EA",
          "upChargeGroup": "A",
          "componentCurrency": "USD",
          "effectiveDate": "2001-12-31",
          "itemDefaultInd": "N",
          "transferAllocationDefaultInd": "Y",
          "computationValueBase": "C1",
          "costBasis": "S",
          "includeInTotalUpChargeInd": "N"
        }
      ]
    }
  ]
}

```

```
},  
"invLayerIdentifierType": "L",  
"applyInvIdentifierType": "Y"  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Delete Department

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to delete the department element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a department is deleted, this service will add the department to a purge staging table for processing in the Daily Purge of Foundation Data process.

Note: Associated custom flex attributes are also deleted when the department is deleted.

For more details on Departments, see the Create Department service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/delete

Input Payload Details

Table 5-947 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department.

Sample Input Message

```
{
  "dept": 3041
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Department Up Charge

Functional Area

Merchandise Hierarchy

Business Overview

This service is used to delete up charges linked with a department. When a request containing an up charge is received, the set of required fields is validated. If the up charge record exists for the department, then this gets deleted. If no up charge component is provided, the system will delete all the up charge components along with the from location / to location combination. Deleting up charges from a department does not automatically remove them from the items or transfers and allocations for items in the department.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/upCharge/delete

Input Payload Details

Table 5-948 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department.
upcharge	No	Collection of Object	References a collection of department upcharge information.

Table 5-949 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.

Table 5-949 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
toLocationType	Yes	String (6)	Contains the type of destination location to which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
details	No	Collection of Object	References a collection of department upcharge detail information.

Table 5-950 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	No	String (10)	This field contains the unique identifier of the Up Charge component.
itemDefaultInd	No	String (1)	Indicates if component rate information is deleted or not for existing items under the department.

Sample Input Message

```
{
  "dept": 3041,
  "upcharge": [
    {
      "fromLocation": "1121",
      "toLocation": "1311",
      "fromLocationType": "S",
      "toLocationType": "S",
      "details": [
        {
          "component": "UC1",
          "itemDefaultInd": "N"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
```

```
{
  "error": "must be one of Y, N",
  "field": "createRecord.arg0.approveInd",
  "inputValue": "X"
}
]
```

Update Department

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to update the department element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a department is updated, this service will first validate that all the required fields are present in the message. Business level validation on the input information will be performed to:

- Verify if the department is present.
- Verify if total market amount is received then it should be at least 1000.
- Verify the child detail, if included, contain all required fields. The child detail contain VAT and upcharge details for a department.

If all the validations are completed successfully, then the department is updated. Active custom flex attributes can also be updated for the department via this service.

For more details on Department, see the Create Department service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/update

Input Payload Details

Table 5-951 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department. This value cannot be modified.
deptName	Yes	String (120)	The name of the department.
buyer	No	Number (4)	The number of the buyer associated to the department. This value must be predefined on the BUYER table.

Table 5-951 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
totalMarketAmount	No	Number (24,4)	The total market amount that is expected for the department. This value cannot be less than 1000.
merchandiser	No	Number (4)	The number of the merchandiser that is associated to the department. This value must be predefined on the MERCHANT table.
groupNo	Yes	Number (4)	The number of the group to which the department belongs. This value must be predefined on the GROUPS table.
budgetedMarkup	No	Number (12,4)	The budgeted markup percentage. The markup percent of cost. If this value is not populated on the message it will be calculated to be the inverse of the budgeted intake percentage. This column will hold 70% as 70, not. 70.
markupCalcType	Yes	String (2)	The code letter that indicates how markup is calculated in this department. Valid values are cost (C) and retail (R).
maximumAverageCounter	No	Number (5)	The maximum count of days with acceptable data to include in an average for items within the department.
averageTolerancePercent	No	Number (12,4)	The tolerance percentage value used in averaging for items within this value. This column will hold 70% as 70, not. 70.
budgetedIntakePercent	No	Number (12,4)	The budgeted intake percentage. The percent of the total take that is income. If this field is not populated on the message it will be calculated as the inverse of the budgeted markup percentage. This column will hold 70% as 70, not. 70.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
invLayerIdentifierType	No	String (6)	This field holds the type of identifier that will be used across systems to manage and identify inventory layers at department level.

Table 5-951 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
applyInvIdentifierType	No	String (1)	This field determine whether the values at the class level should be overwritten with department-level values.

Table 5-952 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "dept": 3041,
  "deptName": "deptName",
  "buyer": 1000,
  "totalMarketAmount": 3,
  "merchandise": 1000,
  "groupNo": 1000,
  "budgetedMarkup": 3,
  "markupCalcType": "C",
  "maximumAverageCounter": 10000,
  "averageTolerancePercent": 3,
  "budgetedIntakePercent": 3,
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": "2001-12-31"
    }
  ],
  "invLayerIdentifierType": "L",
  "applyInvIdentifierType": "N"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Department Taxes

Functional Area

Merchandise Hierarchy

Business Overview

If you are configured to run Merchandising using Simple VAT (SVAT) for your default tax type, then you can update the VAT rates by region for the department using this service. This service will check for all the required fields in the message and updates the VAT information for a department. When updating VAT details for a department containing items, the VAT information will default to all items within the department.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/department/tax/update

Input Payload Details

Table 5-953 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department. This value cannot be modified.
vat	No	Collection of Object	References a collection of department VAT information.

Table 5-953 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
invLayerIdentifierType	No	String (6)	This field holds the type of identifier that will be used across systems to manage and identify inventory layers at department level.
applyInvIdentifierType	No	String (1)	This field determines whether the values at the class level should be overwritten with department-level values.

Table 5-954 Vat - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vatRegion	Yes	Number (4)	The number of the VAT region to which this department is associated. This value must be predefined on the VAT_REGION table.
vatCode	Yes	String (6)	The alphanumeric identifier of the VAT code. This value must be predefined on the VAT_CODES table.
vatType	Yes	String (1)	Indicates if the VAT rate is used for purchasing (C), selling (R), or both (B).
reverseVatInd	No	String (1)	Indicates if items in the department are subject to reverse charge VAT at the vat region. Valid values are Y and N.

Sample Input Message

```
{
  "dept": 3041,
  "vat": [
    {
      "vatRegion": 1000,
      "vatCode": "S",
      "vatType": "C",
      "reverseVatInd": "Y"
    }
  ],
  "invLayerIdentifierType": "L",
  "applyInvIdentifierType": "N"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Department Up Charge

Functional Area

Merchandise Hierarchy

Business Overview

This service is used to update up charges linked with a department. When a message containing up charge details is received, the set of required fields is validated. If an up charge record exists for a department and the from-location and to-location combination in the request payload, then the up charge details are updated for the department. As part of the update there is also an option to have the up charges updated for items in the department, or unshipped transfers and allocations for items in the department. The department up charges will be updated as soon as the request payload is consumed, but the updates will be cascaded to items, transfers, and allocations via batches which runs at the end of every day.

Service Type

PUT

ReST URL

`MerchIntegrations/services/merchandiseHierarchy/department/upCharge/update`

Input Payload Details

Table 5-955 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	Yes	Number (4)	The number which uniquely identifies the department. This value cannot be modified.
upcharge	No	Collection of Object	References a collection of department upcharges.
invLayerIdentifierType	No	String (6)	This field holds type of identifier that will be used across systems to manage and identify inventory layers at department level.
applyInvIdentifierType	No	String (1)	This field will determine if the values at the class level should be overwritten with department level values.

Table 5-956 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.

Table 5-956 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses
toLocationType	Yes	String (6)	Contains the type of destination location to which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
details	No	Collection of Object	References a collection of department upcharge detail information.

Table 5-957 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains the unique identifier of the Up Charge component.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the cost of the Item/To Location combinations within the department.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies.
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified.
upChargeGroup	Yes	String (6)	This field contains the up charge group to which the component ID belongs. Valid values can be found on the codes table with a code type of UCHG. Examples are Admin. Fee (A), Freight (F) and Special K Fees (K).
componentCurrency	Yes	String (3)	This field contains the currency of the Up Charge component.
effectiveDate	No	date	The date from which the new values are effective in the system.
itemDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing items under the department.
transferAllocationDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing transfers and allocations under the department.
computationValueBase	No	String (10)	Contains the Computation Value Basis for the Up Charge component.

Table 5-957 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
costBasis	No	String (6)	Contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
includeInTotalUpChargeInd	No	String (1)	Used to determine if the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level.

Sample Input Message

```

{
  "dept": 3041,
  "upcharge": [
    {
      "fromLocation": "1231",
      "toLocation": "1521",
      "fromLocationType": "S",
      "toLocationType": "S",
      "details": [
        {
          "component": "UC1",
          "componentRate": 3.55,
          "perCount": 3,
          "perCountUom": "EA",
          "upChargeGroup": "A",
          "componentCurrency": "USD",
          "effectiveDate": "2001-12-31",
          "itemDefaultInd": "N",
          "transferAllocationDefaultInd": "Y",
          "computationValueBase": null,
          "costBasis": "S",
          "includeInTotalUpChargeInd": "N"
        }
      ]
    }
  ]
}

```

```
},
"invLayerIdentifierType": "L",
"applyInvIdentifierType": "N"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Division Download Service

The following services are included in this functional area:

- [Get Division Details for Provided Division](#)
- [Get Divisions](#)

Get Division Details for Provided Division

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-958 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-959 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-960 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-961 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-961 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-962 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-962 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-963 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-963 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-964 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-964 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-965 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-966 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-967 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-967 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-967 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-968 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-968 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-969 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,
    }
  ]
}
```

```

"budgetedIntakePercent": 3,
"deptVatIncludeInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"inventoryLayerIdentifierType": null,
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-970 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-971 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-971 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-971 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-972 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-973 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```

{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}

```

Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
],
"hasMore": false,
"limit": 1,
"count": 1,
"links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Get Divisions

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-974 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-975 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-976 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-977 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-977 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-978 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-978 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-979 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-979 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-980 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-980 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-981 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-982 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-983 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-983 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-983 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-984 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-984 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-985 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,
    }
  ]
}
```

```

"budgetedIntakePercent": 3,
"deptVatIncludeInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"inventoryLayerIdentifierType": null,
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-986 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-987 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-987 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-987 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-988 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-989 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
    }
  ]
}
```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Division Upload Service

The following services are included in this functional area:

- [Create Division](#)
- [Delete Division](#)
- [Update Division](#)

Create Division

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to create the division element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a new division is created, this service will first validate that all the required

fields are present in the request payload. Business level validation on the input information will be performed to:

- Verify division is not already present
- Verify if the total market amount is received, then it should be at least 1000.

If both the above validations are met, the division in the message data is created.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/division/create

Input Payload Details

Table 5-990 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
division	Yes	Number (4)	The unique identifier of the division.
divisionName	Yes	String (120)	The name of the division.
merchandiser	No	Number (4)	The number of the merchant associated with the division. This value must be predefined on the MERCHANT table.
buyer	No	Number (4)	The number of the buyer associated with the division. This value must be predefined on the BUYER table.
totalMarketAmount	No	Number (24,4)	The total market amount that is expected for the division. If this field is not null it must be at least 1000.

Sample Input Message

```
{
  "division": 200,
  "divisionName": "Division 200",
  "merchandiser": 612,
  "buyer": 307,
  "totalMarketAmount": 1010
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```


Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Division

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to delete the division element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a division is deleted, this service will first validate that all required fields are present in the message. Business level validation on the input information will be performed to:

- Verify the division already exists.
- Verify the division is not associated to a differentiator group.
- Verify the division is not associated to a season ID.
- Verify the division is not associated to a ticket type.
- Verify the division is not associated to a UDA.

If the information passes the above validations, the division will be deleted in Merchandising. Deletion of a division will get processed immediately upon receipt of the request, assuming that there are no dependent levels below it.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/division/delete

Input Payload Details

Table 5-991 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
division	Yes	Number (4)	The unique identifier of the division.

Sample Input Message

```
{
  "division": 1000
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Division

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to update the division element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy

information. When a division is updated, this service will first validate that all required fields are present in the request payload. Business level validation on the input information will be performed to:

- Verify division is present.
- Verify if the total market amount is received, then it should be at least 1000.

If all the validations are met, the details of the division are updated.

For more details on Division, see the Create Division service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/division/update

Input Payload Details

Table 5-992 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
division	Yes	Number (4)	The unique identifier of the division.
divisionName	Yes	String (120)	The name of the division.
merchandiser	No	Number (4)	The number of the merchant associated with the division. This value must be predefined on the MERCHANT table.
buyer	No	Number (4)	The number of the buyer associated with the division. This value must be predefined on the BUYER table.
totalMarketAmount	No	Number (24,4)	The total market amount that is expected for the division. If this field is not null it must be at least 1000.

Sample Input Message

```
{
  "division": 1000,
  "divisionName": "divName",
  "merchandiser": 1000,
  "buyer": 1000,
  "totalMarketAmount": 3
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Group Download Service

The following services are included in this functional area:

- [Get Group for Provided Group](#)
- [Get Groups](#)

Get Group for Provided Group

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table

MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:

MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-993 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-994 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-995 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-996 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-996 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-997 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-997 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-998 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-998 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-999 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-999 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1000 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-1001 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1002 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-1002 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-1002 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1003 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-1003 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1004 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,

```

```

"budgetedIntakePercent": 3,
"deptVatIncludeInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"inventoryLayerIdentifierType": null,
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-1005 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1006 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-1006 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-1006 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1007 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1008 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```


Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
    }
  ]
}
```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Get Groups

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-1009 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1010 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-1011 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1012 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-1012 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1013 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-1013 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-1014 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-1014 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1015 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-1015 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1016 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-1017 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1018 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-1018 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-1018 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1019 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-1019 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1020 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,
    }
  ]
}
```

```

"budgetedIntakePercent": 3,
"deptVatIncludeInd": "Y",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"inventoryLayerIdentifierType": null,
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-1021 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1022 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-1022 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-1022 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1023 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1024 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```


Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
    }
  ]
}
```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Group Upload Service

The following services are included in this functional area:

- [Create Group](#)
- [Delete Group](#)
- [Update Group](#)

Create Group

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to create the group element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information.

When a new group is created, this service will first validate that all the required fields are present in the request payload. Business level validation on the input information will be performed, which checks whether the group already exists. If it does not exist, the group is created.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/group/create

Input Payload Details

Table 5-1025 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
groupNo	Yes	Number (4)	The number which uniquely identifies the group. This field cannot be modified.
groupName	Yes	String (120)	Description is not available.
division	Yes	Number (4)	The identifier of the division of which the group is a member. This value must be predefined on the DIVISION table.
merchandiser	No	Number (4)	The number of the merchandiser associated to the group. This value must be predefined on the MERCHANT table.
buyer	No	Number (4)	The number of the buyer associated to the group. This value must be predefined on the BUYER table.

Sample Input Message

```
{
  "groupNo": 1000,
  "groupName": "groupName",
  "division": 1000,
  "merchandiser": 1000,
  "buyer": 1000
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Group

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to delete the group element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy information. When a group is deleted, this service will first validate that all required fields are present in the request payload. Business level validation on the input information will be performed to:

- Verify the group already exists.
- Verify the group is not associated to a differentiator group.
- Verify the group is not associated to a season ID
- Verify the group is not associated to a ticket type.
- Verify the group is not associated to a UDA.

If the information passes the above validations, the group will be deleted from Merchandising. Deletion of a group will get processed immediately upon receipt of the request, assuming that there are no dependent levels below it.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/group/delete

Input Payload Details

Table 5-1026 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
groupNo	Yes	Number (4)	Description is not available.

Sample Input Message

```
{
  "groupNo": 1000
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Group

Functional Area

Merchandise Hierarchy

Business Overview

This service may be used to update the group element of the merchandise hierarchy based on an external system if Merchandising is not the system of record for merchandise hierarchy

information. When a group is updated, this service will first validate that all required fields are present in the request payload. Business level validation on the input information will be performed to verify whether the group to be updated already exists. If group already exists, the details of the group are updated.

For more details on Group, see the Create Group service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/group/update

Input Payload Details

Table 5-1027 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
groupNo	Yes	Number (4)	The number which uniquely identifies the group. This field cannot be modified.
groupName	Yes	String (120)	Description is not available.
division	Yes	Number (4)	The identifier of the division of which the group is a member. This value must be predefined on the DIVISION table.
merchandiser	No	Number (4)	The number of the merchandiser associated to the group. This value must be predefined on the MERCHANT table.
buyer	No	Number (4)	The number of the buyer associated to the group. This value must be predefined on the BUYER table.

Sample Input Message

```
{
  "groupNo": 1000,
  "groupName": "groupName",
  "division": 1000,
  "merchandiser": 1000,
  "buyer": 1000
}
```


Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Merchandise Hierarchy Download Service

The following services are included in this functional area:

- [Get Merchandise Hierarchy](#)

Get Merchandise Hierarchy

This section describes the `MerchHierarchyDetail` service.

Business Overview

`MerchHierarchyDetail` service allows user to retrieve full merchandising hierarchy information for the input Department.

Service Type

Get

ReST URL

`/MerchHierarchy/merchHierarchy`

Input Parameters

Department ID

Output

RestMerchHierarchyRecRDO

Parameter Name	Data Type
profitCalcType	BigDecimal
deptVatInclInd	String
classAttribute	BigDecimal
division	BigDecimal
classVatInd	String
subclass	BigDecimal
buyer	BigDecimal
dept	BigDecimal
className	String
subName	String
groupNo	BigDecimal
otbCalcType	String
groupName	String
divName	String
purchaseType	BigDecimal
merch	BigDecimal
deptName	String

JSON Structure

```
{
  "profitCalcType": null,
  "deptVatInclInd": null,
  "classAttribute": null,
  "division": null,
  "classVatInd": null,
  "subclass": null,
  "buyer": null,
  "dept": null,
  "className": null,
  "subName": null,
  "groupNo": null,
  "otbCalcType": null,
  "groupName": null,
  "divName": null,
  "purchaseType": null,
  "merch": null,
  "deptName": null,
  "links": [],
  "hyperMediaContent": {
```

```
    "linkRDO": []  
  }  
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_MERCH_HIERARCHY	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Subclass Download Service

The following services are included in this functional area:

- [Get Subclass Details for Provided Subclass](#)
- [Get Subclasses](#)

Get Subclass Details for Provided Subclass

Endpoints

MerchIntegrations/services/foundation/merchhier/division/{division}

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/groups/{group}

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/deps/{dept}

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

MerchIntegrations/services/foundation/merchhier/class

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

MerchIntegrations/services/foundation/merchhier/subclass

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-1028 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1029 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffzh:tzm
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/
{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/
{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-1030 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch

Table 5-1030 (Cont.) PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1031 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.

Table 5-1031 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1032 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```

{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ]
}

```

```

],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Groups

Table 5-1033 PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1034 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.

Table 5-1034 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1035 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-1035 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Department**Table 5-1036 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-1036 (Cont.) PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1037 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.

Table 5-1037 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercentage	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.

Table 5-1037 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1038 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1039 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,
      "budgetedIntakePercent": 3,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "inventoryLayerIdentifierType": null,
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
```

```

{
  "href": "https://<server>:<port>/Application/services/resources?
  limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
  "rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-1040 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-1040 (Cont.) PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1041 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.

Table 5-1041 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.

Table 5-1041 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1042 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1043 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-1043 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    },
    {
      "hasMore": true,
      "limit": 1000,
      "count": 1000,
      "links": [
        {
          "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
          "rel": "next"
        }
      ]
    }
  ]
}
```

Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",
    }
  ]
}
```



```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
],
"hasMore": false,
"limit": 1,
"count": 1,
"links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Get Subclasses

Endpoints

`MerchIntegrations/services/foundation/merchhier/division/{division}`

`MerchIntegrations/services/foundation/merchhier/division`

`MerchIntegrations/services/foundation/merchhier/groups/{group}`

`MerchIntegrations/services/foundation/merchhier/groups`

`MerchIntegrations/services/foundation/merchhier/deps/{dept}`

`MerchIntegrations/services/foundation/merchhier/deps`

`MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}`

`MerchIntegrations/services/foundation/merchhier/class`

`MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}`

`MerchIntegrations/services/foundation/merchhier/subclass`

Functional Area

Merchandise Hierarchy

Business Overview

This service publishes merchandise hierarchy department data to downstream consuming systems. It is the department-level projection of the shared merchandise hierarchy cache and publishes the current department definition together with department name, buyer, purchase type, group, budgeting and markup attributes, inventory-layer identifier type, and department custom flex attributes.

The merch hierarchy publication for the different hierarchy level (Division, Groups, Department, Class and Subclass) shares the same underlying processing logic and each of the level have its own endpoint. The below technical details covers all the hierarchy levels.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on

how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: DIVISION, GROUPS, DEPS, CLASS, SUBCLASS

Additional business tables: COMPHEAD, DEPS_CFA_EXT, CLASS_CFA_EXT, SUBCLASS_CFA_EXT

JSON cache table: MERCHAPI_EXT_MERCHHIER

JSON generation view: not applicable; this service reads from the shared cache maintained by foundation/merchhier

This service is the department-level child merchapi of the shared foundation/merchhier family.

It publishes only department rows from the shared cache table MERCHAPI_EXT_MERCHHIER, so the source tables, delta drivers, refresh logic, CFA helper handling, batch job, and background-engine controls are the same as the parent merchapi family section above.

Functionally, it is the department-only projection of the current merchandise hierarchy.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_MERCHHIER_JSON and apply no status filter; publishes current hierarchy rows that exist in the source tables.

Rebuild marks cache rows deleted when the hierarchy key no longer exists in V_MERCHAPI_EXT_MERCHHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_DIVISION, ICL_RMS_GROUPS, ICL_RMS_DEPS, ICL_RMS_CLASS, ICL_RMS_SUBCLASS (ICL consumer name = MERCHAPI)

This service relies on the shared delta-processing flow used by foundation/merchhier.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_MERCHHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_MERCH_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_MERCH_HIER_ADHOC_JOB

Webhook configuration api name:

Division: foundation/merchhier/division

Groups: foundation/merchhier/groups

Department: foundation/merchhier/deps

Class: foundation/merchhier/class

Subclass: foundation/merchhier/subclass

Special handling

The parent merchapi configuration name for all the hierarchy level is 'foundation/merchhier'.

Table 5-1044 Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (10)	Offset key (Hierarchy ID)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1045 Input Parameter for "MerchIntegrations/services/foundation/merchhier/deps/{dept}"

Parameter Name	Required	Data Type	Description
dept	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URLs for Division

MerchIntegrations/services/foundation/merchhier/division

MerchIntegrations/services/foundation/merchhier/division/{division}

ReST URLs for Groups

MerchIntegrations/services/foundation/merchhier/groups

MerchIntegrations/services/foundation/merchhier/groups/{group}

ReST URLs for Department

MerchIntegrations/services/foundation/merchhier/deps

MerchIntegrations/services/foundation/merchhier/deps/{dept}

ReST URLs for Class

/MerchIntegrations/services/foundation/merchhier/class?
since={since}&before={before}&offsetkey={offsetkey}&limit={limit}&dept={dept}&
class={class}

/MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

ReST URLs for Subclass

MerchIntegrations/services/foundation/merchhier/subclass

MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Common Input parameters for URLs without path parameter

Parameter Name	Required	Data Type	Description
since	No	String	Since Date. Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
before	No	String	Before Date Format: yyyy-mm-dd"T"hh24:mi:ss.fffz:tz
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/class

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return classes which belong to this dept

Parameter Name	Required	Data Type	Description
class	No	Number	If class is provided, the service will return class(es) matching the input class number. To query a specific class, value for both dept and class should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input query parameter for /MerchIntegrations/services/foundation/merchhier/subclass

Parameter Name	Required	Data Type	Description
dept	No	Number	If dept is provided, the service will return subclasses which belong to this dept
class	No	Number	If class is provided, the service will return subclass(es) whose parent class number matches the input class number.
subclass	No	Number	If subclass is provided, the service will return subclass(es) matching the input subclass number. To query a specific subclass, value for dept and class and subclass should be provided.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/division/{division}

Parameter Name	Required	Data Type	Description
division	Yes	Number	This field contains the number which uniquely identifies the division of the company
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/groups/{group}

Parameter Name	Required	Data Type	Description
group	Yes	Number	This field contains the number which uniquely identifies the group.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/deps/{dept}

Parameter Name	Required	Data Type	Description
dept	Yes	Number	This field contains the number which uniquely identifies the department.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/class/{uniqueClassId}

Parameter Name	Required	Data Type	Description
uniqueClassId	Yes	Number	This field contains the number that uniquely identifies the department/class combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input path parameter for /MerchIntegrations/services/foundation/merchhier/subclass/{uniqueSubclassId}

Parameter Name	Required	Data Type	Description
uniqueSubclassId	Yes	Number	This field contains the number that uniquely identifies the department/class/subclass combination.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Division

Table 5-1046 PageResultsItemsMerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1047 MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for division details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.
divisionName	No	String (120)	This field contains the name which, along with the division number, identifies the division of the company.

Table 5-1047 (Cont.) MerchApiDivision - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the division.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the division.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the division. The field can be used for analytics or reporting.
company	No	Number(4,0)	This field contains the number which uniquely identifies the company.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1048 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-1048 (Cont.) LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "division": 1,
      "divisionName": "divName",
      "buyer": 1,
      "merchantId": 1,
      "totalMarketAmount": 3,
      "company": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output - Groups**Table 5-1049 PageResultsItemsMerchApiGroup - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-1049 (Cont.) PageResultsItemsMerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1050 MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for group details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
group	No	Number(4,0)	This field contains the number which uniquely identifies the group.
groupName	No	String (120)	This field contains the description which, along with the group number, identifies the group.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the group.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the group.
division	No	Number(4,0)	This field contains the number which uniquely identifies the division of the company.

Table 5-1050 (Cont.) MerchApiGroup - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1051 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "group": 1,
      "groupName": "groupName",
      "buyer": 1,
      "merchantId": 1,
      "division": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",

```

```

"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Department

Table 5-1052 PageResultsItemsMerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1053 MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for department details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
dept	No	Number(4,0)	This field contains the number which uniquely identifies the department.

Table 5-1053 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptName	No	String (120)	This field contains the description which identifies the department.
buyer	No	Number(4,0)	This field contains the number of the buyer associated with the department.
purchaseType	No	Number(1,0)	This field contains a code which indicates whether items in this department are, by default, owned (normal) merchandise, consignment stock, or concession stock.
totalMarketAmount	No	Number(24,4)	This field contains the total market amount that is expected for the department. The field can be used for analytics or reporting.
merchantId	No	Number(4,0)	This field contains the number of the merchandiser associated with the department.
group	No	Number(4,0)	This field contains the number of the group to which the department belongs.
budgetedMarkup	No	Number(12,4)	This field contains the budgeted markup percentage.
profitCalcType	No	Number(1,0)	This field contains a number which indicates whether profit will be calculated by cost or retail, indicating the accounting method to be used for the department.
markupCalcType	No	String (2)	This field contains the code which determines how markup is calculated in this department.

Table 5-1053 (Cont.) MerchApiDepartment - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbCalcType	No	String (1)	This field contains the code which determines how OTB is calculated in this department.
budgetedIntakePercent	No	Number(12,4)	This field contains the budgeted intake percentage.
deptVatIncludeInd	No	String (1)	This field indicates the default value for the class VAT indicator. When classes are initially set up they inherit this value.
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the department.
inventoryLayerIdentifierType	No	String (6)	This field holds the inventory layer identifier type at the department level.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1054 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.

Table 5-1054 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1055 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "dept": 3041,
      "deptName": "deptName",
      "buyer": 1,
      "purchaseType": 1,
      "totalMarketAmount": 3,
      "merchantId": 1,
      "group": 1,
      "budgetedMarkup": 3,
      "profitCalcType": 1,
      "markupCalcType": null,
      "otbCalcType": null,

```

```

    "budgetedIntakePercent": 3,
    "deptVatIncludeInd": "Y",
    "createDateTime": "2001-12-31T23:59:59.000Z",
    "updateDateTime": "2001-12-31T23:59:59.000Z",
    "customFlexAttribute": [
    {
    "name": "attribute1",
    "value": "value1",
    "valueDate": "2001-12-31",
    "createDateTime": "2001-12-31T23:59:59.000Z",
    "updateDateTime": "2001-12-31T23:59:59.000Z"
    }
    ],
    "inventoryLayerIdentifierType": null,
    "cacheTimestamp": "2001-12-31T23:59:59.000Z",
    "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
  }
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Output - Class

Name	Data Type	Description
action	String	Action for class details. Valid values are INSERT, UPDATE and DELETE.
hierarchylevel	String	This field contains the Merchandising Hierarchy Level.
class	Number	Contains the number which uniquely identifies the class within the department.
className	String	This field contains the name of the class which, along with the class number, identifies the class.
classVATInd	String	This field is no longer used by Merchandising and will always default to N.
dept	Number	This field contains the number of the department of which the class is a member.
uniqueClassId	Number	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Name	Data Type	Description
createDateTime	String	This field holds the record creation date.
updateDateTime	String	This field holds the date when the record was last updated.
customFlexAttributes		This list holds custom flex attributes for the Department.
cacheTimestamp	String	This field specifies date and time when Merch Hier record was last maintained.
inventoryLayerIdentifierType	String	This field holds type of identifier that will be used across systems to manage and identify inventory layers at class level.

Output - Subclass

Table 5-1056 PageResultsItemsMerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1057 MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for subclass details.
hierarchyLevel	No	String (10)	This field contains the merchandise hierarchy level.
subclass	No	Number(4,0)	This field contains the number which uniquely identifies the subclass within the department and class.

Table 5-1057 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclassName	No	String (120)	This field contains the name of the subclass which, along with the subclass number, uniquely identifies the subclass.
class	No	Number(4,0)	This field contains the number which uniquely identifies the class within the department.
dept	No	Number(4,0)	This field contains the number of the department of which the subclass is a member.
uniqueSubclassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class/subclass combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class/subclass key in their hierarchy and item structures.
uniqueClassId	No	Number(10,0)	This field contains the number that uniquely identifies the department/class combination. This column is populated by the system and not displayed to end users. This will be used for integration to downstream systems as many downstream systems are unable to accommodate the composite dept/class key in their hierarchy and item structures.

Table 5-1057 (Cont.) MerchApiSubclass - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time.
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated.
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the subclass in merchandise hierarchy.
cacheTimestamp	No	dateTime	This field specifies the date and time when the merchandise hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1058 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1059 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "subclass": 1,
      "subclassName": "subclassName",
      "class": 1,
      "dept": 3041,
      "uniqueSubclassId": 1,
      "uniqueClassId": 1,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "customFlexAttribute": [
        {
          "name": "attribute1",
          "value": "value1",
          "valueDate": "2001-12-31",
          "createDateTime": "2001-12-31T23:59:59.000Z",
          "updateDateTime": "2001-12-31T23:59:59.000Z"
        }
      ],
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
      limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Output – Custom Flex Attribute

Name	Data Type	Description
name	String	This field will hold the custom flex attribute name.
value	String	This field will hold the numeric or string value of the custom flex attribute.
valueDate	String	This field will hold the date value of the custom flex attribute.
createDateTime	String	This field will hold the date time when the custom flex attribute was inserted.
updateDateTime	String	This field will hold the date time when the custom flex attribute was last updated.

Elements in JSON Output

Elements	Description
items	The first element is named as items and contains a number of child elements. Each child element corresponds to a row in the result set generated by the query.
hasMore	This is dependent on limit value. If the total rows are more than then limit defined then hasMore will be set to true else false.
limit	This is used to limit the rows to control the paging of result data. Based on the limit, next and previous links are displayed. For example: If the limit is set to 25, internal it tries to fetch 26 rows. If 26th row exists then 25 rows are returned along with a "next" link else no links would be added to the JSON response.
links	"self" : This contains URI pointing to current page. "next": This contains URI pointing to the next page of results.

Example JSON Output

Division

```
{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DIVISION",
      "division": 1000,
      "divisionName": "Grocery",
      "buyer": 1002,
      "merchantId": "612",
      "totalMarketAmount": null,
      "company": 1,
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T11:34:21.000Z",

```



```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Groups

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "GROUP",
      "group": 1841,
      "groupName": "Smoke Group",
      "buyer": 1003,
      "merchantId": "845",
      "division": 3978,
      "createDateTime": "2020-08-24T17:06:58.000Z",
      "updateDateTime": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Department

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "DEPARTMENT",
      "dept": 3061,
      "deptName": "Cats",
      "buyer": 610,
      "purchaseType": 0,
      "totalMarketAmount": null,
      "merchantId": "840",
      "group": 306,
      "budgetedMarkup": 33.3333,
      "profitCalcType": 1,
      "markupCalcType": "C",
      "otbCalcType": "C",
      "budgetedIntakePercent": 25,
      "deptVatIncludeInd": "Y",
      "createDateTime": "2020-08-18T08:45:53.000Z",
      "updateDateTime": "2021-07-23T12:01:34.000Z",
      "customFlexAttribute": null,
    }
  ]
}

```

```

        "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
],
"hasMore": false,
"limit": 1,
"count": 1,
"links": null
}

```

Class

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "CLASS",
      "class": 1,
      "className": "Childrens",
      "classVatInd": "Y",
      "dept": 3100,
      "uniqueClassId": 204,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": null,
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,
  "links": null
}

```

Subclass

```

{
  "items": [
    {
      "action": null,
      "hierarchyLevel": "SUBCLASS",
      "subclass": 2,
      "subclassName": "Reels",
      "class": 3,
      "dept": 3032,
      "uniqueSubclassId": 15,
      "uniqueClassId": 137,
      "createDateTime": "2020-08-18T08:45:55.000Z",
      "updateDateTime": "2021-07-23T12:12:17.000Z",
      "customFlexAttribute": null,
      "cacheTimestamp": "2021-08-19T06:55:11.469Z"
    }
  ],
  "hasMore": false,
  "limit": 1,
  "count": 1,

```

```
"links": null
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_DIVISION	Yes	No	Yes	Yes
ICL_RMS_GROUPS	Yes	No	Yes	Yes
ICL_RMS_DEPS	Yes	No	Yes	Yes
ICL_RMS_CLASS	Yes	No	Yes	Yes
ICL_RMS_SUBCLASS	Yes	No	Yes	Yes
COMPHEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
MERCHAPI_EXT_MERCHHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_MERCH_HIER_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCHAPI_EXT_CLASS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_DEPS_CFA	Yes	Yes	No	Yes
MERCHAPI_EXT_SUBCLASS_CFA	Yes	Yes	No	Yes
DEPS_CFA_EXT	Yes	No	No	No
CLASS_CFA_EXT	Yes	No	No	No
SUBCLASS_CFA_EXT	Yes	No	No	No

Subclass Upload Service

The following services are included in this functional area:

- [Create Subclass](#)
- [Delete Subclass](#)
- [Update Subclass](#)

Create Subclass

Functional Area

Merchandise Hierarchy

Business Overview

This service is can be used to create subclasses for an existing department-class in Merchandising. When a new subclass is created, this service will first validate that all required fields are present in the request payload. Business level validation on the input information will

be performed, which checks whether the subclass already exists. If the subclass does not exist, then it is created. Active custom flex attributes that are associated with a subclass can also be created through this service.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/subclass/create

Input Payload Details

Table 5-1060 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclass	Yes	Number (4)	The unique number which identifies the subclass. This value cannot be modified.
subclassName	Yes	String (120)	The name of the subclass.
class	Yes	Number (4)	The number of the class which contains the subclass. This value must be predefined on the CLASS table. This value cannot be modified.
dept	Yes	Number (4)	The number of the department which contains the subclass. This value must be predefined on the DEPS table. This value cannot be modified.
intercompanyTransferPrice Basis	No	String (6)	This specifies the Default Intercompany Transfer Price that will be used during recording of transaction data postings on intercompany transfers and allocations. Valid values are 'W'(Weighted Average Cost) and 'S'(Supplier Cost). If not defined, the value will default to 'W' for subclasses under departments with cost-based profit calculation type or to 'R' (Retail Based) for subclasses under departments with retail-based profit calculation type.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-1060 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updSellDateOnStOrdRcpt	No	String (1)	This field is used to capture whether the selling phase start date at an item/ location level would need to be updated in case of receipts against purchase orders having a store as their destination.
updSellDateOnWh2stRcpt	No	String (1)	This field is used to capture whether the selling phase start date at an item/ location level would need to be updated in case of receipts against Warehouse to Store transfers or allocation.
updSellDateOnSt2stRcpt	No	String (1)	This field is used to capture whether the selling phase start date at an item/ location level would need to be updated in case of receipts against Store to Store transfers.
minUnitsForSellDateReset	No	Number (12,4)	This field is to capture at the subclass level the minimum number of units that would trigger an update to the selling phase start date to the receipt date in case of a warehouse to store transfer/allocation or a store to store transfer.

Table 5-1061 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "subclass": 1,
  "subclassName": "subclassName",
  "class": 1,
  "dept": 3041,
  "intercompanyTransferPriceBasis": null,
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": "2001-12-31"
    }
  ],
  "updSellDateOnStOrdRcpt": "N",
  "updSellDateOnWh2stRcpt": "N",
  "updSellDateOnSt2stRcpt": "N",
}
```

```
"minUnitsForSellDateReset": 50
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Subclass

Functional Area

Merchandise Hierarchy

Business Overview

This service is used to delete an existing subclass from Merchandising. When a subclass is deleted, this service will add the subclass to a purging staging table for processing in the Daily Purge of Foundation Data process.

Note: Associated custom flex attributes are deleted when the subclass is deleted.

For more details on Subclass, see the Create Subclass service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/subclass/delete

Input Payload Details

Table 5-1062 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclass	Yes	Number (4)	The unique number which identifies the subclass.
class	Yes	Number (4)	The number of the class which contains the subclass. This value must be predefined on the CLASS table.
dept	Yes	Number (4)	The number of the department which contains the subclass. This value must be predefined on the DEPS table.

Sample Input Message

```
{
  "subclass": 1,
  "class": 1,
  "dept": 3041
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Subclass

Functional Area

Merchandise Hierarchy

Business Overview

This service is used to update an existing subclass in Merchandising. When a subclass is updated, this service will first validate that all the required fields are present in the request payload. Business level validation on the input information will be performed which verifies if the subclass to be updated already exists. If the subclass already exists, it is updated. Active custom flex attributes that are associated with a subclass can also be updated using this service.

For more details on Subclass, see the Create Subclass service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/subclass/update

Input Payload Details

Table 5-1063 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclass	Yes	Number (4)	The unique number which identifies the subclass. This value cannot be modified.
subclassName	Yes	String (120)	The name of the subclass.
class	Yes	Number (4)	The number of the class which contains the subclass. This value must be predefined on the CLASS table. This value cannot be modified.
dept	Yes	Number (4)	The number of the department which contains the subclass. This value must be predefined on the DEPS table. This value cannot be modified.

Table 5-1063 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
intercompanyTransferPriceBasis	No	String (6)	This specifies the Default Intercompany Transfer Price that will be used during recording of transaction data postings on intercompany transfers and allocations. Valid values are 'W'(Weighted Average Cost) and 'S'(Supplier Cost). If not defined, the value will default to 'W' for subclasses under departments with cost-based profit calculation type or to 'R' (Retail Based) for subclasses under departments with retail-based profit calculation type.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
updSellDateOnStOrdRcpt	No	String (1)	This field is used to capture whether the selling phase start date at an item/location level would need to be updated in case of receipts against purchase orders having a store as their destination.
updSellDateOnWh2stRcpt	No	String (1)	This field is used to capture whether the selling phase start date at an item/location level would need to be updated in case of receipts against Warehouse to Store transfers or allocation.
updSellDateOnSt2stRcpt	No	String (1)	This field is used to capture whether the selling phase start date at an item/location level would need to be updated in case of receipts against Store to Store transfers.
minUnitsForSellDateReset	No	Number (12,4)	This field is to capture at the subclass level the minimum number of units that would trigger an update to the selling phase start date to the receipt date in case of a warehouse to store transfer/allocation or a store to store transfer.

Table 5-1064 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.

Table 5-1064 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "subclass": 1,
  "subclassName": "subclassName",
  "class": 1,
  "dept": 3041,
  "intercompanyTransferPriceBasis": null,
  "customFlexAttribute": [
    {
      "name": "REVIEW_DATE",
      "value": null,
      "valueDate": "2001-12-31"
    }
  ],
  "updSellDateOnStOrdRcpt": "N",
  "updSellDateOnWh2stRcpt": "N",
  "updSellDateOnSt2stRcpt": "N",
  "minUnitsForSellDateReset": 50
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

```
    }  
  ]  
}
```

Reclassification

The following services are included in this functional area:

- [Item Reclassification Download Service](#)
- [Item Reclassification Upload Service](#)
- [Merchandise Hierarchy Reclassification Upload Service](#)

Item Reclassification Download Service

Get Item Reclassifications

This section describes the Reclass Detail service.

Business Overview

This service is used to retrieve reclassification details for a given item.

Service Type

Get

ReST URL

Reclass/reclass?item={itemNumber}

Input Parameters

Parameter Name	Required	Description
Item	Yes	Item number

Output

RestReclassRecRDO

Parameter Name	Data Type
toClass	BigDecimal
reclassDate	Timestamp
reclassDesc	String
toSubclass	BigDecimal
reclassNo	BigDecimal
toDept	toDept

JSON Structure

```
[  
  {  
    "toClass": null,  
    "reclassDate": null,  
    "reclassDesc": null,  
  }  
]
```

```
        "toSubclass": null,
        "reclassNo": null,
        "toDept": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
RECLASS_HEAD	Yes	No	No	No
RECLASS_ITEM	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Item Reclassification Upload Service

Create Item Reclassification

Functional Area

Merchandise Hierarchy - Reclassification

Business Overview

This service allows external systems to create item reclassification events within Merchandising which are necessary in order to keep it in sync with the external system. Merchandising subscribes to item reclassification messages, which updates the department, class, and/or subclass for an item, based on the data that is published by the external system. Only the following item types can be interfaced using this API:

- Transaction level items without a parent
- Parent items, whose child items are the transaction level, such as with a fashion style (parent) and its SKUs (children)
- Complex pack items - but the reclassification cannot include the component items in the pack

The following item types cannot be reclassified:

- Child items with a parent - these are reclassified when the parent is updated
- Reference items - these below transaction level items are automatically reclassified with the transaction level item or its parent, whichever applies.
- Simple Packs - these are reclassified when the component item is reclassified

This service also allows a reclassification event to be created for a department/class/subclass combination that does not exist yet. This is valid as long as the merchandise hierarchy is scheduled to be created on or prior to the reclassification taking effect. In order to create a reclassification, both header and detail level records are required.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/itemReclassification/create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1065 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of item reclassification records.

Table 5-1066 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reclassNo	Yes	Number (4)	The unique number which identifies the reclass event.
description	Yes	String (120)	The description of the reclass event.
reclassDate	Yes	date	The date on which the reclass event is scheduled to take place.
toDept	Yes	Number (4)	The department to which the item will belong after the reclass event.
toClass	Yes	Number (4)	The class to which the item will belong after the reclass event.
toSubclass	Yes	Number (4)	The subclass to which the item will belong after the reclass event.
items	No	Collection of Object	References a collection of items in the reclassification event.

Table 5-1067 Items.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item in the reclass event.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "reclassNo": 1,
      "description": "reclassDesc",
      "reclassDate": "2001-12-31",
      "toDept": 3041,
      "toClass": 1,
      "toSubclass": 1,

```

```

        "items": [
            {
                "item": "100100076"
            }
        ]
    }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Item Reclassification Detail

Functional Area

Merchandise Hierarchy - Reclassification

Business Overview

This service allows external systems to add items to an existing item reclassification event in Merchandising in order to keep it in sync with the external system. Merchandising subscribes to item reclassification requests, which update the department, class, and/or subclass for the item, based on the data that is published by the external system.

Before an item is added to the reclassification event, the following things would be validated:

- Must be a level 1 item
- If the item is a pack, it should not be a simple pack
- Must not be on any approved order
- Must not be an orderable buyer pack that can be received as component items

- Must not be on an existing reclassification

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/itemReclassification/detail/
create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1068 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of item reclassification details.

Table 5-1069 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reclassNo	Yes	Number (4)	The unique number which identifies the reclass event.
description	Yes	String (120)	The description of the reclass event.
reclassDate	Yes	date	The date on which the reclass event is scheduled to take place.
toDept	Yes	Number (4)	The department to which the item will belong after the reclass event.
toClass	Yes	Number (4)	The class to which the item will belong after the reclass event.
toSubclass	Yes	Number (4)	The subclass to which the item will belong after the reclass event.
items	No	Collection of Object	References a collection of items in the reclassification event.

Table 5-1070 Items.Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item to be added to the reclassification event.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
```

```

    "reclassNo": 1,
    "description": "reclassDesc",
    "reclassDate": "2001-12-31",
    "toDept": 3041,
    "toClass": 1,
    "toSubclass": 1,
    "items": [
      {
        "item": "100100076"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Delete Item Reclassification

Functional Area

Merchandise Hierarchy - Reclassification

Business Overview

This service allows external systems to delete item reclassification events within Merchandising in order to keep it in sync with the external system. Reclassifications can be deleted using any of the following:

- Deleting a single reclassification event
- Deleting all reclassification events on a particular event date

- Deleting all reclassification events

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/itemReclassification/delete

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1071 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
reclassNo	No	Number (4)	If this is provided, then the particular reclass event will be deleted. The purgeAllInd must be N when this is provided.
reclassDate	No	date	If this is provided, then all reclass events with this date will be deleted. The purgeAllInd must be N when this is provided.
purgeAllInd	No	String (1)	If this field is Y, then all item reclass events will be deleted. If this is provided, both reclassNo and reclassDate must not be provided.

Sample Input Message

```
{
  "reclassNo": 1,
  "reclassDate": "2001-12-31",
  "purgeAllInd": "Y"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Item Reclassification Detail**Functional Area**

Merchandise Hierarchy - Reclassification

Business Overview

This service allows external systems to remove items from an existing item reclassification event in Merchandising in order to keep it in sync with the external system. The entire reclassification event gets deleted once there are no remaining item lines based on detail level deletion.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/itemReclassification/detail/delete

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details**Table 5-1072 Delete - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
reclassNo	Yes	Number (4)	The reclass event for which deletion activity is occurring.
items	Yes	Collection of Object	References a collection of items to be deleted from a reclassification event.

Table 5-1073 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	The item in the reclassification event that needs to be deleted.

Sample Input Message

```
{
  "reclassNo": 1,
  "items": [
    {
      "item": "100100076"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Merchandise Hierarchy Reclassification Upload Service

Create Merchandise Hierarchy Reclassification

Functional Area

Merchandise Hierarchy - Reclassification

Business Overview

This service allows Merchandising to subscribe to merchandise hierarchy reclassification requests from an external system and is intended to be used by retailers who manage their hierarchies in an external system which is outside Merchandising. This service creates pending merchandise hierarchy reclassification events within Merchandising based on the data sent by the external system.

When a new merchandise hierarchy reclassification is created, the service will first validate that all the required fields are present in the payload. There are certain fields that are required

irrespective of the hierarchy level, while there are a few which depend on other hierarchy configurations. After this is completed, business level validation will be carried out on the input information.

There is a separate batch process to read the information that is present in the pending merchandise hierarchy table and create the merchandise hierarchy information in Merchandising on the effective date.

Service Type

POST

ReST URL

MerchIntegrations/services/merchandiseHierarchy/reclassification/create

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1074 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (1)	Indicates the level of the merchandise hierarchy. Valid values are V (division), G (group), D (department), C (class), and S (subclass).
hierarchyId	Yes	Number (4)	Contains the merchandise hierarchy ID.
parentHierarchyId	No	Number (4)	The parent of the hierarchy identified in the hierarchyId field. This field is optional when hierarchyLevel is V (Division). For a Division, if parent is provided, it must be Company.
grandparentHierarchyId	No	Number (4)	The grandparent of the hierarchy identified in the hierarchyId field. This field will only be used when the hierarchyLevel indicates subclass. It will contain the number of the department the subclass belongs to.
hierarchyName	Yes	String (120)	The name of the hierarchy value.
effectiveDate	Yes	date	The date the hierarchy change will become effective.
actionType	No	String (1)	Indicates if this field is an addition or modification. Valid values are A and M. It is required on a create message and should not be populated on a modify message.
buyer	No	Number (4)	The number of the buyer associated with the entity. This value must be predefined on the BUYER table. This field should only hold a value if the hierarchy level indicates division, group, or department.

Table 5-1074 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	Number (1)	The code which indicates whether items in the department are normal merchandise (0) or consignment stock (1). This field will only be used if the hierarchy level indicates department.
totalMarketAmount	No	Number (24,4)	The total market amount that is expected for the entity. This field will only be used if the hierarchy value indicates division or department.
merchandiser	No	Number (4)	The number of the merchandiser associated with the entity. This value must be predefined on the MERCHANT table. This field should hold a value only if the hierarchy level indicates division, group, or department.
budgetedMarkupPercent	No	Number (12,4)	The budgeted markup percentage. The markup percent of cost. This field will only be used if the hierarchy level indicates department.
profitCalcType	No	Number (1)	The number which indicates whether profit will be calculated by direct cost (1) or retail inventory (2). This field will only be used if the hierarchy level indicates department.
markupCalcType	No	String (2)	The code letter which indicates how markup is calculated in the department. This field will only be used if the hierarchy level indicates department. Valid values are C (cost) and R (retail).
otbCalcType	No	String (1)	The code letter which indicates how OTB is calculated in the department. Valid values are cost (C) and retail (R). This field will only be used if the hierarchy value indicates department.
maximumAverageCounter	No	Number (5)	The maximum count of days with acceptable data to include in an average for items with the department. This field will only be used if the hierarchy level indicates department. It is required if RPM is installed.
averageTolerancePercent	No	Number (12,4)	The tolerance percentage value used in averaging for items within this value. This column will hold 70% as 70, not .70. This field will only be used if the hierarchy value indicates department.

Table 5-1074 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
budgetedIntakePercent	No	Number (12,4)	The budgeted intake percentage. The percent of the total take that is income. This field will only be used if the hierarchy value indicates department.
deptVatIncludeInd	No	String (1)	Indicates the default value for the class VAT indicator. When classes are initially set up they will inherit this value. This field will only be populated when the hierarchy level indicates department.
classVatInd	No	String (1)	Indicates if retail is displayed and held with or without VAT for items within a class. Valid values are Y (yes) and N (no). This field will only be used if the hierarchy level indicates class.

Sample Input Message

```
{
  "hierarchyLevel": "D",
  "hierarchyId": 100,
  "parentHierarchyId": 108,
  "grandparentHierarchyId": 1000,
  "hierarchyName": "Dept 100 MrchRcls Action New",
  "effectiveDate": null,
  "actionType": "A",
  "buyer": 305,
  "purchaseType": 0,
  "totalMarketAmount": 3,
  "merchandiser": 601,
  "budgetedMarkupPercent": 3,
  "profitCalcType": 1,
  "markupCalcType": "C",
  "otbCalcType": "C",
  "maximumAverageCounter": 10000,
  "averageTolerancePercent": 3,
  "budgetedIntakePercent": 15.00,
  "deptVatIncludeInd": "Y",
  "classVatInd": null
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Merchandise Hierarchy Reclassification**Functional Area**

Merchandise Hierarchy - Reclassification

Business Overview

This service allows for pending merchandise hierarchy reclassification events to be deleted in Merchandising. Prior to deleting a previously created reclassification event, the existence of any related child reclassification records will be checked and the necessary validations will be executed to ensure there are no conflicts.

Service Type

DELETE

ReST URL

MerchIntegrations/services/merchandiseHierarchy/reclassification/delete

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1075 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (1)	Indicates the level of the merchandise hierarchy. Valid values are V (division), G (group), D (department), C (class), and S (subclass).
hierarchyId	Yes	Number (4)	Contains the merchandise hierarchy ID.
parentHierarchyId	No	Number (4)	The parent of the hierarchy identified in the hierarchyId field.

Table 5-1075 (Cont.) Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
grandparentHierarchyId	No	Number (4)	The grandparent of the hierarchy identified in the hierarchyId field. This column will only be populated if the hierarchyLevel = subclass. In this case, the column will contain the dept number.

Sample Input Message

```
{
  "hierarchyLevel": "D",
  "hierarchyId": 100,
  "parentHierarchyId": 108,
  "grandparentHierarchyId": 1000
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Merchandise Hierarchy Reclassification**Functional Area**

Merchandise Hierarchy - Reclassification

Business Overview

This service allows for pending merchandise hierarchy reclassification events to be modified in Merchandising. In order to update a previously created reclassification, the hierarchy type must be already present in Merchandising. While updating the effective date of a reclassification that involves an addition, there should not be any child hierarchy addition with an earlier effective date. For example, if you are adding a department within Merchandising on a given date, there cannot be a reclassification event for adding a class within the department with an earlier effective date.

Service Type

PUT

ReST URL

MerchIntegrations/services/merchandiseHierarchy/reclassification/update

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1076 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (1)	Indicates the level of the merchandise hierarchy. Valid values are V (division), G (group), D (department), C (class), and S (subclass).
hierarchyId	Yes	Number (4)	Contains the merchandise hierarchy ID.
parentHierarchyId	No	Number (4)	The parent of the hierarchy identified in the hierarchyId field. This field is optional when hierarchyLevel is V (Division). For a Division, if parent is provided, it must be Company.
grandparentHierarchyId	No	Number (4)	The grandparent of the hierarchy identified in the hierarchyId field. This field will only be used when the hierarchyLevel indicates subclass. It will contain the number of the department the subclass belongs to.
hierarchyName	Yes	String (120)	The name of the hierarchy value.
effectiveDate	Yes	date	The date the hierarchy change will become effective.
buyer	No	Number (4)	The number of the buyer associated with the entity. This value must be predefined on the BUYER table. This field should only hold a value if the hierarchy level indicates division, group, or department.
purchaseType	No	Number (1)	The code which indicates whether items in the department are normal merchandise (0) or consignment stock (1). This field will only be used if the hierarchy level indicates department.

Table 5-1076 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
totalMarketAmount	No	Number (24,4)	The total market amount that is expected for the entity. This field will only be used if the hierarchy value indicates division or department.
merchandiser	No	Number (4)	The number of the merchandiser associated with the entity. This value must be predefined on the MERCHANT table. This field should hold a value only if the hierarchy level indicates division, group, or department.
budgetedMarkupPercent	No	Number (12,4)	The budgeted markup percentage. The markup percent of cost. This field will only be used if the hierarchy level indicates department.
profitCalcType	No	Number (1)	The number which indicates whether profit will be calculated by direct cost (1) or retail inventory (2). This field will only be used if the hierarchy level indicates department.
markupCalcType	No	String (2)	The code letter which indicates how markup is calculated in the department. This field will only be used if the hierarchy level indicates department. Valid values are C (cost) and R (retail).
otbCalcType	No	String (1)	The code letter which indicates how OTB is calculated in the department. Valid values are cost (C) and retail (R). This field will only be used if the hierarchy value indicates department.
maximumAverageCounter	No	Number (5)	The maximum count of days with acceptable data to include in an average for items with the department. This field will only be used if the hierarchy level indicates department.
averageTolerancePercent	No	Number (12,4)	The tolerance percentage value used in averaging for items within this value. This column will hold 70% as 70, not .70. This field will only be used if the hierarchy value indicates department.
budgetedIntakePercent	No	Number (12,4)	The budgeted intake percentage. The percent of the total take that is income. This field will only be used if the hierarchy value indicates department.

Table 5-1076 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deptVatIncludeInd	No	String (1)	Indicates the default value for the class VAT indicator. When classes are initially set up they will inherit this value. This field will only be populated when the hierarchy level indicates department.
classVatInd	No	String (1)	Indicates if retail is displayed and held with or without VAT for items within a class. Valid values are Y (yes) and N (no). This field will only be used if the hierarchy level indicates class.

Sample Input Message

```
{
  "hierarchyLevel": "D",
  "hierarchyId": 100,
  "parentHierarchyId": 108,
  "grandparentHierarchyId": 1000,
  "hierarchyName": "merchHierName",
  "effectiveDate": "2001-12-31",
  "buyer": 1000,
  "purchaseType": 1,
  "totalMarketAmount": 3,
  "merchandiser": 1000,
  "budgetedMarkupPercent": 3,
  "profitCalcType": 1,
  "markupCalcType": "C",
  "otbCalcType": "C",
  "maximumAverageCounter": 10000,
  "averageTolerancePercent": 3,
  "budgetedIntakePercent": 15.00,
  "deptVatIncludeInd": "Y",
  "classVatInd": null
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Organizational Hierarchy

The following services are included in this functional area:

- [Organizational Hierarchy Download Service](#)
 - [Get Organizational Hierarchy for Provided Level](#)
 - [Get Organizational Hierarchy for Provided Level and Hierarchy Identifier](#)
 - [Get Organizational Hierarchy](#)
- [Organizational Hierarchy Upload Service](#)
 - [Create Organizational Hierarchy](#)
 - [Delete Organizational Hierarchy](#)
 - [Update Organizational Hierarchy](#)
- [Stores](#)
 - [Store Download Service](#)
 - * [Get Basic Store Details](#)
 - * [Get Store Details](#)
 - * [Get Store Details for Provided Store](#)
 - [Store Upload Service](#)
 - * [Create Store](#)
 - * [Create Store Address](#)
 - * [Create Store Department Up Charges](#)
 - * [Create Store Hours](#)
 - * [Create Store Location Trait](#)
 - * [Create Walk-Through Store](#)
 - * [Delete Store](#)
 - * [Delete Store Address](#)
 - * [Delete Store Department Up Charges](#)
 - * [Delete Store Hours](#)

- * [Delete Store Location Trait](#)
- * [Delete Walk-Through Store](#)
- * [Update Store](#)
- * [Update Store Address](#)
- * [Update Store Department Up Charges](#)
- * [Update Store Hours](#)
- [Stores by Hierarchy Level Upload Service](#)
 - * [Create Store Location Trait by Hierarchy Level](#)
 - * [Delete Store Location Trait by Hierarchy Level](#)
- [Warehouse](#)
 - [Warehouse Download Service](#)
 - * [Get Basic Warehouse Details](#)
 - * [Get Warehouse Details](#)
 - * [Get Warehouse Details for Provided Warehouse](#)

Get Company Closings

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/companyClosings/get

Functional Area

Foundation

Business Overview

This service retrieves company-wide closings along with their associated location-level exceptions. This service accepts optional query parameters for closed date to retrieve details for the specified closed date.

Input Parameters

Response Code: 200 (Success)

Company Closings - Object. See list of elements for detail

LocationExceptions - Object. See list of elements for detail

Translations - Object. See list of elements for detail

Sample Response Message

```
{
  "closedDate": "2026-06-29",
  "closedDescription": "Christmas",
  "locationExceptions": [
```

```
{
  "location": 6000,
  "locationType": "S",
  "openForSales": "N",
  "openForShipping": "Y",
  "openForReceiving": "Y"
},
"translations": [
  {
    "languageCode": 3,
    "languageName": "French",
    "isoCode": "FR",
    "closedDescription": "Noel"
  }
]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Get Location Closings

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/locationClosings/get

Functional Area

Foundation

Business Overview

This service retrieves location-specific closings for one or more specified locations, or for all locations. This service accepts optional query parameters for location, location type and closed date to retrieve details for the specified location, location type and closed date.

Input Parameters

Response Code: 200 (Success)

Location Closings - Object. See list of elements for detail

Translations - Object. See list of elements for detail

Sample Response Message

```
{
  "location": 6000,
  "locationType": "S",
  "closedDate": "2026-06-29",
  "closedForSales": "N",
  "closedForShipping": "Y",
  "closedForReceiving": "Y",
  "reason": "Christmas",
  "translations": [
    {
      "languageCode": 3,
      "languageName": "French",
      "isoCode": "FR",
      "reason": "Noel"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Get Locations for AIF Applications

Business Overview

The service is used to fetch all locations from Merchandising for AIF application usage, data warehousing needs, or other downstream consuming systems. This service follows the AIF publish pattern and provides cache-backed JSON messages for consumers that need current location data.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and interacts with webhooks, refer to the common Publish API operational guidance.

Core business tables: STORE, WH, PARTNER, ADDR

Additional business tables: DISTRICT, REGION, AREA, CHAIN, COMPANY and related foundation reference tables

JSON cache table: MERCHAPI_AIF_LOCATION

JSON generation view: V_MERCHAPI_AIF_LOCATION_JSON

Builds a unified location message for stores, warehouses, and external finishers, including address, hierarchy, channel, banner, and operational attributes.

A location is published when it is present in the location JSON view and has a cacheable location message for AIF downstream consumption.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_AIF_LOCATION_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the location no longer exists in V_MERCHAPI_AIF_LOCATION_JSON.

ICL (Integration Change Log) table: ICL_RMS_ADDR, ICL_RMS_PARTNER, ICL_RMS_STORE, ICL_RMS_WH (ICL consumer name = MERCHAPI_4)

ICL entries are created using the delivered entity map and table map for foundation/aif/locations. Qualifying inserts, updates, and deletes are written with change type U for cache refresh processing.

Batch Configuration. Background-engine execution is the delivered asynchronous option for this API.

Process configuration name: API_AIF_LOCATION

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM batch scheduler based delta processing is not supported for this API.

Webhook configuration api name: foundation/aif/locations

Special handling:

Additional GET query parameters: location, locationType.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime profiles.

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through `API_NO_OF_THREADS`, and commit sizing is controlled through `API_COMMIT_MAX_COUNTER`. Choose these settings together based on expected row volume, average message size, and the SQL cost of building each JSON message.

Dropping indexes and constraints during seeding

`DROP_INDEX_ON_REFRESH` applies only to initial seeding, including truncate-and-load execution. When enabled, indexes and constraints are dropped before seeding and rebuilt after the load completes. `GATHER_STATS_ON_REFRESH` should normally remain enabled unless implementation testing shows a specific reason to change it.

REST fetch considerations

Select REST page size based on average payload size so the average response remains below about 10 MB per call. For full downloads, omit the `since` query parameter. For delta downloads, use a short time window; wider outage or recovery windows should be handled through smaller polling windows, using `optimalBefore` where applicable.

Input Parameters**Response Code: 200 (Success)**

`PageResultsItemsMerchApiAifLocation` - Object. See list of elements for detail

`MerchApiAifLocation` - Object. See list of elements for detail

Location address and operational attribute objects - Object. See list of elements for detail

`LinkItems` - Object. See list of elements for detail

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "location": 101,
      "locationType": "S",
      "currencyCode": "USD",
      "locationName": "Store 101",
      "addressType": "01",
      "address1": "100 Main Street",
      "city": "Austin",
```

```

        "state": "TX",
        "post": "78701",
        "country": "US",
        "district": 10,
        "region": 1,
        "channelId": 1,
        "channelName": "Stores",
        "stockholdingFlag": "Y",
        "createDateTime": "2001-12-31T10:15:30Z",
        "updateDateTime": "2001-12-31T10:15:30Z"
    }
],
"hasMore": false,
"limit": 1,
"count": 1,
"links": []
}

```

Manage Company Closings

Functional Area

Foundation

Business Overview

This service endpoint will be used to create, update or delete company-wide closings and any exception at location level.

Service Type

POST

ReST URL

MerchIntegrations/services/foundation/companyClosings/manage

Input Payload Details

Manage - Object. See list of elements for detail

CompanyClosings - Object. See list of elements for detail

LocationExceptions - Object. See list of elements for detail

Translations - Object. See list of elements for detail

Sample Input Message

```

{
  "companyClosings": [
    {
      "action": "CREATE",
      "closedDate": "2001-12-31",
      "closedDescription": "Independence Day",
      "locationExceptions": [
        {
          "action": "CREATE",

```

```

"location": 1521,
"locationType": "S",
"openForSales": "N",
"openForShipping": "N",
"openForReceiving": "Y"
},
"translations": [
{
"action": "CREATE",
"lang": 3,
"closedDescription": "Jour de l\u0027ind\u00e9pendance"
}
]
}
}
}

```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
"status": "ERROR",
"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
]
}

```

Manage Location Closings

Functional Area

Foundation

Business Overview

This service endpoint will be used to create, update or delete location-specific closings, for individual locations.

Service Type

POST

ReST URL

MerchIntegrations/services/foundation/locationClosings/manage

Input Payload Details

Manage - Object. See list of elements for detail

LocationClosings - Object. See list of elements for detail

Translations - Object. See list of elements for detail

Sample Input Message

```
{
  "locationClosings": [
    {
      "action": "CREATE",
      "location": 1521,
      "locationType": "S",
      "closedDate": "2001-12-31",
      "closedForSales": "N",
      "closedForShipping": "Y",
      "closedForReceiving": "Y",
      "reason": "Christmas",
      "translations": [
        {
          "action": "CREATE",
          "lang": 3,
          "reason": "Noël"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)**Response Code: 400 (Error)**

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

```
}  
}
```

Manage Warehouse

Functional Area

Organizational Hierarchy - Warehouses

Business Overview

This service supports the maintenance of physical warehouses and their associated virtual warehouses, along with related data such as addresses and translations based on the request from an external system if Merchandising is not the system of record for organizational hierarchy information. Warehouse creation also supports importing custom flexible attributes (CFAS) from an external system.

Valid actions are Create, Update and Delete.

When the provided Action is 'Create', a new warehouse is created in Merchandising synchronously after the payload is validated first. When creating a new physical warehouse, the payload must include mandatory address details and at least one virtual warehouse.

When the provided Action is 'Update', an existing warehouse's data can be updated after the data has been validated.

When the provided Action is 'Delete', an existing physical warehouse and associated data can be deleted. After the data has been validated, the service will add the warehouse to the DAILY_PURGE table for processing deletion through a batch process.

Physical Warehouse Translations

This sub node allows to create, update, or delete translations of a warehouse's name and secondary name in Merchandising. This service will accept a warehouse, language and name specified in the language, to add or update a translation, or a warehouse and language to delete a translation.

Addresses

This sub node supports maintenance of addresses along with their translations for an existing physical warehouse in Merchandising based on the request from an external system if Merchandising is not the system of record for organizational hierarchy information. Address creation also supports importing custom flexible attributes (CFAS) from an external system.

Valid actions are Create, Update and Delete.

When the provided Action is 'Create', a new address is created in Merchandising after the payload has been validated.

When the provided Action is 'Update', an existing warehouse's address data can be updated after the payload has been validated.

When the provided Action is 'Delete', an existing address for the physical warehouse and its associated data can be deleted after the payload has been validated.

Address Translations

This sub node allows to create, update, or delete translations of a warehouse's address in Merchandising. This service will accept a warehouse, in the language, to add or update a translation, or a warehouse and language to delete a translation.

Virtual Warehouses

This sub node supports the maintenance of virtual warehouses and along with related translations based on the request from an external system if Merchandising is not the system of record for organizational hierarchy information. This also supports importing custom flexible attributes (CFAS) from an external system.

Valid actions are Create, Update and Delete.

When a physical warehouse is scheduled for one or more stock counts that are not yet within the lockout period. Newly created virtual warehouses will be added to the existing open stock counts.

Each channel that includes a finishing virtual warehouse must also include a corresponding non-finishing virtual warehouse.

If replenishment is used in Merchandising, each physical warehouse is expected to include a replenishable virtual warehouse.

When the provided Action is 'Create', an additional virtual warehouse for the existing physical warehouse is created after the payload has been validated.

When the provided Action is 'Update', an existing virtual warehouse's data is updated after the payload has been validated.

When the provided Action is 'Delete', an existing virtual warehouse and associated data can be deleted after the payload has been validated. The service will add the warehouse to the DAILY_PURGE table for processing deletion through a batch process.

Virtual Warehouse Translations

This sub node allows you to create, update, or delete translations of a virtual warehouse's name and secondary name in Merchandising. This service will accept a warehouse, language and name specified in the language, to add or update a translation, or a warehouse and language to delete a translation.

Note: For action UPDATE and subnodes Physical Warehouse Translations, Address, Address Translations, Virtual Warehouse Translations and CFAS, all the fields are expected to be available. If not provided, all such fields will be updated to null.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/warehouse/manage

Input Payload Details

Manage - Object. See list of elements for detail

Items - Object. See list of elements for detail

Address - Object. See list of elements for detail

Translation - Object. See list of elements for detail

CustomFlexAttribute - Object. See list of elements for detail

VirtualWarehouse - Object. See list of elements for detail

VirtualWarehouse.Translation - Object. See list of elements for detail

ManageError - Object. See list of elements for detail

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "physicalWarehouse": 990000001,
      "name": "String",
      "secondaryName": "String",
      "clearSecondaryName": "N",
      "email": "String",
      "clearEmail": "N",
      "currencyCode": "USD",
      "inboundHandlingDays": 1,
      "deliveryPolicy": "NDD",
      "costLocation": 6,
      "breakPackWarehouseInd": "N",
      "redistributionWarehouseInd": "N",
      "taxRegion": 1000,
      "taxId": "123456789",
      "clearTaxId": "N",
      "dunsNumber": "123456789",
      "clearDunsNumber": "N",
      "dunsLoc": "1234",
      "clearDunsLocation": "N",
      "reportingLevelType": null,
      "clearReportingLevelType": "N",
      "reportingLevel": null,
      "address": [
        {
          "action": "CREATE",
          "addressType": "01",
          "externalReferenceId": "ADDR_01",
          "primaryAddressType": "Y",
          "primaryAddress": "Y",
          "add1": "Primary Address",
          "add2": "Suite 100",
          "add3": null,
          "city": "Redwood City",
          "state": "CA",
          "country": "US",
          "post": "94065",
          "county": "San Mateo",
          "latitude": "37.5300",
          "longitude": "-122.2600",
          "jurisdictionCode": null,
          "contactName": "Contact",
          "contactPhone": "6505550100",
          "contactTelex": null,
          "contactFax": null,
          "contactEmail": "String",
          "phone2Type": "MA",
          "phone2": "6505550101",

```

```

"phone3Type": "W0",
"phone3": "6505550102",
"oracleVendorSiteId": null,
"translation": [
{
"action": "CREATE",
"addressType": "01",
"language": 2,
"add1": "Primaere Adresse",
"add2": "Uebersetzte Suite 100",
"add3": null,
"city": "Redwood City DE",
"contactName": "Uebersetzter Kontakt",
"county": "Uebersetzter Bezirk"
}
],
"customFlexAttribute": [
{
"name": "ADDR_PC",
"value": "10001",
"valueDate": "2001-12-31"
}
]
},
"virtualWarehouse": [
{
"action": "CREATE",
"virtualWarehouse": 9900000002,
"name": "String",
"secondaryName": "String",
"clearSecondaryName": "N",
"primaryVirtualWarehouseInd": "Y",
"orgEntityType": "R",
"channel": 1,
"virtualWarehouseType": "CS_NT",
"transferEntity": 1379346183,
"organizationalUnit": 385168665527164,
"legalEntity": null,
"clearLegalEntity": "N",
"pricingLocation": 6,
"defaultWarehouse": null,
"roundingWarehouse": null,
"protectedInd": "N",
"restrictedInd": "N",
"forecastableInd": "Y",
"replenishmentWarehouseInd": "Y",
"replenishmentWarehouseLink": null,
"sourceSequence": null,
"investmentBuyWarehouseInd": "N",
"investmentBuyWarehouseLink": null,
"inventoryBuyAutoClearInventory": "N",
"finisherInd": "N",
"customerOrderInd": "N",
"customerShippingInd": "N",
"giftWrappingInd": "N",

```



```

"translation": [
{
"action": "CREATE",
"language": 2,
"warehouseName": "Physisches Lager",
"warehouseSecondaryName": "Physisches Lager Zweitname"
}
],
"customFlexAttribute": [
{
"name": "ADDR_PC",
"value": "10001",
"valueDate": "2001-12-31"
}
]
},
"translation": [
{
"action": "CREATE",
"language": 2,
"warehouseName": "Physisches Lager",
"warehouseSecondaryName": "Physisches Lager Zweitname"
}
],
"customFlexAttribute": [
{
"name": "ADDR_PC",
"value": "10001",
"valueDate": "2001-12-31"
}
]
]
}

```

Response Code: 200 (Success)**Response Code: 400 (Error)**

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

ManageError - Object. See list of elements for detail

Sample Error Message

```

{
"status": "ERROR",
"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",

```

```

    "field": "createRecord.arg0.approveInd",
    "inputValue": "X"
  },
  "businessError": [
    {
      "physicalWarehouse": 987654321,
      "addressType": "01",
      "externalReferenceId": "ADDR_01",
      "virtualWarehouse": 987654322,
      "language": 2,
      "cfaName": "ADDR_PC",
      "errorMessage": [
        "Invalid value for deliveryPolicy."
      ]
    }
  ]
}

```

Organizational Hierarchy Download Service

The following services are included in this functional area:

- [Get Organizational Hierarchy for Provided Level](#)
- [Get Organizational Hierarchy for Provided Level and Hierarchy Identifier](#)
- [Get Organizational Hierarchy](#)

Get Organizational Hierarchy for Provided Level

Endpoints

MerchIntegrations/services/foundation/orghier

MerchIntegrations/services/foundation/orghier/{orgLevel}

MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Functional Area

Organizational Hierarchy

Business Overview

This service publishes organizational hierarchy master data to downstream consuming systems. It publishes the company, chain, area, region, and district nodes used to describe the organizational structure, together with hierarchy name, manager name, currency code, and parent-hierarchy linkage. Functionally, only the five organizational levels COMPANY, CHAIN, AREA, REGION, and DISTRICT are published; store and warehouse nodes are not part of this external API even though the broader org-hierarchy framework also uses those levels elsewhere.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: COMPHEAD, CHAIN, AREA, REGION, DISTRICT

Additional business tables: SYSTEM_OPTIONS, CODE_DETAIL

JSON cache table: MERCHAPI_EXT_ORGHIER

JSON generation view: V_MERCHAPI_EXT_ORGHIER_JSON

Builds the payload by unioning COMPHEAD, CHAIN, AREA, REGION, and DISTRICT, uses SYSTEM_OPTIONS to supply the company currency code, and uses CODE_DETAIL to translate internal hierarchy codes into the published hierarchy-level values and their numeric sequence order.

The view only publishes the five organizational levels COMPANY, CHAIN, AREA, REGION, and DISTRICT; store and warehouse nodes are not part of this external API even though the broader org-hierarchy framework also uses those levels elsewhere.

Initial refresh supports both truncate-and-load and rebuild behavior, and rebuild marks cache rows deleted when a hierarchy node no longer exists in V_MERCHAPI_EXT_ORGHIER_JSON.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_ORGHIER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the hierarchy node no longer exists in V_MERCHAPI_EXT_ORGHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_COMPHEAD, ICL_RMS_CHAIN, ICL_RMS_AREA, ICL_RMS_REGION, ICL_RMS_DISTRICT (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on COMPHEAD, CHAIN, AREA, REGION, DISTRICT.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ORGHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ORG_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_ORG_HIER_ADHOC_JOB

Webhook configuration api name: foundation/orghier

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/orghier

MerchIntegrations/services/foundation/orghier/{orgLevel}

MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Input Parameters for MerchIntegrations/services/foundation/orghier

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (15)	Offset key ([level order],[hierarchy id] Example: 30,17)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1077 Input Parameter for "MerchIntegrations/services/foundation/orghier/{orgLevel}"

Parameter Name	Required	Data Type	Description
orgLevel	Yes	String	Organization level - COMPANY or CHAIN or AREA or REGION or DISTRICT
offsetkey	No	String (15)	Offset key ([level order],[hierarchy id] Example: 30,17)
limit	No	String (6)	Pagination limit. Default value is 1000.

Table 5-1077 (Cont.) Input Parameter for "MerchIntegrations/services/foundation/orghier/{orgLevel}"

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1078 Input Parameter for "MerchIntegrations/services/foundation/orghier/{param1}/{param2}"

Parameter Name	Required	Data Type	Description
param1	Yes	String	Organization level - COMPANY or CHAIN or AREA or REGION or DISTRICT
param2	Yes	String (10)	Hierarchy Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/orghier/{orgLevel}

Parameter Name	Required	Data Type	Description
orgLevel	Yes	String	Level of the organization hierarchy. Valid values are COMPANY, CHAIN, AREA, REGION, DISTRICT.
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Parameter Name	Required	Data Type	Description
param1	Yes	String	This is the organization hierarchy level. Valid values are COMPANY, CHAIN, AREA, REGION, DISTRICT.
param2	Yes	String	This is the hierarchy ID.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Diff ID

Table 5-1079 PageResultsItemsMerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1080 MerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for organizational hierarchy details.
hierarchyLevel	No	String (10)	This field contains the organizational hierarchy level.
hierarchyId	No	Number(10,0)	This field contains the number that uniquely identifies the hierarchy value.
hierarchyName	No	String (120)	This field contains the description of the hierarchy value.
managerName	No	String (120)	This field contains the manager's name.
currencyCode	No	String (3)	This field contains the currency code for the hierarchy.
parentHierarchyId	No	Number(10,0)	This field contains the identifier of the parent hierarchy value.
parentHierarchyLevel	No	String (10)	This field contains the parent level of the current organizational hierarchy.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
cacheTimestamp	No	dateTime	This field specifies the date and time when the organization hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1081 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "hierarchyId": 1,
      "hierarchyName": "hierDesc",
      "managerName": "mgrName",
      "currencyCode": "USD",
      "parentHierarchyId": 1,
      "parentHierarchyLevel": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",
      "cacheTimestamp": "2001-12-31T23:59:59.000Z",
      "cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
    }
  ],
  "hasMore": true,
  "limit": 1000,
  "count": 1000,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
      "rel": "next"
    }
  ]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_CHAIN	Yes	No	Yes	Yes
ICL_RMS_COMPHEAD	Yes	No	Yes	Yes
ICL_RMS_AREA	Yes	No	Yes	Yes
ICL_RMS_REGION	Yes	No	Yes	Yes
ICL_RMS_DISTRICT	Yes	No	Yes	Yes
MERCHAPI_EXT_ORGHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ORGHIER_JSON	Yes	No	No	No

Get Organizational Hierarchy for Provided Level and Hierarchy Identifier

Endpoints

MerchIntegrations/services/foundation/orghier

MerchIntegrations/services/foundation/orghier/{orgLevel}

MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Functional Area

Organizational Hierarchy

Business Overview

This service publishes organizational hierarchy master data to downstream consuming systems. It publishes the company, chain, area, region, and district nodes used to describe the organizational structure, together with hierarchy name, manager name, currency code, and parent-hierarchy linkage. Functionally, only the five organizational levels COMPANY, CHAIN, AREA, REGION, and DISTRICT are published; store and warehouse nodes are not part of this external API even though the broader org-hierarchy framework also uses those levels elsewhere.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: COMPHEAD, CHAIN, AREA, REGION, DISTRICT

Additional business tables: SYSTEM_OPTIONS, CODE_DETAIL

JSON cache table: MERCHAPI_EXT_ORGHIER

JSON generation view: V_MERCHAPI_EXT_ORGHIER_JSON

Builds the payload by unioning COMPHEAD, CHAIN, AREA, REGION, and DISTRICT, uses SYSTEM_OPTIONS to supply the company currency code, and uses CODE_DETAIL to translate internal hierarchy codes into the published hierarchy-level values and their numeric sequence order.

The view only publishes the five organizational levels COMPANY, CHAIN, AREA, REGION, and DISTRICT; store and warehouse nodes are not part of this external API even though the broader org-hierarchy framework also uses those levels elsewhere.

Initial refresh supports both truncate-and-load and rebuild behavior, and rebuild marks cache rows deleted when a hierarchy node no longer exists in V_MERCHAPI_EXT_ORGHIER_JSON.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_ORGHIER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the hierarchy node no longer exists in V_MERCHAPI_EXT_ORGHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_COMPHEAD, ICL_RMS_CHAIN, ICL_RMS_AREA, ICL_RMS_REGION, ICL_RMS_DISTRICT (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on COMPHEAD, CHAIN, AREA, REGION, DISTRICT.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ORGHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ORG_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_ORG_HIER_ADHOC_JOB

Webhook configuration api name: foundation/orghier

Special handling

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/orghier

MerchIntegrations/services/foundation/orghier/{orgLevel}

MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Input Parameters for MerchIntegrations/services/foundation/orghier

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (15)	Offset key ([level order],[hierarchy id] Example: 30,17)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1082 Input Parameter for "MerchIntegrations/services/foundation/orghier/{orgLevel}"

Parameter Name	Required	Data Type	Description
orgLevel	Yes	String	Organization level - COMPANY or CHAIN or AREA or REGION or DISTRICT
offsetkey	No	String (15)	Offset key ([level order],[hierarchy id] Example: 30,17)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1083 Input Parameter for "MerchIntegrations/services/foundation/orghier/{param1}/{param2}"

Parameter Name	Required	Data Type	Description
param1	Yes	String	Organization level - COMPANY or CHAIN or AREA or REGION or DISTRICT
param2	Yes	String (10)	Hirerachy Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/orghier/{orgLevel}

Parameter Name	Required	Data Type	Description
orgLevel	Yes	String	Level of the organization hierarchy. Valid values are COMPANY, CHAIN, AREA, REGION, DISTRICT.
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Parameter Name	Required	Data Type	Description
param1	Yes	String	This is the organization hierarchy level. Valid values are COMPANY, CHAIN, AREA, REGION, DISTRICT.
param2	Yes	String	This is the hierarchy ID.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Diff ID

Table 5-1084 PageResultsItemsMerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1085 MerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for organizational hierarchy details.
hierarchyLevel	No	String (10)	This field contains the organizational hierarchy level.
hierarchyId	No	Number(10,0)	This field contains the number that uniquely identifies the hierarchy value.
hierarchyName	No	String (120)	This field contains the description of the hierarchy value.
managerName	No	String (120)	This field contains the manager's name.

Table 5-1085 (Cont.) MerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencyCode	No	String (3)	This field contains the currency code for the hierarchy.
parentHierarchyId	No	Number(10,0)	This field contains the identifier of the parent hierarchy value.
parentHierarchyLevel	No	String (10)	This field contains the parent level of the current organizational hierarchy.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
cacheTimestamp	No	dateTime	This field specifies the date and time when the organization hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1086 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
```

```

"action": "INSERT",
"hierarchyLevel": null,
"hierarchyId": 1,
"hierarchyName": "hierDesc",
"managerName": "mgrName",
"currencyCode": "USD",
"parentHierarchyId": 1,
"parentHierarchyLevel": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_CHAIN	Yes	No	Yes	Yes
ICL_RMS_COMPHEAD	Yes	No	Yes	Yes
ICL_RMS_AREA	Yes	No	Yes	Yes
ICL_RMS_REGION	Yes	No	Yes	Yes
ICL_RMS_DISTRICT	Yes	No	Yes	Yes
MERCHAPI_EXT_ORGHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ORGHIER_JSON	Yes	No	No	No

Get Organizational Hierarchy

Endpoints

MerchIntegrations/services/foundation/orghier

MerchIntegrations/services/foundation/orghier/{orgLevel}

MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Functional Area

Organizational Hierarchy

Business Overview

This service publishes organizational hierarchy master data to downstream consuming systems. It publishes the company, chain, area, region, and district nodes used to describe the organizational structure, together with hierarchy name, manager name, currency code, and parent-hierarchy linkage. Functionally, only the five organizational levels COMPANY, CHAIN, AREA, REGION, and DISTRICT are published; store and warehouse nodes are not part of this external API even though the broader org-hierarchy framework also uses those levels elsewhere.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: COMPHEAD, CHAIN, AREA, REGION, DISTRICT

Additional business tables: SYSTEM_OPTIONS, CODE_DETAIL

JSON cache table: MERCHAPI_EXT_ORGHIER

JSON generation view: V_MERCHAPI_EXT_ORGHIER_JSON

Builds the payload by unioning COMPHEAD, CHAIN, AREA, REGION, and DISTRICT, uses SYSTEM_OPTIONS to supply the company currency code, and uses CODE_DETAIL to translate internal hierarchy codes into the published hierarchy-level values and their numeric sequence order.

The view only publishes the five organizational levels COMPANY, CHAIN, AREA, REGION, and DISTRICT; store and warehouse nodes are not part of this external API even though the broader org-hierarchy framework also uses those levels elsewhere.

Initial refresh supports both truncate-and-load and rebuild behavior, and rebuild marks cache rows deleted when a hierarchy node no longer exists in V_MERCHAPI_EXT_ORGHIER_JSON.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_ORGHIER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the hierarchy node no longer exists in V_MERCHAPI_EXT_ORGHIER_JSON.

ICL (Integration Change Log) table: ICL_RMS_COMPHEAD, ICL_RMS_CHAIN, ICL_RMS_AREA, ICL_RMS_REGION, ICL_RMS_DISTRICT (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on COMPHEAD, CHAIN, AREA, REGION, DISTRICT.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ORGHIER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ORG_HIER_ADHOC_PROCESS ->
MERCHAPI_EXT_ORG_HIER_ADHOC_JOB

Webhook configuration api name: foundation/orghier

Special handling**Service Type**

GET

ReST URL

MerchIntegrations/services/foundation/orghier

MerchIntegrations/services/foundation/orghier/{orgLevel}

MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Input Parameters for MerchIntegrations/services/foundation/orghier

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String (15)	Offset key ([level order],[hierarchy id] Example: 30,17)
limit	No	String (6)	Pagination limit. Default value is 1000.

Parameter Name	Required	Data Type	Description
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1087 Input Parameter for "MerchIntegrations/services/foundation/orghier/{orgLevel}"

Parameter Name	Required	Data Type	Description
orgLevel	Yes	String	Organization level - COMPANY or CHAIN or AREA or REGION or DISTRICT
offsetkey	No	String (15)	Offset key ([level order],[hierarchy id] Example: 30,17)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1088 Input Parameter for "MerchIntegrations/services/foundation/orghier/{param1}/{param2}"

Parameter Name	Required	Data Type	Description
param1	Yes	String	Organization level - COMPANY or CHAIN or AREA or REGION or DISTRICT
param2	Yes	String (10)	Hierarchy Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Table 5-1088 (Cont.) Input Parameter for "MerchIntegrations/services/foundation/orghier/{param1}/{param2}"

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/orghier/{orgLevel}

Parameter Name	Required	Data Type	Description
orgLevel	Yes	String	Level of the organization hierarchy. Valid values are COMPANY, CHAIN, AREA, REGION, DISTRICT.
offsetkey	No	String	Offset Key. Valid Value is combination of sort_order_seq and node id.
limit	No	BigDecimal	Pagination limit. Default Value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Input Parameters for MerchIntegrations/services/foundation/orghier/{param1}/{param2}

Parameter Name	Required	Data Type	Description
param1	Yes	String	This is the organization hierarchy level. Valid values are COMPANY, CHAIN, AREA, REGION, DISTRICT.
param2	Yes	String	This is the hierarchy ID.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output - Diff ID**Table 5-1089 PageResultsItemsMerchApiOrghier - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-1089 (Cont.) PageResultsItemsMerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1090 MerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for organizational hierarchy details.
hierarchyLevel	No	String (10)	This field contains the organizational hierarchy level.
hierarchyId	No	Number(10,0)	This field contains the number that uniquely identifies the hierarchy value.
hierarchyName	No	String (120)	This field contains the description of the hierarchy value.
managerName	No	String (120)	This field contains the manager's name.
currencyCode	No	String (3)	This field contains the currency code for the hierarchy.
parentHierarchyId	No	Number(10,0)	This field contains the identifier of the parent hierarchy value.
parentHierarchyLevel	No	String (10)	This field contains the parent level of the current organizational hierarchy.

Table 5-1090 (Cont.) MerchApiOrghier - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
cacheTimestamp	No	dateTime	This field specifies the date and time when the organization hierarchy record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1091 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "hierarchyLevel": null,
      "hierarchyId": 1,
      "hierarchyName": "hierDesc",
      "managerName": "mgrName",
      "currencyCode": "USD",
      "parentHierarchyId": 1,
      "parentHierarchyLevel": null,
      "createDateTime": "2001-12-31T23:59:59.000Z",
      "updateDateTime": "2001-12-31T23:59:59.000Z",

```

```

"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_CHAIN	Yes	No	Yes	Yes
ICL_RMS_COMPHEAD	Yes	No	Yes	Yes
ICL_RMS_AREA	Yes	No	Yes	Yes
ICL_RMS_REGION	Yes	No	Yes	Yes
ICL_RMS_DISTRICT	Yes	No	Yes	Yes
MERCHAPI_EXT_ORGHIER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ORGHIER_JSON	Yes	No	No	No

Organizational Hierarchy Upload Service

The following services are included in this functional area:

- [Create Organizational Hierarchy](#)
- [Delete Organizational Hierarchy](#)
- [Update Organizational Hierarchy](#)

Create Organizational Hierarchy

Functional Area

Organizational Hierarchy

Business Overview

If Merchandising is not the system of record for organizational hierarchy information for an implementation, then this service may be used to create hierarchy based on an external system.

The following organizational hierarchy elements can be created using this service: chain, area, region, or district. The organizational hierarchy must be created from the highest level down.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/create

Input Payload Details

Table 5-1092 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The ID of the specified hierarchy level. This value must be unique among other values of the same hierarchy level.
hierarchyDescription	Yes	String (120)	The description of the hierarchy value.
hierarchyLevel	Yes	String (2)	The hierarchy level of the hierarchy value. This field will indicate the level of the organizational hierarchy to which the cost change applies. Valid values are CH (chain), AR (area), RE (region), DI (district).
parentHierarchyId	No	Number (10)	The ID of the hierarchy value's parent hierarchy value. This value must be predefined on either the CHAIN, AREA or REGION table.
managerName	No	String (120)	The manager name of the hierarchy value.
currencyCode	No	String (3)	The code which identifies the currency under which the hierarchy value operates. This value must be predefined on the CURRENCIES table.

Sample Input Message

```
{
  "hierarchyValue": 6,
  "hierarchyDescription": "6 South East",
  "hierarchyLevel": "AR",
  "parentHierarchyId": 3,
  "managerName": "McMillan",
```

```
"currencyCode": "USD"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Organizational Hierarchy

Functional Area

Organizational Hierarchy

Business Overview

If Merchandising is not the system of record for organizational hierarchy information for an implementation, then this service may be used to delete an existing hierarchy in Merchandising based on the request from an external system.

The following organizational hierarchy elements can be deleted using this service: chain, area, region, or district. The organizational hierarchy must be deleted from the lowest level up.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/delete

Input Payload Details

Table 5-1093 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The ID of the specified hierarchy level. This value must be unique among other values of the same hierarchy level.
hierarchyLevel	Yes	String (2)	The hierarchy level of the hierarchy value. This field will indicate the level of the organizational hierarchy to which the cost change applies. Valid values are CH (chain), AR (area), RE (region), DI (district).

Sample Input Message

```
{
  "hierarchyValue": 4,
  "hierarchyLevel": "CH"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Organizational Hierarchy

Functional Area

Organizational Hierarchy

Business Overview

If Merchandising is not the system of record for organizational hierarchy information for an implementation, then this service may be used to update an existing hierarchy based on the request from an external system.

The following organizational hierarchy elements can be modified using this service: chain, area, region, or district.

Service Type

PUT

ReST URL

MerchIntegrations/services/organizationalHierarchy/update

Input Payload Details

Table 5-1094 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The ID of the specified hierarchy level. This value must be unique among other values of the same hierarchy level.
hierarchyDescription	Yes	String (120)	The description of the hierarchy value.
hierarchyLevel	Yes	String (2)	The hierarchy level of the hierarchy value. This field will indicate the level of the organizational hierarchy to which the cost change applies. Valid values are CH (chain), AR (area), RE (region), DI (district).
parentHierarchyId	No	Number (10)	The ID of the hierarchy value's parent hierarchy value. This value must be predefined on either the CHAIN, AREA or REGION table.
managerName	No	String (120)	The manager name of the hierarchy value.
currencyCode	No	String (3)	The code which identifies the currency under which the hierarchy value operates. This value must be predefined on the CURRENCIES table.

Sample Input Message

```
{
  "hierarchyValue": 4,
  "hierarchyDescription": "Chain 4 North America",
  "hierarchyLevel": "CH",
  "parentHierarchyId": 117516,
  "managerName": "McCarthy",
  "currencyCode": "USD"
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Stores

The following services are included in this functional area:

- [Store Download Service](#)
 - [Get Basic Store Details](#)
 - [Get Store Details](#)
 - [Get Store Details for Provided Store](#)
- [Store Upload Service](#)
 - [Create Store](#)
 - [Create Store Address](#)

- [Create Store Department Up Charges](#)
- [Create Store Hours](#)
- [Create Store Location Trait](#)
- [Create Walk-Through Store](#)
- [Delete Store](#)
- [Delete Store Address](#)
- [Delete Store Department Up Charges](#)
- [Delete Store Hours](#)
- [Delete Store Location Trait](#)
- [Delete Walk-Through Store](#)
- [Update Store](#)
- [Update Store Address](#)
- [Update Store Department Up Charges](#)
- [Update Store Hours](#)
- [Stores by Hierarchy Level Upload Service](#)
 - [Create Store Location Trait by Hierarchy Level](#)
 - [Delete Store Location Trait by Hierarchy Level](#)

Store Download Service

The following services are included in this functional area:

- [Get Basic Store Details](#)
- [Get Store Details](#)
- [Get Store Details for Provided Store](#)

Get Basic Store Details

This section describes the Store Detail service.

Business Overview

Store Detail service allows user to retrieve Store information for a selected store or for all stores.

Service Type

Get

ReST URL

Store/storeDetail?store={storeNumber}

Input Parameters

Parameter Name	Required	Description
Store	No	Store ID. If none is specified, all stores will be retrieved.

Output

RestStoreRecRDO

Parameter Name	Data Type
store	BigDecimal
storeName	String
storeName10	String
storeName3	String
storeNameSecondary	String
storeClass	String
storeOpenDate	Timestamp
storeCloseDate	Timestamp
acquiredDate	Timestamp
remodelDate	Timestamp
vatRegion	BigDecimal
vatIncludeInd	String
stockholdingInd	String
channelId	BigDecimal
transferZone	BigDecimal
defaultWh	BigDecimal
stopOrderDays	BigDecimal
startOrderDays	BigDecimal
currencyCode	String
lang	BigDecimal
dunsNumber	String
dunsLoc	String
sisterStore	BigDecimal
tsfEntityId	BigDecimal
orgUnitId	BigDecimal
storeType	String
wfCustomerId	BigDecimal
timezoneName	String
customerOrderLocInd	String
company	BigDecimal
chain	BigDecimal
area	BigDecimal
region	BigDecimal
district	BigDecimal
add1	String
add2	String
add3	String
city	String
state	String

Parameter Name	Data Type
countryId	String
post	String
contactName	String
contactPhone	String
contactEmail	String

JSON Structure

```
{
  "store": null,
  "storeName": null,
  "storeName10": null,
  "storeName3": null,
  "storeNameSecondary": null,
  "storeClass": null,
  "storeOpenDate": null,
  "storeCloseDate": null,
  "acquiredDate": null,
  "remodelDate": null,
  "vatRegion": null,
  "vatIncludeInd": null,
  "stockholdingInd": null,
  "channelId": null,
  "transferZone": null,
  "defaultWh": null,
  "stopOrderDays": null,
  "startOrderDays": null,
  "currencyCode": null,
  "lang": null,
  "dunsNumber": null,
  "dunsLoc": null,
  "sisterStore": null,
  "tsfEntityId": null,
  "orgUnitId": null,
  "storeType": null,
  "wfCustomerId": null,
  "timezoneName": null,
  "customerOrderLocInd": null,
  "company": null,
  "chain": null,
  "area": null,
  "region": null,
  "district": null,
  "add1": null,
  "add2": null,
  "add3": null,
  "city": null,
  "state": null,
  "countryId": null,
  "post": null,
  "contactName": null,
```

```

    "contactPhone": null,
    "contactEmail": null,
    "links": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_STORE	Yes	No	No	No
STORE_HIERARCHY	Yes	No	No	No
ADDR	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Store Details

Endpoint

MerchIntegrations/services/foundation/store

MerchIntegrations/services/foundation/store/{storeId}

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service publishes store master data to downstream consuming systems. It publishes the store definition together with primary and additional addresses, store hours, store services, store conditions, and related descriptive attributes used to interpret the store record. There is no explicit filter in V_MERCHAPI_EXT_STORE_JSON, so all STORE rows are eligible for publication.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: STORE, ADDR

Additional business tables: ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, COUNTRY, STORE_HOURS, STORE_SERVICE, STORE_CONDITION, CHANNELS, STORE_FORMAT, DISTRICT, TSFZONE, WH, LANG, CODE_DETAIL, STORE_CFA_EXT, ADDR_CFA_EXT

JSON cache table: MERCHAPI_EXT_STORE

JSON generation view: V_MERCHAPI_EXT_STORE_JSON

Builds the store header from STORE, derives the primary address from ADDR joined to ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, and COUNTRY, aggregates all store addresses from ADDR, aggregates store hours from STORE_HOURS, store services from STORE_SERVICE, store conditions from STORE_CONDITION, and enriches descriptive values from CHANNELS, STORE_FORMAT, DISTRICT, TSFZONE, WH, LANG, and CODE_DETAIL.

Store- and address-level custom flex attributes are added from MERCHAPI_EXT_STORE_CFA and MERCHAPI_EXT_STORE_ADDR_CFA.

There is no explicit filter in V_MERCHAPI_EXT_STORE_JSON, so functionally all rows from STORE are eligible for publication.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_STORE_JSON and apply no additional filter criteria; all STORE rows are eligible for publication.

The current REBUILD branch merges changed and new rows only; it does not include a delete-marking pass for stores that disappear from V_MERCHAPI_EXT_STORE_JSON.

ICL (Integration Change Log) table: ICL_RMS_STORE and ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on STORE, and address changes are captured through ICL_RMS_ADDR for ADDR rows on modules ST and WFST.

ICL_RMS_ADDR rows are filtered to MODULE IN ('ST','WFST') and republish the parent store payload.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_STORE

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_STORE_ADHOC_PROCESS ->
MERCHAPI_EXT_STORE_ADHOC_JOB

Webhook configuration api name: foundation/store

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/store

MerchIntegrations/services/foundation/store/{storeId}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)

Parameter Name	Required	Data Type	Description
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Store)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1095 Input Parameter for "MerchIntegrations/services/foundation/store/{storeId}"

Parameter Name	Required	Data Type	Description
storeId	Yes	Number	Store Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-1096 PageResultsItemsMerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-1096 (Cont.) PageResultsItemsMerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1097 MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for store details.
store	No	Number(10,0)	This field contains the number that uniquely identifies the store.
storeType	No	String (6)	This field indicates whether a particular store is a franchise or company store.
storeName	No	String (150)	This field contains the name of the store which, along with the store number, identifies the store.
storeName10	No	String (10)	This field contains a ten-character abbreviation of the store name.
storeName3	No	String (3)	This field contains a three-character abbreviation of the store name.
storeClass	No	String (1)	This field contains the code letter indicating the class of which the store is a member.
storeClassDescription	No	String (250)	This field contains the description indicating the class of which the store is a member.
managerName	No	String (120)	This field contains the name of the store manager.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
openDate	No	date	This field contains the date on which the store opened.
closeDate	No	date	This field contains the date on which the store closed.
acquireDate	No	date	This field contains the date on which the store was acquired.
remodelDate	No	date	This field contains the date on which the store was last remodeled.
faxNumber	No	String (20)	This field contains the fax number for the store.
phoneNumber	No	String (20)	This field contains the phone number for the store.
email	No	String (100)	This field contains the email address for the location.
totalSquareFeet	No	Number(8,0)	This field contains the total square footage of the store.
sellingSquareFeet	No	Number(8,0)	This field contains the total square footage of the store's selling area.
linearDistance	No	Number(8,0)	This field contains the total merchandisable space of the location.
stockholdingInd	No	String (1)	This field indicates whether the store can hold stock.
channelId	No	Number(4,0)	This field contains the channel with which the store is associated.
channelName	No	String (120)	This field contains the channel description with which the store is associated.
storeFormat	No	Number(4,0)	This field contains the number indicating the format of the store. Valid values are found on the store format table.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeFormatName	No	String (60)	This field contains the description for the format of the store.
mallName	No	String (120)	This field contains the name of the mall in which the store is located.
district	No	Number(10,0)	This field contains the number of the district in which the store is a member.
districtName	No	String (120)	This field contains the name of the district in which the store is a member.
transferZone	No	Number(4,0)	This field contains the transfer zone ID in which the store is located.
transferZoneDescription	No	String (120)	This field contains the transfer zone description in which the store is located.
defaultWarehouse	No	Number(10,0)	This field contains the number of the warehouse that is the primary sourcing warehouse for the store.
stopOrderdays	No	Number(3,0)	This field contains the number of days before a store closing that the store will stop accepting orders. This column will be used when the store close date is defined.
startOrderDays	No	Number(3,0)	This field contains the number of days before the store open date that the store will begin accepting orders.
currencyCode	No	String (3)	This field contains the currency code under which the store operates.
language	No	String (6)	This field contains the ISO code of the language to be used for the given store.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
integratedPosInd	No	String (1)	This field indicates whether Sales Audit should expect files from this store for processing.
dunsNumber	No	String (9)	This field contains the Dun and Bradstreet number to identify the store.
dunsLocation	No	String (4)	This field contains the Dun and Bradstreet number to identify the location.
pricingStore	No	Number(10,0)	This field contains the location from which pricing records will be copied for a new store or will be used to determine the price for this store if not otherwise defined for an item.
timezoneName	No	String (64)	This field contains the text value of the time zone of the store.
pricingStoreCurrency	No	String (3)	This field contains the currency code of the location from which pricing records will be copied.
orgUnitId	No	Number(15,0)	This field contains the organizational unit ID that this store is associated with.
storeNameSecondary	No	String (150)	This field contains the secondary name of the store.
vatRegion	No	Number(4,0)	This field contains the ID of the tax region the store is associated with.
vatIncludeInd	No	String (1)	This field indicates whether tax will be included in the retail prices for the store.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sisterStore	No	Number(10,0)	This field contains the store number that will be used to relate the current store to the historical data of an existing store. This is used by Allocation.
trasferEntityId	No	Number(10,0)	This field contains the transfer entity ID with which the store is associated.
autoReceiveInd	No	String (1)	This field indicates whether receipts for the store will be made automatically by Merchandising.
remerchandisingInd	No	String (1)	This field is currently not used.
franchiseCustomerId	No	Number(10,0)	This field indicates which franchise customer this store is associated with.
customerOrderLocationInd	No	String (1)	This field indicates whether the location is a customer order location. If the indicator is Y, then the location can be used for sourcing and fulfillment orders; else it cannot be used and inventory information will not be shared with OMS.
giftWrappingInd	No	String (1)	This field indicates if the location will support gift wrapping.
customerOrderShippingInd	No	String (1)	This field indicates if the location will support shipping to a customer for customer orders.
onlineStoreInd	No	String (1)	This field indicates that the store is an online store.
taxId	No	String (20)	This field contains the unique tax identification number of the store.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressAdd1	No	String (240)	This field contains the first line of the address.
primaryAddressAdd2	No	String (240)	This field contains the second line of the address.
primaryAddressAdd3	No	String (240)	This field contains the third line of the address.
primaryAddressCity	No	String (120)	This field contains the name of the city that is associated with the address.
primaryAddressState	No	String (3)	This field contains the state abbreviation for the address.
primaryAddressStateName	No	String (120)	This field contains the state description for the address.
primaryAddressCountryId	No	String (3)	This field contains the country where the address exists.
primaryAddressCountryName	No	String (120)	This field contains the country description where the address exists.
primaryAddressPost	No	String (30)	This field contains the zip code for the address.
primaryAddressContactName	No	String (120)	This field contains the name of the contact person at this address.
primaryAddressContactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
primaryAddressContactTelex	No	String (20)	This field contains the telex number of the contact person at this address.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressContactFax	No	String (20)	This field contains the fax number of the contact person at this address.
primaryAddressContactEmail	No	String (100)	This field contains the email address of the contact person at this address.
primaryAddressPhone2	No	String (20)	This field contains an additional phone number associated with the primary address.
primaryAddressPhone2Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressPhone3	No	String (20)	This field contains a second additional phone number associated with the primary address.
primaryAddressPhone3Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressLatitude	No	String (30)	This field contains the north-south geographic coordinate of the primary address, represented in decimal degrees.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressLongitude	No	String (30)	This field contains the east-west geographic coordinate of the primary address, represented in decimal degrees.
primaryAddressOracleVendorSiteId	No	Number(15,0)	This field contains the Oracle vendor site ID.
primaryAddressCounty	No	String (250)	This field contains the county where the address exists.
primaryAddressJurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the Store.
address	No	Collection of Object	This List holds all the Store addresses.
storeHours	No	Collection of Object	This List holds all the Store Hours attributes.
storeServices	No	Collection of Object	This List holds all the Store Services attributes. Each StoreService element contains the code representing the store service provided at the store and its creation datetime.
storeConditions	No	Collection of Object	This List holds all the Store Conditions attributes.
storeTranslations	No	Collection of Object	This List holds all the store translations, providing language-specific store names and details.

Table 5-1097 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cacheTimestamp	No	dateTime	This field specifies the date and time when the store record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1098 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1099 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressKey	No	Number(11,0)	This field contains the unique address key.
addressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.

Table 5-1099 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressInd	No	String (1)	This field indicates whether the address is the primary address for the address type.
add1	No	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	No	String (120)	This field contains the name of the city that is associated with the address.
state	No	String (3)	This field contains the state abbreviation that is associated with the address.
stateName	No	String (120)	This field contains the state description that is associated with the address.
countryId	No	String (3)	This field contains the country where the address exists.
countryName	No	String (120)	This field contains the country description that is associated with the address.
post	No	String (30)	This field contains the zip code for the address.
contactName	No	String (120)	This field contains the name of the contact person at this address.
contactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
contactTelex	No	String (20)	This field contains the telex number of the contact person at this address.

Table 5-1099 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactFax	No	String (20)	This field contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This field contains the email address of the contact person at this address.
phone2Type	No	String (6)	This field contains the type of the phone number stored in the phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This field contains an additional phone number associated with the address.
phone3Type	No	String (6)	This field contains the type of the phone number stored in the phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This field contains a second additional phone number associated with the address.
latitude	No	String (30)	This field contains the north-south geographic coordinate of the address, represented in decimal degrees.

Table 5-1099 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
longitude	No	String (30)	This field contains the east-west geographic coordinate of the address, represented in decimal degrees.
county	No	String (250)	This field contains the county where the address exists.
jurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse's address.

Table 5-1100 StoreHours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	No	Number(1,0)	This field contains the numeric value that indicates the day that the open and close times pertain to. Valid values are 1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, or 7-Saturday.
storeOpenTime	No	String (8)	This field contains the open time for the store for each day of the week. The time format will be in HH:MM AM or HH:MM PM.
storeCloseTime	No	String (8)	This field contains the close time for the store for each day of the week. The time format will be in HH:MM AM or HH:MM PM.

Table 5-1100 (Cont.) StoreHours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).

Table 5-1101 StoreService - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeService	No	String (6)	This field contains the code representing the store service provided at the store.
createDateTime	No	dateTime	This field holds the record creation date and time of the store service (UTC).

Table 5-1102 StoreConditions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
conditionType	No	String (6)	This field holds the condition type configured under code type SCND with the used indicator set to Y.
startDate	No	date	This field holds the start date of the condition.
endDate	No	date	This field holds the end date of the condition.
comments	No	String (250)	This field holds the extra information about the condition setup.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).

Table 5-1103 StoreTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	String	The short code for the language (e.g. EN, FR).
languageName	No	String	The name of the language (e.g. English, French).
isoCode	No	String	The ISO code for the language (e.g. en_US, fr_FR).
storeName	No	String	The store name in this language.
storeNameSecondary	No	String	The secondary store name in this language.
createDateTime	No	String	Record creation datetime for this translation.
updateDateTime	No	String	Datetime this translation was last updated.

Table 5-1104 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "store": 6000,
      "storeType": null,
      "storeName": "storeName",
      "storeName10": null,
      "storeName3": null,
      "storeClass": null,
      "storeClassDescription": null,
      "managerName": "storeMgrName",
      "openDate": "2001-12-31",
      "closeDate": "2001-12-31",

```

```
"acquireDate": "2001-12-31",
"remodelDate": "2001-12-31",
"faxNumber": null,
"phoneNumber": "1800800800",
"email": "someone@email.com",
"totalSquareFeet": 1,
"sellingSquareFeet": 1,
"linearDistance": 1,
"stockholdingInd": "Y",
"channelId": 1,
"channelName": null,
"storeFormat": 1,
"storeFormatName": null,
"mallName": "mallName",
"district": 1,
"districtName": null,
"transferZone": 1,
"transferZoneDescription": null,
"defaultWarehouse": 1,
"stopOrderdays": 1,
"startOrderDays": 1,
"currencyCode": "USD",
"language": null,
"integratedPosInd": "Y",
"dunsNumber": null,
"dunsLocation": null,
"pricingStore": 1,
"timezoneName": "timezoneName",
"pricingStoreCurrency": null,
"orgUnitId": 1,
"storeNameSecondary": "storeNameSecondary",
"vatRegion": 1,
"vatIncludeInd": "Y",
"sisterStore": 1,
"transferEntityId": 1,
"autoReceiveInd": null,
"remerchandisingInd": null,
"franchiseCustomerId": 1,
"customerOrderLocationInd": "Y",
"giftWrappingInd": "Y",
"customerOrderShippingInd": "Y",
"onlineStoreInd": "Y",
"taxId": null,
"primaryAddressType": "01",
"primaryAddressTypeDescription": null,
"primaryAddressAdd1": "101 First Street",
"primaryAddressAdd2": "201 Second Street",
"primaryAddressAdd3": null,
"primaryAddressCity": "Minneapolis",
"primaryAddressState": "MN",
"primaryAddressStateName": "stateName",
"primaryAddressCountryId": "US",
"primaryAddressCountryName": "countryName",
"primaryAddressPost": "55555-1234",
"primaryAddressContactName": "contactName",
"primaryAddressContactPhone": "1800800800",
```



```

"primaryAddressContactTelex": null,
"primaryAddressContactFax": "617-897-0902",
"primaryAddressContactEmail": "someone@email.com",
"primaryAddressPhone2": "5551234567",
"primaryAddressPhone2Type": "W0",
"primaryAddressPhone3": "5551234567",
"primaryAddressPhone3Type": "W0",
"primaryAddressLatitude": "44.38129",
"primaryAddressLongitude": "-92.03521",
"primaryAddressOracleVendorSiteId": 1,
"primaryAddressCounty": "Some County",
"primaryAddressJurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"address": [
{
"addressKey": 114772,
"addressType": "01",
"addressTypeDescription": null,
"primaryAddressInd": "Y",
"add1": "101 First Street",
"add2": "201 Second Street",
"add3": null,
"city": "Minneapolis",
"state": "MN",
"stateName": "MN",
"countryId": "US",
"countryName": "US",
"post": "55555-1234",
"contactName": "Laura Johnson",
"contactPhone": "1800800800",
"contactTelex": null,
"contactFax": "617-897-0902",
"contactEmail": "someone@email.com",
"phone2Type": "W0",
"phone2": "5551234567",
"phone3Type": "W0",
"phone3": "5551234567",
"latitude": "44.38129",
"longitude": "-92.03521",
"county": null,
"jurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",

```

```

"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeHours": [
{
"dayNo": 1,
"storeOpenTime": null,
"storeCloseTime": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeServices": [
{
"storeService": null,
"createDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeConditions": [
{
"conditionType": null,
"startDate": "2001-12-31",
"endDate": "2001-12-31",
"comments": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeTranslations": [
{
"languageCode": "String",
"languageName": "String",
"isoCode": "String",
"storeName": "String",
"storeNameSecondary": "String",
"createDateTime": "String",
"updateDateTime": "String"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",

```

```

"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	Yes	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_STORE	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes
ADDR	Yes	No	No	No
STORE_CFA_EXT	Yes	No	No	No
ADDR_CFA_EXT CFA	Yes	No	No	No
MERCHAPI_EXT_STORE_CFA	No	Yes	No	Yes
MERCHAPI_EXT_STORE_ADDR_CFA	No	Yes	No	Yes
MERCHAPI_EXT_STORE	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_STORE_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

Get Store Details for Provided Store

Endpoint

MerchIntegrations/services/foundation/store

MerchIntegrations/services/foundation/store/{storeId}

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service publishes store master data to downstream consuming systems. It publishes the store definition together with primary and additional addresses, store hours, store services, store conditions, and related descriptive attributes used to interpret the store record. There is no explicit filter in V_MERCHAPI_EXT_STORE_JSON, so all STORE rows are eligible for publication.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: STORE, ADDR

Additional business tables: ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, COUNTRY, STORE_HOURS, STORE_SERVICE, STORE_CONDITION, CHANNELS, STORE_FORMAT, DISTRICT, TSFZONE, WH, LANG, CODE_DETAIL, STORE_CFA_EXT, ADDR_CFA_EXT

JSON cache table: MERCHAPI_EXT_STORE

JSON generation view: V_MERCHAPI_EXT_STORE_JSON

Builds the store header from STORE, derives the primary address from ADDR joined to ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, and COUNTRY, aggregates all store addresses from ADDR, aggregates store hours from STORE_HOURS, store services from STORE_SERVICE, store conditions from STORE_CONDITION, and enriches descriptive values from CHANNELS, STORE_FORMAT, DISTRICT, TSFZONE, WH, LANG, and CODE_DETAIL.

Store- and address-level custom flex attributes are added from MERCHAPI_EXT_STORE_CFA and MERCHAPI_EXT_STORE_ADDR_CFA.

There is no explicit filter in V_MERCHAPI_EXT_STORE_JSON, so functionally all rows from STORE are eligible for publication.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_STORE_JSON and apply no additional filter criteria; all STORE rows are eligible for publication.

The current REBUILD branch merges changed and new rows only; it does not include a delete-marking pass for stores that disappear from V_MERCHAPI_EXT_STORE_JSON.

ICL (Integration Change Log) table: ICL_RMS_STORE and ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on STORE, and address changes are captured through ICL_RMS_ADDR for ADDR rows on modules ST and WFST.

ICL_RMS_ADDR rows are filtered to MODULE IN ('ST','WFST') and republish the parent store payload.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_STORE

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_STORE_ADHOC_PROCESS ->
MERCHAPI_EXT_STORE_ADHOC_JOB

Webhook configuration api name: foundation/store

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/store

MerchIntegrations/services/foundation/store/{storeId}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Store)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1105 Input Parameter for "MerchIntegrations/services/foundation/store/{storeId}"

Parameter Name	Required	Data Type	Description
storeId	Yes	Number	Store Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-1106 PageResultsItemsMerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-1106 (Cont.) PageResultsItemsMerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1107 MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for store details.
store	No	Number(10,0)	This field contains the number that uniquely identifies the store.
storeType	No	String (6)	This field indicates whether a particular store is a franchise or company store.
storeName	No	String (150)	This field contains the name of the store which, along with the store number, identifies the store.
storeName10	No	String (10)	This field contains a ten-character abbreviation of the store name.
storeName3	No	String (3)	This field contains a three-character abbreviation of the store name.
storeClass	No	String (1)	This field contains the code letter indicating the class of which the store is a member.
storeClassDescription	No	String (250)	This field contains the description indicating the class of which the store is a member.
managerName	No	String (120)	This field contains the name of the store manager.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
openDate	No	date	This field contains the date on which the store opened.
closeDate	No	date	This field contains the date on which the store closed.
acquireDate	No	date	This field contains the date on which the store was acquired.
remodelDate	No	date	This field contains the date on which the store was last remodeled.
faxNumber	No	String (20)	This field contains the fax number for the store.
phoneNumber	No	String (20)	This field contains the phone number for the store.
email	No	String (100)	This field contains the email address for the location.
totalSquareFeet	No	Number(8,0)	This field contains the total square footage of the store.
sellingSquareFeet	No	Number(8,0)	This field contains the total square footage of the store's selling area.
linearDistance	No	Number(8,0)	This field contains the total merchandisable space of the location.
stockholdingInd	No	String (1)	This field indicates whether the store can hold stock.
channelId	No	Number(4,0)	This field contains the channel with which the store is associated.
channelName	No	String (120)	This field contains the channel description with which the store is associated.
storeFormat	No	Number(4,0)	This field contains the number indicating the format of the store. Valid values are found on the store format table.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeFormatName	No	String (60)	This field contains the description for the format of the store.
mallName	No	String (120)	This field contains the name of the mall in which the store is located.
district	No	Number(10,0)	This field contains the number of the district in which the store is a member.
districtName	No	String (120)	This field contains the name of the district in which the store is a member.
transferZone	No	Number(4,0)	This field contains the transfer zone ID in which the store is located.
transferZoneDescription	No	String (120)	This field contains the transfer zone description in which the store is located.
defaultWarehouse	No	Number(10,0)	This field contains the number of the warehouse that is the primary sourcing warehouse for the store.
stopOrderdays	No	Number(3,0)	This field contains the number of days before a store closing that the store will stop accepting orders. This column will be used when the store close date is defined.
startOrderDays	No	Number(3,0)	This field contains the number of days before the store open date that the store will begin accepting orders.
currencyCode	No	String (3)	This field contains the currency code under which the store operates.
language	No	String (6)	This field contains the ISO code of the language to be used for the given store.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
integratedPosInd	No	String (1)	This field indicates whether Sales Audit should expect files from this store for processing.
dunsNumber	No	String (9)	This field contains the Dun and Bradstreet number to identify the store.
dunsLocation	No	String (4)	This field contains the Dun and Bradstreet number to identify the location.
pricingStore	No	Number(10,0)	This field contains the location from which pricing records will be copied for a new store or will be used to determine the price for this store if not otherwise defined for an item.
timezoneName	No	String (64)	This field contains the text value of the time zone of the store.
pricingStoreCurrency	No	String (3)	This field contains the currency code of the location from which pricing records will be copied.
orgUnitId	No	Number(15,0)	This field contains the organizational unit ID that this store is associated with.
storeNameSecondary	No	String (150)	This field contains the secondary name of the store.
vatRegion	No	Number(4,0)	This field contains the ID of the tax region the store is associated with.
vatIncludeInd	No	String (1)	This field indicates whether tax will be included in the retail prices for the store.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sisterStore	No	Number(10,0)	This field contains the store number that will be used to relate the current store to the historical data of an existing store. This is used by Allocation.
trasferEntityId	No	Number(10,0)	This field contains the transfer entity ID with which the store is associated.
autoReceiveInd	No	String (1)	This field indicates whether receipts for the store will be made automatically by Merchandising.
remerchandisingInd	No	String (1)	This field is currently not used.
franchiseCustomerId	No	Number(10,0)	This field indicates which franchise customer this store is associated with.
customerOrderLocationInd	No	String (1)	This field indicates whether the location is a customer order location. If the indicator is Y, then the location can be used for sourcing and fulfillment orders; else it cannot be used and inventory information will not be shared with OMS.
giftWrappingInd	No	String (1)	This field indicates if the location will support gift wrapping.
customerOrderShippingInd	No	String (1)	This field indicates if the location will support shipping to a customer for customer orders.
onlineStoreInd	No	String (1)	This field indicates that the store is an online store.
taxId	No	String (20)	This field contains the unique tax identification number of the store.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressAdd1	No	String (240)	This field contains the first line of the address.
primaryAddressAdd2	No	String (240)	This field contains the second line of the address.
primaryAddressAdd3	No	String (240)	This field contains the third line of the address.
primaryAddressCity	No	String (120)	This field contains the name of the city that is associated with the address.
primaryAddressState	No	String (3)	This field contains the state abbreviation for the address.
primaryAddressStateName	No	String (120)	This field contains the state description for the address.
primaryAddressCountryId	No	String (3)	This field contains the country where the address exists.
primaryAddressCountryName	No	String (120)	This field contains the country description where the address exists.
primaryAddressPost	No	String (30)	This field contains the zip code for the address.
primaryAddressContactName	No	String (120)	This field contains the name of the contact person at this address.
primaryAddressContactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
primaryAddressContactTelex	No	String (20)	This field contains the telex number of the contact person at this address.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressContactFax	No	String (20)	This field contains the fax number of the contact person at this address.
primaryAddressContactEmail	No	String (100)	This field contains the email address of the contact person at this address.
primaryAddressPhone2	No	String (20)	This field contains an additional phone number associated with the primary address.
primaryAddressPhone2Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressPhone3	No	String (20)	This field contains a second additional phone number associated with the primary address.
primaryAddressPhone3Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressLatitude	No	String (30)	This field contains the north-south geographic coordinate of the primary address, represented in decimal degrees.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressLongitude	No	String (30)	This field contains the east-west geographic coordinate of the primary address, represented in decimal degrees.
primaryAddressOracleVendorSiteId	No	Number(15,0)	This field contains the Oracle vendor site ID.
primaryAddressCounty	No	String (250)	This field contains the county where the address exists.
primaryAddressJurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the Store.
address	No	Collection of Object	This List holds all the Store addresses.
storeHours	No	Collection of Object	This List holds all the Store Hours attributes.
storeServices	No	Collection of Object	This List holds all the Store Services attributes. Each StoreService element contains the code representing the store service provided at the store and its creation datetime.
storeConditions	No	Collection of Object	This List holds all the Store Conditions attributes.
storeTranslations	No	Collection of Object	This List holds all the store translations, providing language-specific store names and details.

Table 5-1107 (Cont.) MerchApiStore - Object. See list of elements for detail

Element Name	Required	Data Type	Description
cacheTimestamp	No	dateTime	This field specifies the date and time when the store record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1108 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1109 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressKey	No	Number(11,0)	This field contains the unique address key.
addressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.

Table 5-1109 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressInd	No	String (1)	This field indicates whether the address is the primary address for the address type.
add1	No	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	No	String (120)	This field contains the name of the city that is associated with the address.
state	No	String (3)	This field contains the state abbreviation that is associated with the address.
stateName	No	String (120)	This field contains the state description that is associated with the address.
countryId	No	String (3)	This field contains the country where the address exists.
countryName	No	String (120)	This field contains the country description that is associated with the address.
post	No	String (30)	This field contains the zip code for the address.
contactName	No	String (120)	This field contains the name of the contact person at this address.
contactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
contactTelex	No	String (20)	This field contains the telex number of the contact person at this address.

Table 5-1109 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactFax	No	String (20)	This field contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This field contains the email address of the contact person at this address.
phone2Type	No	String (6)	This field contains the type of the phone number stored in the phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This field contains an additional phone number associated with the address.
phone3Type	No	String (6)	This field contains the type of the phone number stored in the phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This field contains a second additional phone number associated with the address.
latitude	No	String (30)	This field contains the north-south geographic coordinate of the address, represented in decimal degrees.

Table 5-1109 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
longitude	No	String (30)	This field contains the east-west geographic coordinate of the address, represented in decimal degrees.
county	No	String (250)	This field contains the county where the address exists.
jurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse's address.

Table 5-1110 StoreHours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	No	Number(1,0)	This field contains the numeric value that indicates the day that the open and close times pertain to. Valid values are 1-Sunday, 2-Monday, 3-Tuesday, 4-Wednesday, 5-Thursday, 6-Friday, or 7-Saturday.
storeOpenTime	No	String (8)	This field contains the open time for the store for each day of the week. The time format will be in HH:MM AM or HH:MM PM.
storeCloseTime	No	String (8)	This field contains the close time for the store for each day of the week. The time format will be in HH:MM AM or HH:MM PM.

Table 5-1110 (Cont.) StoreHours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).

Table 5-1111 StoreService - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeService	No	String (6)	This field contains the code representing the store service provided at the store.
createDateTime	No	dateTime	This field holds the record creation date and time of the store service (UTC).

Table 5-1112 StoreConditions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
conditionType	No	String (6)	This field holds the condition type configured under code type SCND with the used indicator set to Y.
startDate	No	date	This field holds the start date of the condition.
endDate	No	date	This field holds the end date of the condition.
comments	No	String (250)	This field holds the extra information about the condition setup.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).

Table 5-1113 StoreTranslation - Object. See list of elements for detail

Element Name	Required	Data Type	Description
languageCode	No	String	The short code for the language (e.g. EN, FR).
languageName	No	String	The name of the language (e.g. English, French).
isoCode	No	String	The ISO code for the language (e.g. en_US, fr_FR).
storeName	No	String	The store name in this language.
storeNameSecondary	No	String	The secondary store name in this language.
createDateTime	No	String	Record creation datetime for this translation.
updateDateTime	No	String	Datetime this translation was last updated.

Table 5-1114 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "store": 6000,
      "storeType": null,
      "storeName": "storeName",
      "storeName10": null,
      "storeName3": null,
      "storeClass": null,
      "storeClassDescription": null,
      "managerName": "storeMgrName",
      "openDate": "2001-12-31",
      "closeDate": "2001-12-31",
    }
  ]
}
```

```
"acquireDate": "2001-12-31",
"remodelDate": "2001-12-31",
"faxNumber": null,
"phoneNumber": "1800800800",
"email": "someone@email.com",
"totalSquareFeet": 1,
"sellingSquareFeet": 1,
"linearDistance": 1,
"stockholdingInd": "Y",
"channelId": 1,
"channelName": null,
"storeFormat": 1,
"storeFormatName": null,
"mallName": "mallName",
"district": 1,
"districtName": null,
"transferZone": 1,
"transferZoneDescription": null,
"defaultWarehouse": 1,
"stopOrderdays": 1,
"startOrderDays": 1,
"currencyCode": "USD",
"language": null,
"integratedPosInd": "Y",
"dunsNumber": null,
"dunsLocation": null,
"pricingStore": 1,
"timezoneName": "timezoneName",
"pricingStoreCurrency": null,
"orgUnitId": 1,
"storeNameSecondary": "storeNameSecondary",
"vatRegion": 1,
"vatIncludeInd": "Y",
"sisterStore": 1,
"transferEntityId": 1,
"autoReceiveInd": null,
"remerchandisingInd": null,
"franchiseCustomerId": 1,
"customerOrderLocationInd": "Y",
"giftWrappingInd": "Y",
"customerOrderShippingInd": "Y",
"onlineStoreInd": "Y",
"taxId": null,
"primaryAddressType": "01",
"primaryAddressTypeDescription": null,
"primaryAddressAdd1": "101 First Street",
"primaryAddressAdd2": "201 Second Street",
"primaryAddressAdd3": null,
"primaryAddressCity": "Minneapolis",
"primaryAddressState": "MN",
"primaryAddressStateName": "stateName",
"primaryAddressCountryId": "US",
"primaryAddressCountryName": "countryName",
"primaryAddressPost": "55555-1234",
"primaryAddressContactName": "contactName",
"primaryAddressContactPhone": "1800800800",
```

```

"primaryAddressContactTelex": null,
"primaryAddressContactFax": "617-897-0902",
"primaryAddressContactEmail": "someone@email.com",
"primaryAddressPhone2": "5551234567",
"primaryAddressPhone2Type": "W0",
"primaryAddressPhone3": "5551234567",
"primaryAddressPhone3Type": "W0",
"primaryAddressLatitude": "44.38129",
"primaryAddressLongitude": "-92.03521",
"primaryAddressOracleVendorSiteId": 1,
"primaryAddressCounty": "Some County",
"primaryAddressJurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"address": [
{
"addressKey": 114772,
"addressType": "01",
"addressTypeDescription": null,
"primaryAddressInd": "Y",
"add1": "101 First Street",
"add2": "201 Second Street",
"add3": null,
"city": "Minneapolis",
"state": "MN",
"stateName": "MN",
"countryId": "US",
"countryName": "US",
"post": "55555-1234",
"contactName": "Laura Johnson",
"contactPhone": "1800800800",
"contactTelex": null,
"contactFax": "617-897-0902",
"contactEmail": "someone@email.com",
"phone2Type": "W0",
"phone2": "5551234567",
"phone3Type": "W0",
"phone3": "5551234567",
"latitude": "44.38129",
"longitude": "-92.03521",
"county": null,
"jurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",

```

```

"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
},
"storeHours": [
{
"dayNo": 1,
"storeOpenTime": null,
"storeCloseTime": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeServices": [
{
"storeService": null,
"createDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeConditions": [
{
"conditionType": null,
"startDate": "2001-12-31",
"endDate": "2001-12-31",
"comments": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"storeTranslations": [
{
"languageCode": "String",
"languageName": "String",
"isoCode": "String",
"storeName": "String",
"storeNameSecondary": "String",
"createDateTime": "String",
"updateDateTime": "String"
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",

```

```
"rel": "next"
}
]
}
```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	Yes	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ICL_RMS_STORE	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes
ADDR	Yes	No	No	No
STORE_CFA_EXT	Yes	No	No	No
ADDR_CFA_EXT CFA	Yes	No	No	No
MERCHAPI_EXT_STORE_CFA	No	Yes	No	Yes
MERCHAPI_EXT_STORE_ADDR_CFA	No	Yes	No	Yes
MERCHAPI_EXT_STORE	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_STORE_JSON	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

Store Upload Service

The following services are included in this functional area:

- [Create Store](#)
- [Create Store Address](#)
- [Create Store Department Up Charges](#)
- [Create Store Hours](#)
- [Create Store Location Trait](#)
- [Create Walk-Through Store](#)
- [Delete Store](#)
- [Delete Store Address](#)
- [Delete Store Department Up Charges](#)
- [Delete Store Hours](#)
- [Delete Store Location Trait](#)
- [Delete Walk-Through Store](#)
- [Update Store](#)
- [Update Store Address](#)
- [Update Store Department Up Charges](#)
- [Update Store Hours](#)

Create Store

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service provides the ability to keep store data in Merchandising in sync with an external system if Merchandising is not being used as the system of record for organizational hierarchy information. The store data handled by this service includes basic store data in addition to addresses, store hours, up-charges.

Location traits and walkthrough store attributes cannot be attached while creating a store. The store create batch must first process the store before it can have these attributes attached to it. Up-charges, store hours, and addresses can be included for new stores.

The store creation also supports the ability to import custom flex attributes from an external system

When creating a new store in Merchandising via this service, the data is first validated, and then the service will create store in an asynchronous mode.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/create

Input Payload Details

Table 5-1115 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeName	Yes	String (150)	The name of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
storeNameSecondary	No	String (150)	Secondary name of the store. This field can only be populated when system_options.secondary_desc_ind = Y.
channelId	No	Number (4)	The identifier of the channel. This value must be predefined on the CHANNELS table. In a multichannel environment this field is required.

Table 5-1115 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
district	Yes	Number (10)	The number of the district of which the store is a member. This value must be predefined on the DISTRICT table.
transferZone	No	Number (4)	The transfer zone in which the store is located. This value must be predefined on the TSFZONE table.
defaultWarehouse	No	Number (10)	This field contains the default warehouse for the store. This value must be a virtual warehouse predefined on the WH table.
currencyCode	Yes	String (3)	The code of the currency under which the store operates. This value must be predefined on the CURRENCIES table. It cannot be modified.
orgUnitId	No	Number (15)	Column will contain the organizational unit ID value.
stockholdingInd	Yes	String (1)	Indicates if the store can hold stock. This field cannot be modified.
customerOrderLocationInd	No	String (1)	Customer order location indicator. This field can have Y or N.
customerOrderShippingInd	No	String (1)	This field indicates wheather the customer order has been been shipped from Warehouse or not.
giftWrappingInd	No	String (1)	This field indicates weathere a gift wrapping needs to be done or not.
onlineStoreInd	No	String (1)	This field indicates how store day will be managed by ReSA. If the indicator is Y then ReSA will automatically open and close the store day.

Table 5-1115 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeName10	No	String (10)	The ten character abbreviation of the store name.
storeName3	No	String (3)	The three character abbreviation of the store name.
storeClass	Yes	String (1)	The code of the class of which the store is a member.
storeManagerName	Yes	String (120)	The name of the store manager.
storeOpenDate	Yes	date	The date on which the store opened.
storeCloseDate	No	date	The date on which the store closed.
acquiredDate	No	date	The date on which the store was acquired.
remodelDate	No	date	The date on which the store was remodeled.
faxNo	No	String (20)	Contains the fax number for the store.
phoneNo	No	String (20)	This type can hold a number of digits count number.
email	No	String (100)	The email address of the store.
totalSquareFeet	No	Number (8)	The total square footage of the store.
sellingSquareFeet	No	Number (8)	The total square footage of the store's selling area.
linearDistance	No	Number (8)	The total merchandise space of the store.
storeFormat	No	Number (4)	The code of the store format of the store. This value must be predefined on the STORE_FORMAT table.
mallName	No	String (120)	The name of the mall in which the store is located.
stopOrderDays	No	Number (3)	The number of days before the store close date that the store will stop accepting orders.

Table 5-1115 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
startOrderDays	Yes	Number (3)	The number of days before the store open date that the store will begin accepting orders.
language	No	Number (6)	The code of the language used at the store. This value must be predefined on the LANG table.
isoCode	No	String (6)	The character code corresponding to the Language used at the store
integratedPosInd	Yes	String (1)	Indicates whether the store's POS is integrated.
dunsNumber	No	String (9)	Holds the Dun and Bradstreet (DUNS) number
dunsLocation	No	String (4)	The Dun and Bradstreet number to identify the location.
copyDeliveryInd	No	String (1)	Indicates if the like store's delivery schedule information should be copied to the new store. This value cannot be modified. It will only be populated on a store create message.
copyActivityInd	No	String (1)	Indicates if the like store's closing date schedule should be copied to the new store. This value cannot be modified. It will only be populated on a store create message.

Table 5-1115 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
priceStore	No	Number (10)	The store from which pricing information will be copied to the new store. The pricing store does not need the same currency as the new store. This value is not required on a create message and should not be populated on a modify message.
costLocation	No	Number (10)	The location from which to copy cost information to the new store. This field should only be populated on store create messages. This value must be predefined on the WH or STORE table. This value cannot be modified.
vatIncludeInd	No	String (1)	Indicates whether retail prices for the store will contain VAT.
vatRegion	No	Number (4)	The vat region of which the store is a part. This value must be predefined on the VAT_REGION table. It is required if VAT is on in the system.
likeStore	No	Number (10)	The store from which the new store will have item locations copied. This value must be predefined on the STORE table. It cannot be modified and will only be populated on a create message.
copyReplenishmentInd	No	String (1)	Indicates whether replenishment information should be copied from the like store to the new store. This field cannot be modified. It will only be populated on a store create message.

Table 5-1115 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferEntity	Yes	Number (10)	The transfer entity of which the store is a part. This value must be predefined on the TSF_ENTITY table. If the system allows intercompany transfers this field is required.
sisterStore	No	Number (10)	The store which will be used to relate historical data to the new store. This value must be predefined on the STORE table.
transactionNoGenerated	Yes	String (6)	The level at which unique POS transaction numbers are generated.
timezoneName	No	String (64)	TimeZone name.
customerId	No	Number (10)	Customer ID associated with the WF Store.
taxId	No	String (20)	Contains the unique tax identification number of the store.
copyClearanceInd	No	String (1)	Indicates if the like store's clearance items should be copied to the new store. This value cannot be modified. It will only be populated on a store create message.
autoReceive	No	String (1)	This column will indicate whether the client is allowing automatic receipt for the store. Valid Values are Y (Yes), N (No).

Table 5-1115 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
address	No	Collection of Object	References a collection of store address.
hours	No	Collection of Object	References a collection of store hours details.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1116 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressType	Yes	String (2)	Indicates the type for the address. Valid values (e. g. 01-business, 02-postal) are in the add_type table.
externalReferenceId	Yes	String (32)	This field holds the unique address ID from the source system. For addresses that are interfaced into Merchandising, it will hold the external system's address ID.
primaryAddressTypeInd	No	String (1)	Indicates whether the address type is the primary address type for the module.
primaryAddressInd	Yes	String (1)	Indicates whether this address is the primary address for this address type.
add1	Yes	String (240)	Contains the first line of the address

Table 5-1116 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
add2	No	String (240)	Contains the second line of the address.
add3	No	String (240)	Contains the thirdline of the address.
city	Yes	String (120)	Contains the name of the city that is associated with the address.
state	No	String (3)	Contains the postal abbreviation for the state in which the warehouse is located.
stateName	No	String (120)	Indicates the State name.
country	Yes	String (3)	Contains the country where the address exists.
countryName	No	String (120)	Indicates the Country Name. For e. g Brazil,United States
post	No	String (30)	Contains the zip code for the address.
county	No	String (250)	Contains the county name for the location.
jurisdictionCode	No	String (10)	ID associated to the tax jurisdiction of the country-state relationship.
contactName	No	String (120)	Contains the name of the contact for the supplier at this address.
contactPhone	No	String (20)	Contains the phone number of the contact person at this address.
contactTelex	No	String (20)	Contains the telex number of the partner or suppliers representative contact.
contactFax	No	String (20)	Contains the fax number of the contact person at this address.
contactEmail	No	String (100)	Contains the email address of the partner or suppliers representative contact.

Table 5-1116 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
oracleVendorSiteId	No	Number (15)	The unique identifier of this address in the Oracle Financials systems, if used.
phone2Type	No	String (6)	This column contains the type of the phone number stored in Phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work also can be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This column contains an additional phone number associated with the address.
phone3Type	No	String (6)	This column contains the type of the phone number stored in Phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work also can be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This column contains a second additional phone number associated with the address.
latitude	No	String (30)	This column contains the north-south geographic coordinate of the address, represented in decimal degrees (e. g. , 37. 7749).

Table 5-1116 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
longitude	No	String (30)	This column contains the east-west geographic coordinate of the address, represented in decimal degrees (e. g. , -122. 4194).
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Table 5-1117 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1118 Hours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	Yes	Number (1)	This field indicates the day of the week for which store timing is being stored.
storeOpenTime	No	String (8)	Contains the open time for the store. The time format must be in HH:MM AM or HH:MM PM.
storeCloseTime	No	String (8)	Contains the close time for the store. The time format must be in HH:MM AM or HH:MM PM.

Table 5-1119 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1120 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6001,
  "storeName": "6001 Retail",
  "storeType": "C",
  "storeNameSecondary": "storeNameSecondary",
  "channelId": 1,
  "district": 121,
  "transferZone": 1000,
  "defaultWarehouse": 5,
  "currencyCode": "USD",
  "orgUnitId": 1111111111,
  "stockholdingInd": "Y",
  "customerOrderLocationInd": "Y",
  "customerOrderShippingInd": "Y",
  "giftWrappingInd": "Y",
  "onlineStoreInd": "N",
  "storeName10": "6001Retail",
  "storeName3": "600",
  "storeClass": "A",
  "storeManagerName": "storeMgrName",
  "storeOpenDate": "2001-12-31",
  "storeCloseDate": "2001-12-31",
  "acquiredDate": "2001-12-31",
  "remodelDate": "2001-12-31",
  "faxNo": null,
  "phoneNo": "1800800800",
  "email": "someone@email.com",
  "totalSquareFeet": 762426,
  "sellingSquareFeet": 570207,
}
```

```

"linearDistance": 717780,
"storeFormat": 10,
"mallName": "mallName",
"stopOrderDays": 5,
"startOrderDays": 10,
"language": 1,
"isoCode": "en",
"integratedPosInd": "Y",
"dunsNumber": null,
"dunsLocation": null,
"copyDeliveryInd": "Y",
"copyActivityInd": "Y",
"priceStore": 6000,
"costLocation": 6000,
"vatIncludeInd": "Y",
"vatRegion": 1000,
"likeStore": null,
"copyReplenishmentInd": "Y",
"transferEntity": 1000,
"sisterStore": 1913,
"transactionNoGenerated": "S",
"timezoneName": "America/New_York",
"customerId": 121902,
"taxId": null,
"copyClearanceInd": "Y",
"autoReceive": "Y",
"customFlexAttribute": [
{
"name": "REVIEW_DATE",
"value": null,
"valueDate": null
}
],
"address": [
{
"addressType": "01",
"externalReferenceId": "REF6001",
"primaryAddressTypeInd": "Y",
"primaryAddressInd": "Y",
"add1": "101 First Street",
"add2": "201 Second Street",
"add3": "301 Third Street",
"city": "Minneapolis",
"state": "MN",
"stateName": "MN",
"country": "US",
"countryName": "US",
"post": "55555-1234",
"county": null,
"jurisdictionCode": null,
"contactName": "contactName",
"contactPhone": "1800800800",
"contactTelex": null,
"contactFax": null,
"contactEmail": "someone@email.com",
"oracleVendorSiteId": 625128,

```

```
"phone2Type": "W0",
"phone2": "5551234567",
"phone3Type": "W0",
"phone3": "5551234567",
"latitude": "44.38129",
"longitude": "-92.03521",
"customFlexAttribute": [
{
"name": "REVIEW_DATE",
"value": null,
"valueDate": null
}
],
"hours": [
{
"dayNo": 1,
"storeOpenTime": null,
"storeCloseTime": null
}
],
"localizationExtensions": [
{
"country": "BR",
"attributes": [
{
"name": "SPECIAL_CATEGORY_CODE",
"value": "WS_PRC_08",
"valueDate": "2001-12-31"
}
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
```

```

{
  "error": "must be one of Y, N",
  "field": "createRecord.arg0.approveInd",
  "inputValue": "X"
}
]
}

```

Create Store Address

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to create additional address for an existing store in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/address/create

Input Payload Details

Table 5-1121 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
address	No	Collection of Object	References a collection of store address.
customerId	No	Number (10)	Customer ID associated with the WF Store.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1122 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressType	Yes	String (2)	Indicates the type for the address. Valid values (e. g. 01-business, 02-postal) are in the add_type table.

Table 5-1122 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
externalReferenceId	Yes	String (32)	This field holds the unique address ID from the source system. For addresses that are interfaced into Merchandising, it will hold the external system's address ID.
primaryAddressTypeInd	No	String (1)	Indicates whether the address type is the primary address type for the module.
primaryAddressInd	Yes	String (1)	Indicates whether this address is the primary address for this address type.
add1	Yes	String (240)	Contains the first line of the address
add2	No	String (240)	Contains the second line of the address.
add3	No	String (240)	Contains the thirdline of the address.
city	Yes	String (120)	Contains the name of the city that is associated with the address.
state	No	String (3)	Contains the postal abbreviation for the state in which the warehouse is located.
stateName	No	String (120)	Indicates the State name.
country	Yes	String (3)	Contains the country where the address exists.
countryName	No	String (120)	Indicates the Country Name. For e. g Brazil,United States
post	No	String (30)	Contains the zip code for the address.
county	No	String (250)	Contains the county name for the location.
jurisdictionCode	No	String (10)	ID associated to the tax jurisdiction of the country-state relationship.

Table 5-1122 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactName	No	String (120)	Contains the name of the contact for the supplier at this address.
contactPhone	No	String (20)	Contains the phone number of the contact person at this address.
contactTelex	No	String (20)	Contains the telex number of the partner or suppliers representative contact.
contactFax	No	String (20)	Contains the fax number of the contact person at this address.
contactEmail	No	String (100)	Contains the email address of the partner or suppliers representative contact.
oracleVendorSiteId	No	Number (15)	The unique identifier of this address in the Oracle Financials systems, if used.
phone2Type	No	String (6)	This column contains the type of the phone number stored in Phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work also can be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This column contains an additional phone number associated with the address.

Table 5-1122 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
phone3Type	No	String (6)	This column contains the type of the phone number stored in Phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work also can be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This column contains a second additional phone number associated with the address.
latitude	No	String (30)	This column contains the north-south geographic coordinate of the address, represented in decimal degrees (e. g. , 37. 7749).
longitude	No	String (30)	This column contains the east-west geographic coordinate of the address, represented in decimal degrees (e. g. , -122. 4194).
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Table 5-1123 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1124 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1125 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
  "storeType": "C",
  "address": [
    {
      "addressType": "01",
      "externalReferenceId": "REF6001",
      "primaryAddressTypeInd": "Y",
      "primaryAddressInd": "Y",
      "add1": "101 First Street",
      "add2": "201 Second Street",
      "add3": "301 Third Street",
      "city": "Minneapolis",
      "state": "MN",
      "stateName": "stateName",
      "country": "US",

```

```

"countryName": "countryName",
"post": "55555-1234",
"county": "Some County",
"jurisdictionCode": null,
"contactName": "contactName",
"contactPhone": "1800800800",
"contactTelex": null,
"contactFax": null,
"contactEmail": "someone@email.com",
"oracleVendorSiteId": 625128,
"phone2Type": "W0",
"phone2": "5551234567",
"phone3Type": "W0",
"phone3": "5551234567",
"latitude": "44.38129",
"longitude": "-92.03521",
"customFlexAttribute": [
{
"name": "REVIEW_DATE",
"value": null,
"valueDate": "2001-12-31"
}
],
"customerId": 121902,
"localizationExtensions": [
{
"country": "BR",
"attributes": [
{
"name": "SPECIAL_CATEGORY_CODE",
"value": "WS_PRC_08",
"valueDate": "2001-12-31"
}
]
}
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Store Department Up Charges**Functional Area**

Organizational Hierarchy - Stores

Business Overview

This service is used to include department level up-charges for an existing store in Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/departmentUpCharges/
create

Input Payload Details**Table 5-1126 Create - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
upcharge	No	Collection of Object	References a collection of store department upcharges.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1127 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (6)	The hierarchy level for Up Charges. Valid values are: DI - Division, GR - Group, DE - Department, AD - All Department.

Table 5-1127 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	No	Number (4)	The value of the hierarchy. This column can contain Division/Group/Department IDs when Hier Level is 'DI', 'GR', or 'DE'. Otherwise, it should be left blank when Hier Level is 'AD'.
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
toLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
details	No	Collection of Object	References a collection of store department upcharge details.

Table 5-1128 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	Yes	String (10)	This field contains the unique identifier of the Up Charge component.

Table 5-1128 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the cost of the Item/To Location combinations within the department.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies.
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified.
upChargeGroup	Yes	String (6)	This field contains the up charge group to which the component ID belongs. Valid values can be found on the codes table with a code type of UCHG. Examples are Admin. Fee (A), Freight (F) and Special K Fees (K).
componentCurrency	Yes	String (3)	This field contains the currency of the Up Charge component.
effectiveDate	No	date	The date from which the new values are effective in the system.
itemDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing items under the department.
transferAllocationDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing transfers and allocations under the department.
computationValueBase	No	String (10)	Contains the Computation Value Basis for the Up Charge component.
costBasis	No	String (6)	Contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
includeInTotalUpChargeInd	No	String (1)	Used to determine if the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level.

Table 5-1129 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.

Table 5-1129 (Cont.) LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1130 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
  "storeType": "C",
  "upcharge": [
    {
      "hierarchyLevel": "DI",
      "hierarchyValue": 1000,
      "fromLocation": "1231",
      "toLocation": "3112",
      "fromLocationType": "S",
      "toLocationType": "S",
      "details": [
        {
          "componentId": "UC1",
          "componentRate": 3.55,
          "perCount": 3,
          "perCountUom": "EA",
          "upChargeGroup": "A",
          "componentCurrency": "USD",
          "effectiveDate": "2001-12-31",
          "itemDefaultInd": "Y",
          "transferAllocationDefaultInd": "Y",
          "computationValueBase": null,
          "costBasis": "S",
          "includeInTotalUpChargeInd": "N"
        }
      ]
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",

```

```
        "valueDate": "2001-12-31"
      }
    ]
  }
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Create Store Hours

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to define the store hours attribute for an existing store in Merchandising.

Service Type

POST

ReST URL

`MerchIntegrations/services/organizationalHierarchy/store/hours/create`

Input Payload Details

Table 5-1131 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
hours	No	Collection of Object	References a collection of store hours details.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1132 Hours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	Yes	Number (1)	This field indicates the day of the week for which store timing is being stored.
storeOpenTime	No	String (8)	Contains the open time for the store. The time format must be in HH:MM AM or HH:MM PM.
storeCloseTime	No	String (8)	Contains the close time for the store. The time format must be in HH:MM AM or HH:MM PM.

Table 5-1133 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1134 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
  "storeType": "C",
  "hours": [
    {
```



```

        "dayNo": 1,
        "storeOpenTime": null,
        "storeCloseTime": null
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Store Location Trait

Functional Area

Organizational Hierarchy - Stores

Business Overview

The primary role of this service is to assign location trait to an existing store in Merchandising. Location traits must already exist prior to being added to the store.

Service Type
POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/locationTrait/create

Input Payload Details

Table 5-1135 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
traits	No	Collection of Object	References a collection of location traits.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1136 Traits - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The identifier of the location trait. Though the node is optional, this field is required if the node is included. The node can not be populated on the store create message.

Table 5-1137 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1138 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
```

```

    "storeType": "C",
    "traits": [
      {
        "traitId": 1000
      }
    ],
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Walk-Through Store

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service can be used to assign walk-through store to an existing store in Merchandising. Walk-through stores are used in Merchandising as part of the transfer reconciliation process

and are used to indicate two or more stores that have a 'walk through' connection between them - on the sales floor and/or the backroom.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/walkThroughStore/
create

Input Payload Details

Table 5-1139 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
walkThrough	No	Collection of Object	References a collection of walkthrough stores.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1140 WalkThrough - Object. See list of elements for detail

Element Name	Required	Data Type	Description
walkThroughStore	Yes	Number (10)	A walk through store of the store being modified. Though the node is optional, if it is included this field is required. This node can not be populated on a store create message.

Table 5-1141 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1142 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
  "storeType": "C",
  "walkThrough": [
    {
      "walkThroughStore": 187308
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Store

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service provides the ability to delete an existing store data based on the request from an external system if Merchandising is not the system of record for organizational hierarchy information.

After the data has been validated, the service will add the store to the DAILY_PURGE table for processing deletion through a batch process.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/delete

Input Payload Details

Table 5-1143 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number being deleted, or for which a location trait or walk through store is being disassociated.

Sample Input Message

```
{
  "store": 6000
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "element": "store",
      "message": "Store number is required"
    }
  ]
}
```

```

    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}

```

Delete Store Address

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to delete the address from an existing store in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/address/delete

Input Payload Details

Table 5-1144 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number being deleted, or for which a location trait or walk through store is being disassociated.
address	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-1145 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
externalReferenceId	Yes	String (32)	The unique identifier of the address being deleted. For address delete messages that are interfaced into RMS, it will hold the external store system's address ID. For address delete messages published out of RMS, it will hold RMS's address ID.

Sample Input Message

```

{
  "store": 6000,
  "address": [
    {
      "externalReferenceId": "ADDREF01"
    }
  ]
}

```

```
    }  
  ]  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Delete Store Department Up Charges

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to delete department level up-charges from an existing store in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/departmentUpCharges/
delete

Input Payload Details

Table 5-1146 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number being deleted, or for which a location trait or walk through store is being disassociated.
upcharge	No	Collection of Object	Child node.

Table 5-1147 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (6)	The hierarchy level for Department Up Charges. Valid values are: DI - Division, GR - Group, DE - Department, AD - All Department.
hierarchyValue	No	Number (4)	The value of the hierarchy.
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
toLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.

Table 5-1147 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
details	No	Collection of Object	References a collection of store department upcharge details.

Table 5-1148 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	No	String (10)	This field contains the unique identifier of the Up Charge component.
itemDefaultInd	No	String (1)	Indicates whether component rate information is deleted or not for existing items under the department.

Sample Input Message

```
{
  "store": 6000,
  "upcharge": [
    {
      "hierarchyLevel": "GR",
      "hierarchyValue": 1000,
      "fromLocation": "1231",
      "toLocation": "100124",
      "fromLocationType": "S",
      "toLocationType": "S",
      "details": [
        {
          "componentId": "UC2",
          "itemDefaultInd": "Y"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Store Hours**Functional Area**

Organizational Hierarchy - Stores

Business Overview

This service is used to delete the store hours from an existing store in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/hours/delete

Input Payload Details**Table 5-1149 Delete - Object. See list of elements for detail**

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number being deleted, or for which a location trait or walk through store is being disassociated.
hours	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-1150 Hours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	Yes	Number (1)	This field indicates the day of the week for which store timing is being stored.

Sample Input Message

```
{
  "store": 6000,
```

```
"hours": [  
  {  
    "dayNo": 4  
  }  
]
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Delete Store Location Trait

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to delete the location trait from an existing store in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/locationTrait/delete

Input Payload Details

Table 5-1151 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number being deleted, or for which a location trait or walk through store is being disassociated.
traits	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-1152 Traits - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The identifier of the location trait. Though the node is optional, this field is required if the node is included. The node can not be populated on the store create message.

Sample Input Message

```
{
  "store": 6000,
  "traits": [
    {
      "traitId": 1000
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
    }
  ]
}
```

```
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}
```

Delete Walk-Through Store

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service can be used to delete the walk-through store from an existing store in Merchandising.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/walkThroughStore/delete

Input Payload Details

Table 5-1153 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number being deleted, or for which a location trait or walk through store is being disassociated.
walkThrough	No	Collection of Object	It's a referenced element. For detailed description, please refer referenced element doc.

Table 5-1154 WalkThrough - Object. See list of elements for detail

Element Name	Required	Data Type	Description
walkThroughStore	Yes	Number (10)	A walk through store of the store being modified. Though the node is optional, if it is included this field is required. This node can not be populated on a store create message.

Sample Input Message

```
{
  "store": 6000,
  "walkThrough": [
    {
      "walkThroughStore": 187308
    }
  ]
}
```

```
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Store

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service provides the ability to update an existing store data based on the request from an external system if Merchandising is not the system of record for organizational hierarchy information.

After the data has been validated, unlike store creation or deletion, the update is applied in Merchandising immediately upon message receipt.

For more details on stores, see the Create store service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/update

Availability During Nightly Batch Cycle

This service will not be available during total duration of nightly batch run cycle.

Input Payload Details

Table 5-1155 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeName	Yes	String (150)	The name of the store.
storeType	No	String (1)	The one character abbreviation of the store type. This field cannot be modified.
storeNameSecondary	No	String (150)	Secondary name of the store. This field can only be populated when system_options.secondary_desc_ind = Y.
channelId	No	Number (4)	The identifier of the channel. This value must be predefined on the CHANNELS table. In a multichannel environment this field is required.
district	Yes	Number (10)	The number of the district of which the store is a member. This value must be predefined on the DISTRICT table.
transferZone	No	Number (4)	The transfer zone in which the store is located. This value must be predefined on the TSFZONE table.
defaultWarehouse	No	Number (10)	This field contains the default warehouse for the store. This value must be a virtual warehouse predefined on the WH table.
currencyCode	Yes	String (3)	The code of the currency under which the store operates. This value must be predefined on the CURRENCIES table. It cannot be modified.
orgUnitId	No	Number (15)	Column will contain the organizational unit ID value. This field cannot be modified.

Table 5-1155 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
stockholdingInd	Yes	String (1)	Indicates if the store can hold stock. This field cannot be modified.
customerOrderLocationInd	No	String (1)	Customer order location indicator. This field can have Y or N.
customerOrderShippingInd	No	String (1)	This field indicates wheather the customer order has been shipped from Warehouse or not.
giftWrappingInd	No	String (1)	This field indicates weathere a gift wrapping needs to be done or not.
onlineStoreInd	No	String (1)	This field indicates how store day will be managed by ReSA. If the indicator is Y then ReSA will automatically open and close the store day.
storeName10	No	String (10)	The ten character abbreviation of the store name.
storeName3	No	String (3)	The three character abbreviation of the store name.
storeClass	Yes	String (1)	The code of the class of which the store is a member.
storeManagerName	Yes	String (120)	The name of the store manager.
storeOpenDate	Yes	date	The date on which the store opened.
storeCloseDate	No	date	The date on which the store closed.
acquiredDate	No	date	The date on which the store was acquired.
remodelDate	No	date	The date on which the store was remodeled.
faxNo	No	String (20)	Contains the fax number for the store.
phoneNo	No	String (20)	This type can hold a number of digits count number.

Table 5-1155 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
email	No	String (100)	The email address of the store.
totalSquareFeet	No	Number (8)	The total square footage of the store.
sellingSquareFeet	No	Number (8)	The total square footage of the store's selling area.
linearDistance	No	Number (8)	The total merchandise space of the store.
storeFormat	No	Number (4)	The code of the store format of the store. This value must be predefined on the STORE_FORMAT table.
mallName	No	String (120)	The name of the mall in which the store is located.
stopOrderDays	No	Number (3)	The number of days before the store close date that the store will stop accepting orders.
startOrderDays	Yes	Number (3)	The number of days before the store open date that the store will begin accepting orders.
language	No	Number (6)	The code of the language used at the store. This value must be predefined on the LANG table.
isoCode	No	String (6)	The character code corresponding to the Language used at the store
integratedPosInd	Yes	String (1)	Indicates whether the store's POS is integrated.
dunsNumber	No	String (9)	Holds the Dun and Bradstreet (DUNS) number
dunsLocation	No	String (4)	The Dun and Bradstreet number to identify the location.

Table 5-1155 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
priceStore	No	Number (10)	The store from which pricing information will be copied to the new store. The pricing store does not need the same currency as the new store. This value is not required on a create message and should not be populated on a modify message.
costLocation	No	Number (10)	The location from which to copy cost information to the new store. This field should only be populated on store create messages. This value must be predefined on the WH or STORE table. This value cannot be modified.
vatIncludeInd	No	String (1)	Indicates whether retail prices for the store will contain VAT.
vatRegion	No	Number (4)	The vat region of which the store is a part. This value must be predefined on the VAT_REGION table. It is required if VAT is on in the system.
likeStore	No	Number (10)	The store from which the new store will have item locations copied. This value must be predefined on the STORE table. It cannot be modified and will only be populated on a create message.
transferEntity	Yes	Number (10)	The transfer entity of which the store is a part. This value must be predefined on the TSF_ENTITY table. If the system allows intercompany transfers this field is required.

Table 5-1155 (Cont.) Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
sisterStore	No	Number (10)	The store which will be used to relate historical data to the new store. This value must be predefined on the STORE table.
transactionNoGenerated	Yes	String (6)	The level at which unique POS transaction numbers are generated.
timezoneName	No	String (64)	TimeZone name.
customerId	No	Number (10)	Customer ID associated with the WF Store.
taxId	No	String (20)	Contains the unique tax identification number of the store.
autoReceive	No	String (1)	This column will indicate whether the client is allowing automatic receipt for the store. Valid Values are Y (Yes), N (No).
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL
hours	No	Collection of Object	References a collection of store hours details.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1156 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.

Table 5-1156 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1157 Hours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	Yes	Number (1)	This field indicates the day of the week for which store timing is being stored.
storeOpenTime	No	String (8)	Contains the open time for the store. The time format must be in HH:MM AM or HH:MM PM.
storeCloseTime	No	String (8)	Contains the close time for the store. The time format must be in HH:MM AM or HH:MM PM.

Table 5-1158 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1159 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6001,
  "storeName": "6001 Retail storeName",
  "storeType": "C",
  "storeNameSecondary": "storeNameSecondary",
  "channelId": 1,
  "district": 121,
  "transferZone": 1000,
  "defaultWarehouse": 5,
  "currencyCode": "USD",
  "orgUnitId": 111111111,
  "stockholdingInd": "Y",
  "customerOrderLocationInd": "Y",
  "customerOrderShippingInd": "Y",
  "giftWrappingInd": "Y",
  "onlineStoreInd": "N",
  "storeName10": "StoreName",
  "storeName3": "Str",
  "storeClass": "A",
  "storeManagerName": "storeMgrName",
  "storeOpenDate": "2001-12-31",
  "storeCloseDate": "2001-12-31",
  "acquiredDate": "2001-12-31",
  "remodelDate": "2001-12-31",
  "faxNo": null,
  "phoneNo": "1800800800",
  "email": "someone@email.com",
  "totalSquareFeet": 762426,
  "sellingSquareFeet": 570207,
  "linearDistance": 717780,
  "storeFormat": 1000,
  "mallName": "mallName",
  "stopOrderDays": 100,
  "startOrderDays": 6,
  "language": 1,
  "isoCode": "en",
  "integratedPosInd": "Y",
  "dunsNumber": null,
  "dunsLocation": null,
  "priceStore": 6000,
  "costLocation": 6000,
  "vatIncludeInd": "Y",
  "vatRegion": 1000,
  "likeStore": null,
  "transferEntity": 1000,
  "sisterStore": 1913,
  "transactionNoGenerated": "S",
  "timezoneName": "timezoneName",
  "customerId": 121902,
  "taxId": null,
  "autoReceive": "Y",
  "customFlexAttribute": [
  {
```

```
"name": "REVIEW_DATE",
"value": null,
"valueDate": "2001-12-31"
},
],
"hours": [
{
"dayNo": 1,
"storeOpenTime": null,
"storeCloseTime": null
}
],
"localizationExtensions": [
{
"country": "BR",
"attributes": [
{
"name": "SPECIAL_CATEGORY_CODE",
"value": "WS_PRC_08",
"valueDate": "2001-12-31"
}
]
}
]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Store Address

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to update the address of an existing store in Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/address/update

Input Payload Details

Table 5-1160 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
address	No	Collection of Object	References a collection of store address.
customerId	No	Number (10)	Customer ID associated with the WF Store.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1161 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressType	Yes	String (2)	Indicates the type for the address. Valid values (e. g. 01-business, 02-postal) are in the add_type table.
externalReferenceId	Yes	String (32)	This field holds the unique address ID from the source system. For addresses that are interfaced into Merchandising, it will hold the external system's address ID.

Table 5-1161 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressTypeInd	No	String (1)	Indicates whether the address type is the primary address type for the module.
primaryAddressInd	Yes	String (1)	Indicates whether this address is the primary address for this address type.
add1	Yes	String (240)	Contains the first line of the address
add2	No	String (240)	Contains the second line of the address.
add3	No	String (240)	Contains the thirdline of the address.
city	Yes	String (120)	Contains the name of the city that is associated with the address.
state	No	String (3)	Contains the postal abbreviation for the state in which the warehouse is located.
stateName	No	String (120)	Indicates the State name.
country	Yes	String (3)	Contains the country where the address exists.
countryName	No	String (120)	Indicates the Country Name. For e. g Brazil,United States
post	No	String (30)	Contains the zip code for the address.
county	No	String (250)	Contains the county name for the location.
jurisdictionCode	No	String (10)	ID associated to the tax jurisdiction of the country-state relationship.
contactName	No	String (120)	Contains the name of the contact for the supplier at this address.
contactPhone	No	String (20)	Contains the phone number of the contact person at this address.
contactTelex	No	String (20)	Contains the telex number of the partner or suppliers representative contact.

Table 5-1161 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactFax	No	String (20)	Contains the fax number of the contact person at this address.
contactEmail	No	String (100)	Contains the email address of the partner or suppliers representative contact.
oracleVendorSiteId	No	Number (15)	The unique identifier of this address in the Oracle Financials systems, if used.
phone2Type	No	String (6)	This column contains the type of the phone number stored in Phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work also can be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This column contains an additional phone number associated with the address.
phone3Type	No	String (6)	This column contains the type of the phone number stored in Phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work also can be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This column contains a second additional phone number associated with the address.

Table 5-1161 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
latitude	No	String (30)	This column contains the north-south geographic coordinate of the address, represented in decimal degrees (e. g. , 37. 7749).
longitude	No	String (30)	This column contains the east-west geographic coordinate of the address, represented in decimal degrees (e. g. , -122. 4194).
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Table 5-1162 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1163 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1164 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```

{
  "store": 6000,
  "storeType": "C",
  "address": [
    {
      "addressType": "01",
      "externalReferenceId": "REF6001",
      "primaryAddressTypeInd": "Y",
      "primaryAddressInd": "Y",
      "add1": "101 First Street",
      "add2": "201 Second Street",
      "add3": "301 Third Street",
      "city": "Minneapolis",
      "state": "MN",
      "stateName": "stateName",
      "country": "US",
      "countryName": "countryName",
      "post": "55555-1234",
      "county": "Some County",
      "jurisdictionCode": null,
      "contactName": "contactName",
      "contactPhone": "1800800800",
      "contactTelex": null,
      "contactFax": null,
      "contactEmail": "someone@email.com",
      "oracleVendorSiteId": 625128,
      "phone2Type": "W0",
      "phone2": "5551234567",
      "phone3Type": "W0",
      "phone3": "5551234567",
      "latitude": "44.38129",
      "longitude": "-92.03521",
      "customFlexAttribute": [
        {
          "name": "REVIEW_DATE",
          "value": null,
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

```

    ],
    "customerId": 121902,
    "localizationExtensions": [
      {
        "country": "BR",
        "attributes": [
          {
            "name": "SPECIAL_CATEGORY_CODE",
            "value": "WS_PRC_08",
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Update Store Department Up Charges

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to update department level up-charges for an existing store in Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/departmentUpCharges/
update

Input Payload Details

Table 5-1165 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
upcharge	No	Collection of Object	References a collection of store department upcharges.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1166 Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (6)	The hierarchy level for Up Charges. Valid values are: DI - Division, GR - Group, DE - Department, AD - All Department.
hierarchyValue	No	Number (4)	The value of the hierarchy. This column can contain Division/Group/Department IDs when Hier Level is 'DI', 'GR', or 'DE'. Otherwise, it should be left blank when Hier Level is 'AD'.
fromLocation	No	String (10)	Contains the source location from which goods will be transferred. This column can contain Country/Area/Region IDs when From Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when From Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when From Location Type is either 'AS' or 'AW'.
toLocation	No	String (10)	Contains the destination location to which goods will be transferred. This column can contain Country/Area/Region IDs when To Location Type is 'C', 'A', or 'R'. It will be a store, virtual warehouse or physical warehouse when To Location Type is 'S', 'W' or 'PW'. Otherwise, it should be left blank when To Location Type is either 'AS' or 'AW'.

Table 5-1166 (Cont.) Upcharge - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fromLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
toLocationType	Yes	String (6)	Contains the type of source location from which goods will be transferred. Valid values are defined in code type 'LTPC'. Examples are: C - Country, A - Area, R - Region, S - Store, W - Virtual Warehouse, PW - Physical Warehouse, AS - All Stores, AW - All Warehouses.
details	No	Collection of Object	References a collection of store department upcharge details.

Table 5-1167 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
componentId	Yes	String (10)	This field contains the unique identifier of the Up Charge component.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the cost of the Item/To Location combinations within the department.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies.
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified.
upChargeGroup	Yes	String (6)	This field contains the up charge group to which the component ID belongs. Valid values can be found on the codes table with a code type of UCHG. Examples are Admin. Fee (A), Freight (F) and Special K Fees (K).
componentCurrency	Yes	String (3)	This field contains the currency of the Up Charge component.
effectiveDate	No	date	The date from which the new values are effective in the system.
itemDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing items under the department.

Table 5-1167 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferAllocationDefaultInd	No	String (1)	Indicates if component rate information is updated or not for existing transfers and allocations under the department.
computationValueBase	No	String (10)	Contains the Computation Value Basis for the Up Charge component.
costBasis	No	String (6)	Contains the basis the Up Charge will be calculated against. Valid values are defined in code type UCCB (Up Charge Cost Basis) with values: Weighted Average Cost (W), Supplier Cost (S) and Off-Invoice Net Cost (N).
includeInTotalUpChargeInd	No	String (1)	Used to determine if the value of the individual up charge component is used to calculate the total up charge (Y) or if it is only used as the basis of calculating another up charge (N). This column will default to the value in the ELC_COMP table but can be updated at the department level.

Table 5-1168 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1169 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
  "storeType": "C",
  "upcharge": [
    {
      "hierarchyLevel": "DI",
      "hierarchyValue": 1000,
      "fromLocation": "1231",
      "toLocation": "3112",

```



```

    "fromLocationType": "S",
    "toLocationType": "S",
    "details": [
      {
        "componentId": "UC1",
        "componentRate": 3.55,
        "perCount": 3,
        "perCountUom": "EA",
        "upChargeGroup": "A",
        "componentCurrency": "USD",
        "effectiveDate": "2001-12-31",
        "itemDefaultInd": "Y",
        "transferAllocationDefaultInd": "Y",
        "computationValueBase": null,
        "costBasis": "S",
        "includeInTotalUpChargeInd": "N"
      }
    ]
  },
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
    }
  ]
}

```

```

        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
    }
  ]
}

```

Update Store Hours

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service is used to modify the store hours for an existing store in Merchandising.

Service Type

PUT

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/hours/update

Input Payload Details

Table 5-1170 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
storeType	No	String (1)	The one character abbreviation of the store type.
hours	No	Collection of Object	References a collection of store hours details.
localizationExtensions	No	Collection of Object	Holds optional localization attributes that will be processed by the localization extension module.

Table 5-1171 Hours - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dayNo	Yes	Number (1)	This field indicates the day of the week for which store timing is being stored.
storeOpenTime	No	String (8)	Contains the open time for the store. The time format must be in HH:MM AM or HH:MM PM.
storeCloseTime	No	String (8)	Contains the close time for the store. The time format must be in HH:MM AM or HH:MM PM.

Table 5-1172 LocalizationExtensions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
country	Yes	String (3)	Id of the country associated with the localization attributes.
attributes	Yes	Collection of Object	This holds the details of localization attributes.

Table 5-1173 Attributes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "store": 6000,
  "storeType": "C",
  "hours": [
    {
      "dayNo": 1,
      "storeOpenTime": null,
      "storeCloseTime": null
    }
  ],
  "localizationExtensions": [
    {
      "country": "BR",
      "attributes": [
        {
          "name": "SPECIAL_CATEGORY_CODE",
          "value": "WS_PRC_08",
          "valueDate": "2001-12-31"
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Manage Store Conditions

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service allows user to add, update or delete store conditions at a given store location. Store conditions are events that may have had an impact on store traffic and sales such as power outages, weather conditions, states of emergency, road construction, or community events.

Store condition types are defined and maintained under the Store Conditions (SCND) code type. The condition code under this code type which have the used indicator set to Yes ('Y') will be applicable for set up. A store condition will have a start date and may or may not have an end date such as an ongoing condition. Conditions that occurred on a single day should have an end date equal to the start date. Extra details about the condition can be provided in the comments.

In case of error, the standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/conditions/manage

Input Payload Details

Table 5-1174 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	No	Number (4)	Number of condition records included in request

Table 5-1174 (Cont.) Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Container for store condition records.

Table 5-1175 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Contains the action for the request. Valid values are CREATE, UPDATE, DELETE.
store	Yes	Number (10)	This column contains the number which uniquely identifies the store.
conditionType	Yes	String (6)	This will hold the condition types configured under code type "SCND" with Used indicator as "Y".
startDate	Yes	date	This will hold the start date of the condition.
endDate	No	date	This will hold the end date of the condition.
clearEndDate	No	String (1)	Indicates whether to clear the end date. Valid values are Y (clear) or N (do not clear) or null.
comments	No	String (250)	This will hold the extra information about the condition set up.
clearComments	No	String (1)	Indicates whether to clear the comments. Valid values are Y (clear) or N (do not clear) or null.

Table 5-1176 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	This column contains the number which uniquely identifies the store.
conditionType	Yes	String (6)	This will hold the condition types configured under code type "SCND" with Used indicator as "Y".

Table 5-1176 (Cont.) ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
startDate	Yes	date	This will hold the start date of the condition.
errorMessages	Yes	Array of String	Contains all the errors because of which the store condition processing in Merchandising failed.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "store": 10080,
      "conditionType": "P0",
      "startDate": "2001-12-31",
      "endDate": "2001-12-31",
      "clearEndDate": "Y",
      "comments": "Power outage due to maintenance",
      "clearComments": "Y"
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-1177 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeConditions	Yes	Collection of Object	Collection of successfully processed store condition records returned in response.

Table 5-1178 StoreConditions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	This column contains the number which uniquely identifies the store.
conditionType	Yes	String (6)	This will hold the condition types configured under code type "SCND" with Used indicator as "Y".

Table 5-1178 (Cont.) StoreConditions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
startDate	Yes	date	This will hold the start date of the condition.

Sample Response Message

```
{
  "storeConditions": [
    {
      "store": 10080,
      "conditionType": "P0",
      "startDate": "2001-12-31"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-1179 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	This column contains the number which uniquely identifies the store.
conditionType	Yes	String (6)	This will hold the condition types configured under code type "SCND" with Used indicator as "Y".
startDate	Yes	date	This will hold the start date of the condition.
errorMessages	Yes	Array of String	Contains all the errors because of which the store condition processing in Merchandising failed.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",

```

```
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
},
"businessError": [
{
"store": 10080,
"conditionType": "P0",
"startDate": "2001-12-31",
"errorMessages": [
"A condition with the same type and start date already exists"
]
}
]
}
```

Manage Store Services

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service allows user to specify /add which services are provided to customers at a given store location, such as Gift Wrapping, Embroidery, Engraving, or Curbside Pickup. The list of possible store services are defined under the Store Services (STSV) code type. The services under this code type which have the used indicator set to Yes ('Y') will be applicable for set up.

The only modification that can be performed via this service is to indicate that the store service is no longer available at the provided store. This can be done by passing in a store and a store service which is currently available at the store, and the delete flag set to Yes (Y) indicating that the service should be removed.

In case of error, the standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/services/manage

Input Payload Details

Table 5-1180 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Total number of store service records included in the request payload.
items	Yes	Collection of Object	Child node holding the collection of store service records to be processed.

Table 5-1181 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	Yes	String (10)	Specifies the action to be performed on the store service record (CREATE, UPDATE, DELETE).
store	Yes	Number (10)	This column contains the number which uniquely identifies the store.
storeService	Yes	String (6)	Code representing the type of service being assigned, modified, or removed for the store.

Table 5-1182 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	No	Number (10)	Store number for which the service request failed validation or processing.
storeService	No	String (6)	Service type associated with the failed store service transaction.
errorMessages	No	Array of String	Child node holding one or more validation or processing error messages for the store service request.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "action": "CREATE",
      "store": 1810,
      "storeService": "EN"
    }
  ]
}
```

Response Code: 200 (Success)

Table 5-1183 ManageResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
storeServices	No	Collection of Object	Child node holding successfully processed store service records returned in the response.

Table 5-1184 StoreServices - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	No	Number (10)	Store number for which the service request has been successfully processed.
storeService	No	String (6)	Type of the service that the store offers. Valid values are available under code type "STSV".

Sample Response Message

```
{
  "storeServices": [
    {
      "store": 1810,
      "storeService": "EN"
    }
  ]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-1185 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	No	Number (10)	Store number for which the service request failed validation or processing.
storeService	No	String (6)	Service type associated with the failed store service transaction.

Table 5-1185 (Cont.) ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
errorMessages	No	Array of String	Child node holding one or more validation or processing error messages for the store service request.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "store": 1810,
      "storeService": "EN",
      "errorMessages": [
        "Invalid service type for store 1810"
      ]
    }
  ]
}
```

Manage Store Translations

Functional Area

Organizational Hierarchy - Stores

Business Overview

This service allows for the creation, update, and deletion of translations of a store's name and secondary name.

In case of error, the standard error response will be returned. The element `_validationErrors_` will be present when input payload or input parameters do not match the schema definition for this service.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/translations/manage

Input Payload Details

Manage - Object. See list of elements for detail

Items - Object. See list of elements for detail

ManageError - Object. See list of elements for detail

Response Code: 200 (Success)

ManageResponse - Object. See list of elements for detail

Responses - Object. See list of elements for detail

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

ManageError - Object. See list of elements for detail

Stores by Hierarchy Level Upload Service

The following services are included in this functional area:

- [Create Store Location Trait by Hierarchy Level](#)
- [Delete Store Location Trait by Hierarchy Level](#)

Create Store Location Trait by Hierarchy Level

Functional Area

Organizational Hierarchy - Stores

Business Overview

If Merchandising is not the system of record for organizational hierarchy information for an implementation, then this service can be used to assign an existing location trait to stores of a given organization hierarchy level in Merchandising based on the request from an external system.

Service Type

POST

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/
locationTraitByHierarchy/create

Input Payload Details

Table 5-1186 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The ID of the specified hierarchy level. This value must be unique among other values of the same hierarchy level.
hierarchyDescription	No	String (120)	The description of the hierarchy value.

Table 5-1186 (Cont.) Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyLevel	Yes	String (2)	The hierarchy level of the hierarchy value. This field will indicate the level of the organizational hierarchy to which the cost change applies. Valid values are CH (chain), AR (area), RE (region), DI (district).
parentHierarchyId	No	Number (10)	The ID of the hierarchy value's parent hierarchy value. This value must be predefined on either the CHAIN, AREA or REGION table.
traits	Yes	Collection of Object	References a collection of traits.

Table 5-1187 Traits - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The location trait associated to the hierarchy value. This value must be predefined on LOC_TRAITS table. A location trait cannot be associated with (or deleted from) a chain.

Sample Input Message

```
{
  "hierarchyValue": 2,
  "hierarchyDescription": null,
  "hierarchyLevel": "AR",
  "parentHierarchyId": 117516,
  "traits": [
    {
      "traitId": 1
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element validationErrors will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Store Location Trait by Hierarchy Level

Functional Area

Organizational Hierarchy - Stores

Business Overview

If Merchandising is not the system of record for organizational hierarchy information for an implementation, then this service can be used to delete a location trait from stores of a given organization hierarchy level in Merchandising based on the request from an external system.

Service Type

DELETE

ReST URL

MerchIntegrations/services/organizationalHierarchy/store/
locationTraitByHierarchy/delete

Input Payload Details

Table 5-1188 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hierarchyValue	Yes	Number (10)	The ID of the specified hierarchy level. This value must be unique among other values of the same hierarchy level.
hierarchyLevel	Yes	String (2)	The hierarchy level of the hierarchy value. This field will indicate the level of the organizational hierarchy to which the cost change applies. Valid values are CH (chain), AR (area), RE (region), DI (district).
traits	No	Collection of Object	References a collection of traits.

Table 5-1189 Traits - Object. See list of elements for detail

Element Name	Required	Data Type	Description
traitId	Yes	Number (4)	The location trait associated to the hierarchy value. This value must be predefined on LOC_TRAITS table. A location trait cannot be associated with (or deleted from) a chain.

Sample Input Message

```
{
  "hierarchyValue": 571,
  "hierarchyLevel": "DI",
  "traits": [
    {
      "traitId": 1
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Warehouse

The following services are included in this functional area:

- [Warehouse Download Service](#)
 - [Get Basic Warehouse Details](#)
 - [Get Warehouse Details](#)
 - [Get Warehouse Details for Provided Warehouse](#)

Warehouse Download Service

The following services are included in this functional area:

- [Get Basic Warehouse Details](#)
- [Get Warehouse Details](#)
- [Get Warehouse Details for Provided Warehouse](#)

Get Basic Warehouse Details

This section describes the Warehouse Detail service.

Business Overview

Warehouse Detail service allows user to retrieve Warehouse information for a selected warehouse or for all warehouses.

Service Type

Get

ReST URL

Wh/whDetail?warehouse={whNumber}

Input Parameters

Parameter Name	Required	Description
Warehouse	No	Warehouse Number. If none is specified, all warehouses will be retrieved.

Output

RestWhRecRDO

Parameter Name	Data Type
warehouse	BigDecimal
warehouseName	String
warehouseSecondaryName	String
vatRegion	BigDecimal
organizationHierarchyType	BigDecimal
organizationHierarchyValue	BigDecimal
currencyCode	String
physicalWarehouse	BigDecimal
primaryVirtualWarehouse	BigDecimal
channelId	BigDecimal
stockholdingIndicator	String

Parameter Name	Data Type
breakPackIndicator	String
redistributeWarehouseIndicator	String
restrictedIndicator	String
protectedIndicator	String
transferEntityId	BigDecimal
finisherInd	String
inboundHandlingDays	BigDecimal
organizationalUnitId	BigDecimal
virtualWarehouseType	String
customerOrderLocationIndicator	String
address1	String
address2	String
address3	String
city	String
state	String
countryId	String
post	String
contactName	String
contactPhone	String
contactEmail	String

JSON Structure

```
{
  "warehouse": null,
  "warehouseName": null,
  "warehouseSecondaryName": null,
  "vatRegion": "null",
  "organizationHierarchyType": null,
  "organizationHierarchyValue": null,
  "currencyCode": null,
  "physicalWarehouse": null,
  "primaryVirtualWarehouse": null,
  "channelId": null,
  "stockholdingIndicator": null,
  "breakPackIndicator": null,
  "redistributeWarehouseIndicator": null,
  "restrictedIndicator": null,
  "protectedIndicator": null,
  "transferEntityId": null,
  "finisherIndicator": null,
  "inboundHandlingDays": null,
  "organizationalUnitId" :null,
  "virtualWarehouseType" :null,
  "customerOrderLocationIndicator" :null,
  "address1": null,
  "address2": null,
```

```

        "address3": null,
        "city": null,
        "state": null,
        "countryId": null,
        "post": null,
        "contactName": null,
        "contactPhone": null,
        "contactEmail": null,
        "links": [],
        "hyperMediaContent": {
            "linkRDO": []
        }
    }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_WH	Yes	No	No	No
ADDR	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Warehouse Details

Endpoints

MerchIntegrations/services/foundation/warehouse

MerchIntegrations/services/foundation/warehouse/{whId}

Functional Area

Organizational Hierarchy - Warehouses

Business Overview

This service publishes warehouse master data to downstream consuming systems. It publishes only regular warehouse entities, including finisher warehouses, and excludes importer and exporter legal-entity warehouses. The payload includes warehouse setup details together with primary-address information, all published warehouse addresses, virtual-warehouse channel information, and warehouse and address custom flex attributes.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: WH, ADDR

Additional business tables: ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, COUNTRY, CHANNELS, WH_CFA_EXT, ADDR_CFA_EXT

JSON cache table: MERCHAPI_EXT_WH

JSON generation view: V_MERCHAPI_EXT_WH_JSON

Builds the warehouse header from WH, derives the primary address from ADDR joined to ADD_TYPE_MODULE and V_ADD_TYPE_TL, enriches address geography from STATE and COUNTRY, and resolves virtual-warehouse channel text from CHANNELS.

Warehouse- and address-level custom flex attributes are added from MERCHAPI_EXT_WH_CFA and MERCHAPI_EXT_WH_ADDR_CFA.

Functionally, the JSON view filters to WH.ORG_ENTITY_TYPE = 'R', so only regular warehouses, including finishers, are published; importer and exporter legal-entity warehouses are excluded from this API.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_WH_JSON and apply WH.ORG_ENTITY_TYPE = 'R'.

Mark cache rows deleted when the warehouse no longer exists in V_MERCHAPI_EXT_WH_JSON.

ICL (Integration Change Log) table: ICL_RMS_WH and ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on WH, and address changes are captured through ICL_RMS_ADDR for ADDR rows where MODULE = 'WH'.

ICL_RMS_ADDR rows are filtered to MODULE = 'WH' and republish the parent warehouse payload.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_WH

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_WAREHOUSE_ADHOC_PROCESS ->
MERCHAPI_EXT_WAREHOUSE_ADHOC_JOB

Webhook configuration api name: foundation/warehouse

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/warehouse

MerchIntegrations/services/foundation/warehouse/{whId}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)

Parameter Name	Required	Data Type	Description
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Warehouse)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1190 Input Parameter for "MerchIntegrations/services/foundation/warehouse/{whId}"

Parameter Name	Required	Data Type	Description
whId	Yes	Number	Warehouse Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-1191 PageResultsItemsMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request

Table 5-1191 (Cont.) PageResultsItemsMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1192 MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for warehouse details.
warehouse	No	Number(10,0)	This field contains the unique number that identifies the warehouse.
warehouseName	No	String (150)	This field contains the warehouse name. This is published for both physical and virtual warehouses.
whNameSecondary	No	String (150)	This field contains the secondary name of the warehouse.
email	No	String (100)	This field holds the email address for the location. This is published only for the physical warehouse.
stockholdingInd	No	String (1)	This field indicates whether the warehouse is a stockholding location. In a multichannel environment, it will be N for a physical warehouse and Y for a virtual warehouse.
channelId	No	Number(4,0)	This field contains the channel for which the virtual warehouse has been assigned. This is published for the virtual warehouse.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
channelDescription	No	String (120)	This field contains the description of the channel for which the virtual warehouse has been assigned.
currencyCode	No	String (3)	This field contains the code for the currency that the warehouse uses. This is published for physical and virtual warehouses.
physicalWarehouse	No	Number(10,0)	This field contains the number of the physical warehouse corresponding to this virtual warehouse. For physical warehouses, this field will be null.
primaryVirtualWarehouse	No	Number(10,0)	This field holds the virtual warehouse that will be used as the basis for all transactions for which only a physical warehouse, and not a virtual warehouse, has been specified.
dunsNo	No	String (9)	This field holds the Dun and Bradstreet number to identify the physical warehouse. This is published for the physical warehouse.
dunsLocation	No	String (4)	This field holds the Dun and Bradstreet location number to identify the physical warehouse. This is published for the physical warehouse.
breakPackInd	No	String (1)	This field indicates whether the warehouse can distribute less than the supplier case quantity. Valid values are Y or N. This is published for both physical and virtual warehouses.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
redistributionWareh useInd	No	String (1)	This field indicates whether the warehouse is a redistribution warehouse, which is used to flag purchase orders for review closer to the time of receipt. If this is set to yes (Y) for a physical warehouse, then it indicates that this is a dummy location for a purchase order and that the order will have its quantities redistributed to actual receiving locations closer to expected receipt. Valid values are yes (Y) and no (N). This is published for both physical and virtual warehouses.
deliveryPolicy	No	String (6)	This field contains the delivery policy of the warehouse. Next Day (NEXT) indicates that if a location is closed, the warehouse will deliver on the next day. Next Valid Delivery Day (NDD) indicates that the warehouse will wait until the next scheduled delivery day. This is published for both physical and virtual warehouses.
pricingLocation	No	Number(10,0)	This field contains the location from which pricing information was copied for this virtual warehouse. This is published for the virtual warehouse.
pricingLocationCurren cy	No	String (3)	This field contains the currency code of the pricing location. This is published for the virtual warehouse.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orgUnitId	No	Number(15,0)	This field holds the org unit ID of the virtual warehouse. This is published for the virtual warehouse.
vatRegion	No	Number(4,0)	This field holds the VAT region where the warehouse belongs. This is published for both physical and virtual warehouses.
orgHierarchyType	No	Number(4,0)	This field contains the organization type that will be used for reporting purposes for the warehouse. The type comes from the organizational hierarchy.
orgHierarchyValue	No	Number(10,0)	This field contains the code associated with the specific organizational hierarchy type. Valid values include the company number, chain number, area number, and so on.
restrictedInd	No	String (1)	This field indicates whether the inventory for this virtual warehouse is restricted, meaning it is impacted last in an inbound type of transaction occurring at the physical warehouse level.
protectedInd	No	String (1)	This field indicates whether the inventory for this virtual warehouse is protected, meaning it is affected last in outbound transactions occurring at the physical warehouse level.
forecastWarehouseInd	No	String (1)	This field determines if a virtual warehouse should be forecasted.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transferEntityId	No	Number(10,0)	This field indicates the transfer entity with which this virtual warehouse is associated.
finisherInd	No	String (1)	This field indicates whether a virtual warehouse is an internal finisher.
inboundHandlingDays	No	Number(2,0)	This field indicates the number of days that the warehouse requires to receive any item and get it to the shelf so that it is ready to pick.
virtualWarehouseType	No	String (6)	This field contains the virtual warehouse type. Valid values are CS_NT - Non-traditional,CS_RG - CSC,XD_GS - Global Sourcing,XD_RG - Cross Dock
orgEntityType	No	String (1)	This field specifies if the warehouse is a legal entity (Importer, Exporter) or a regular warehouse.
customerOrderLocationInd	No	String (1)	This field indicates if the virtual warehouse can source or fulfill customer orders.
defaultWarehouse	No	Number(10,0)	This field contains the default sourcing warehouse linked to the virtual warehouse.
giftWrappingInd	No	String (1)	This field indicates whether the warehouse supports gift wrapping.
customerOrderShippingInd	No	String (1)	This field indicates whether the virtual warehouse supports shipping customer orders.
investmentBuyInd	No	String (1)	This field indicates whether investment buy is configured for this warehouse.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
investmentBuyWarehouseLink	No	Number(10,0)	This field contains the warehouse linked to the investment buy.
replenishableInd	No	String (1)	This field determines if a warehouse is replenishable.
replenishmentSourceOrder	No	Number(1,0)	This field contains the order from which the inventory is sourced for the linked warehouses.
replenishableWarehouseLink	No	Number(10,0)	This field holds the replenishable warehouse that is linked to this virtual warehouse. This link implies that the virtual warehouse is included in the net inventory calculations for the replenishable warehouse.
taxId	No	String (20)	This field contains the unique tax identification number of the physical warehouse.
primaryAddressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressAdd1	No	String (240)	This field contains the first line of the address.
primaryAddressAdd2	No	String (240)	This field contains the second line of the address.
primaryAddressAdd3	No	String (240)	This field contains the third line of the address.
primaryAddressCity	No	String (120)	This field contains the name of the city that is associated with the address.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressState	No	String (3)	This field contains the state abbreviation for the address.
primaryAddressStateName	No	String (120)	This field contains the state description for the address.
primaryAddressCountryId	No	String (3)	This field contains the country where the address exists.
primaryAddressCountryName	No	String (120)	This field contains the country description where the address exists.
primaryAddressPost	No	String (30)	This field contains the zip code for the address.
primaryAddressContactName	No	String (120)	This field contains the name of the contact person at this address.
primaryAddressContactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
primaryAddressContactTelex	No	String (20)	This field contains the telex number of the contact person at this address.
primaryAddressContactFax	No	String (20)	This field contains the fax number of the contact person at this address.
primaryAddressContactEmail	No	String (100)	This field contains the email address of the warehouse location contact person.
primaryAddressPhone2	No	String (20)	This field contains an additional phone number associated with the primary address.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressPhone2Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressPhone3	No	String (20)	This field contains a second additional phone number associated with the primary address.
primaryAddressPhone3Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressLatitude	No	String (30)	This field contains the north-south geographic coordinate of the primary address, represented in decimal degrees.
primaryAddressLongitude	No	String (30)	This field contains the east-west geographic coordinate of the primary address, represented in decimal degrees.
primaryAddressCounty	No	String (250)	This field contains the county where the address exists.
primaryAddrJurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.

Table 5-1192 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse.
address	No	Collection of Object	This List holds all the warehouse addresses.
cacheTimestamp	No	dateTime	This field specifies the date and time when the warehouse record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1193 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1194 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressKey	No	Number(11,0)	This field contains the unique address key.
addressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
addressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressInd	No	String (1)	This field indicates whether the address is the primary address for the address type.
add1	No	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	No	String (120)	This field contains the name of the city that is associated with the address.
state	No	String (3)	This field contains the state abbreviation that is associated with the address.
stateName	No	String (120)	This field contains the state description that is associated with the address.
countryId	No	String (3)	This field contains the country where the address exists.
countryName	No	String (120)	This field contains the country description that is associated with the address.
post	No	String (30)	This field contains the zip code for the address.
contactName	No	String (120)	This field contains the name of the contact person at this address.

Table 5-1194 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
contactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
contactTelex	No	String (20)	This field contains the telex number of the contact person at this address.
contactFax	No	String (20)	This field contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This field contains the email address of the contact person at this address.
phone2Type	No	String (6)	This field contains the type of the phone number stored in the phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This field contains an additional phone number associated with the address.
phone3Type	No	String (6)	This field contains the type of the phone number stored in the phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This field contains a second additional phone number associated with the address.

Table 5-1194 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
latitude	No	String (30)	This field contains the north-south geographic coordinate of the address, represented in decimal degrees.
longitude	No	String (30)	This field contains the east-west geographic coordinate of the address, represented in decimal degrees.
county	No	String (250)	This field contains the county where the address exists.
jurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse's address.

Table 5-1195 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
```



```
"warehouse": 6000,  
"warehouseName": null,  
"whNameSecondary": null,  
"email": null,  
"stockholdingInd": "Y",  
"channelId": 1,  
"channelDescription": null,  
"currencyCode": "USD",  
"physicalWarehouse": 6000,  
"primaryVirtualWarehouse": 1,  
"dunsNo": null,  
"dunsLocation": "1055",  
"breakPackInd": null,  
"redistributionWarehouseInd": null,  
"deliveryPolicy": "NEXT",  
"pricingLocation": 1,  
"pricingLocationCurrency": null,  
"orgUnitId": 1111111111,  
"vatRegion": 1000,  
"orgHierarchyType": 1,  
"orgHierarchyValue": 1,  
"restrictedInd": null,  
"protectedInd": null,  
"forecastWarehouseInd": null,  
"transferEntityId": 1,  
"finisherInd": null,  
"inboundHandlingDays": 1,  
"virtualWarehouseType": null,  
"orgEntityType": null,  
"customerOrderLocationInd": "Y",  
"defaultWarehouse": 5,  
"giftWrappingInd": "Y",  
"customerOrderShippingInd": "Y",  
"investmentBuyInd": null,  
"investmentBuyWarehouseLink": 1,  
"replenishableInd": null,  
"replenishmentSourceOrder": 1,  
"replenishableWarehouseLink": 1,  
"taxId": null,  
"primaryAddressType": "01",  
"primaryAddressTypeDescription": null,  
"primaryAddressAdd1": "101 First Street",  
"primaryAddressAdd2": "201 Second Street",  
"primaryAddressAdd3": null,  
"primaryAddressCity": "Minneapolis",  
"primaryAddressState": "MN",  
"primaryAddressStateName": "MN",  
"primaryAddressCountryId": "US",  
"primaryAddressCountryName": "US",  
"primaryAddressPost": "55555-1234",  
"primaryAddressContactName": "Laura Johnson",  
"primaryAddressContactPhone": "1800800800",  
"primaryAddressContactTelex": null,  
"primaryAddressContactFax": "617-897-0902",  
"primaryAddressContactEmail": "someone@email.com",  
"primaryAddressPhone2": "5551234567",
```

```

"primaryAddressPhone2Type": "W0",
"primaryAddressPhone3": "5551234567",
"primaryAddressPhone3Type": "W0",
"primaryAddressLatitude": "44.38129",
"primaryAddressLongitude": "-92.03521",
"primaryAddressCounty": null,
"primaryAddrJurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"address": [
{
"addressKey": 114772,
"addressType": "01",
"addressTypeDescription": null,
"primaryAddressInd": "Y",
"add1": "101 First Street",
"add2": "201 Second Street",
"add3": null,
"city": "Minneapolis",
"state": "MN",
"stateName": "MN",
"countryId": "US",
"countryName": "US",
"post": "55555-1234",
"contactName": "Laura Johnson",
"contactPhone": "1800800800",
"contactTelex": null,
"contactFax": "617-897-0902",
"contactEmail": "someone@email.com",
"phone2Type": "W0",
"phone2": "5551234567",
"phone3Type": "W0",
"phone3": "5551234567",
"latitude": "44.38129",
"longitude": "-92.03521",
"county": null,
"jurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
}
]

```

```

]
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
WH	Yes	No	No	No
WH_CFA_EXT	Yes	No	No	No
MERCHAPI_EXT_WH	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_WH_JSON	Yes	No	No	No
ICL_RMS_WH	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes

Get Warehouse Details for Provided Warehouse

Endpoints

MerchIntegrations/services/foundation/warehouse

MerchIntegrations/services/foundation/warehouse/{whId}

Functional Area

Organizational Hierarchy - Warehouses

Business Overview

This service publishes warehouse master data to downstream consuming systems. It publishes only regular warehouse entities, including finisher warehouses, and excludes importer and exporter legal-entity warehouses. The payload includes warehouse setup details

together with primary-address information, all published warehouse addresses, virtual-warehouse channel information, and warehouse and address custom flex attributes.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: WH, ADDR

Additional business tables: ADD_TYPE_MODULE, V_ADD_TYPE_TL, STATE, COUNTRY, CHANNELS, WH_CFA_EXT, ADDR_CFA_EXT

JSON cache table: MERCHAPI_EXT_WH

JSON generation view: V_MERCHAPI_EXT_WH_JSON

Builds the warehouse header from WH, derives the primary address from ADDR joined to ADD_TYPE_MODULE and V_ADD_TYPE_TL, enriches address geography from STATE and COUNTRY, and resolves virtual-warehouse channel text from CHANNELS.

Warehouse- and address-level custom flex attributes are added from MERCHAPI_EXT_WH_CFA and MERCHAPI_EXT_WH_ADDR_CFA.

Functionally, the JSON view filters to WH.ORG_ENTITY_TYPE = 'R', so only regular warehouses, including finishers, are published; importer and exporter legal-entity warehouses are excluded from this API.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild use V_MERCHAPI_EXT_WH_JSON and apply WH.ORG_ENTITY_TYPE = 'R'.

Mark cache rows deleted when the warehouse no longer exists in V_MERCHAPI_EXT_WH_JSON.

ICL (Integration Change Log) table: ICL_RMS_WH and ICL_RMS_ADDR (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on WH, and address changes are captured through ICL_RMS_ADDR for ADDR rows where MODULE = 'WH'.

ICL_RMS_ADDR rows are filtered to MODULE = 'WH' and republish the parent warehouse payload.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_WH

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_WAREHOUSE_ADHOC_PROCESS ->
MERCHAPI_EXT_WAREHOUSE_ADHOC_JOB

Webhook configuration api name: foundation/warehouse

Service Type

GET

ReST URL

MerchIntegrations/services/foundation/warehouse

MerchIntegrations/services/foundation/warehouse/{whId}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
offsetkey	No	String	Offset key (Warehouse)
limit	No	String (6)	Pagination limit. Default value is 1000.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1196 Input Parameter for "MerchIntegrations/services/foundation/warehouse/{whId}"

Parameter Name	Required	Data Type	Description
whId	Yes	Number	Warehouse Id
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-1197 PageResultsItemsMerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1198 MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for warehouse details.
warehouse	No	Number(10,0)	This field contains the unique number that identifies the warehouse.
warehouseName	No	String (150)	This field contains the warehouse name. This is published for both physical and virtual warehouses.
whNameSecondary	No	String (150)	This field contains the secondary name of the warehouse.
email	No	String (100)	This field holds the email address for the location. This is published only for the physical warehouse.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
stockholdingInd	No	String (1)	This field indicates whether the warehouse is a stockholding location. In a multichannel environment, it will be N for a physical warehouse and Y for a virtual warehouse.
channelId	No	Number(4,0)	This field contains the channel for which the virtual warehouse has been assigned. This is published for the virtual warehouse.
channelDescription	No	String (120)	This field contains the description of the channel for which the virtual warehouse has been assigned.
currencyCode	No	String (3)	This field contains the code for the currency that the warehouse uses. This is published for physical and virtual warehouses.
physicalWarehouse	No	Number(10,0)	This field contains the number of the physical warehouse corresponding to this virtual warehouse. For physical warehouses, this field will be null.
primaryVirtualWarehouse	No	Number(10,0)	This field holds the virtual warehouse that will be used as the basis for all transactions for which only a physical warehouse, and not a virtual warehouse, has been specified.
dunsNo	No	String (9)	This field holds the Dun and Bradstreet number to identify the physical warehouse. This is published for the physical warehouse.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dunsLocation	No	String (4)	This field holds the Dun and Bradstreet location number to identify the physical warehouse. This is published for the physical warehouse.
breakPackInd	No	String (1)	This field indicates whether the warehouse can distribute less than the supplier case quantity. Valid values are Y or N. This is published for both physical and virtual warehouses.
redistributionWareh useInd	No	String (1)	This field indicates whether the warehouse is a redistribution warehouse, which is used to flag purchase orders for review closer to the time of receipt. If this is set to yes (Y) for a physical warehouse, then it indicates that this is a dummy location for a purchase order and that the order will have its quantities redistributed to actual receiving locations closer to expected receipt. Valid values are yes (Y) and no (N). This is published for both physical and virtual warehouses.
deliveryPolicy	No	String (6)	This field contains the delivery policy of the warehouse. Next Day (NEXT) indicates that if a location is closed, the warehouse will deliver on the next day. Next Valid Delivery Day (NDD) indicates that the warehouse will wait until the next scheduled delivery day. This is published for both physical and virtual warehouses.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pricingLocation	No	Number(10,0)	This field contains the location from which pricing information was copied for this virtual warehouse. This is published for the virtual warehouse.
pricingLocationCurrency	No	String (3)	This field contains the currency code of the pricing location. This is published for the virtual warehouse.
orgUnitId	No	Number(15,0)	This field holds the org unit ID of the virtual warehouse. This is published for the virtual warehouse.
vatRegion	No	Number(4,0)	This field holds the VAT region where the warehouse belongs. This is published for both physical and virtual warehouses.
orgHierarchyType	No	Number(4,0)	This field contains the organization type that will be used for reporting purposes for the warehouse. The type comes from the organizational hierarchy.
orgHierarchyValue	No	Number(10,0)	This field contains the code associated with the specific organizational hierarchy type. Valid values include the company number, chain number, area number, and so on.
restrictedInd	No	String (1)	This field indicates whether the inventory for this virtual warehouse is restricted, meaning it is impacted last in an inbound type of transaction occurring at the physical warehouse level.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
protectedInd	No	String (1)	This field indicates whether the inventory for this virtual warehouse is protected, meaning it is affected last in outbound transactions occurring at the physical warehouse level.
forecastWarehouseInd	No	String (1)	This field determines if a virtual warehouse should be forecasted.
transferEntityId	No	Number(10,0)	This field indicates the transfer entity with which this virtual warehouse is associated.
finisherInd	No	String (1)	This field indicates whether a virtual warehouse is an internal finisher.
inboundHandlingDays	No	Number(2,0)	This field indicates the number of days that the warehouse requires to receive any item and get it to the shelf so that it is ready to pick.
virtualWarehouseType	No	String (6)	This field contains the virtual warehouse type. Valid values are CS_NT - Non-traditional,CS_RG - CSC,XD_GS - Global Sourcing,XD_RG - Cross Dock
orgEntityType	No	String (1)	This field specifies if the warehouse is a legal entity (Importer, Exporter) or a regular warehouse.
customerOrderLocationInd	No	String (1)	This field indicates if the virtual warehouse can source or fulfill customer orders.
defaultWarehouse	No	Number(10,0)	This field contains the default sourcing warehouse linked to the virtual warehouse.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
giftWrappingInd	No	String (1)	This field indicates whether the warehouse supports gift wrapping.
customerOrderShippingInd	No	String (1)	This field indicates whether the virtual warehouse supports shipping customer orders.
investmentBuyInd	No	String (1)	This field indicates whether investment buy is configured for this warehouse.
investmentBuyWarehouseLink	No	Number(10,0)	This field contains the warehouse linked to the investment buy.
replenishableInd	No	String (1)	This field determines if a warehouse is replenishable.
replenishmentSourceOrder	No	Number(1,0)	This field contains the order from which the inventory is sourced for the linked warehouses.
replenishableWarehouseLink	No	Number(10,0)	This field holds the replenishable warehouse that is linked to this virtual warehouse. This link implies that the virtual warehouse is included in the net inventory calculations for the replenishable warehouse.
taxId	No	String (20)	This field contains the unique tax identification number of the physical warehouse.
primaryAddressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
primaryAddressTypeDescription	No	String (40)	This field specifies the address type description.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressAdd1	No	String (240)	This field contains the first line of the address.
primaryAddressAdd2	No	String (240)	This field contains the second line of the address.
primaryAddressAdd3	No	String (240)	This field contains the third line of the address.
primaryAddressCity	No	String (120)	This field contains the name of the city that is associated with the address.
primaryAddressState	No	String (3)	This field contains the state abbreviation for the address.
primaryAddressStateName	No	String (120)	This field contains the state description for the address.
primaryAddressCountryId	No	String (3)	This field contains the country where the address exists.
primaryAddressCountryName	No	String (120)	This field contains the country description where the address exists.
primaryAddressPost	No	String (30)	This field contains the zip code for the address.
primaryAddressContactName	No	String (120)	This field contains the name of the contact person at this address.
primaryAddressContactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
primaryAddressContactTelex	No	String (20)	This field contains the telex number of the contact person at this address.
primaryAddressContactFax	No	String (20)	This field contains the fax number of the contact person at this address.
primaryAddressContactEmail	No	String (100)	This field contains the email address of the warehouse location contact person.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddressPhone 2	No	String (20)	This field contains an additional phone number associated with the primary address.
primaryAddressPhone 2Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone 2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressPhone 3	No	String (20)	This field contains a second additional phone number associated with the primary address.
primaryAddressPhone 3Type	No	String (6)	This field contains the type of the phone number stored in the primaryAddressPhone 3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
primaryAddressLatitude	No	String (30)	This field contains the north-south geographic coordinate of the primary address, represented in decimal degrees.
primaryAddressLongitude	No	String (30)	This field contains the east-west geographic coordinate of the primary address, represented in decimal degrees.
primaryAddressCounty	No	String (250)	This field contains the county where the address exists.

Table 5-1198 (Cont.) MerchApiWarehouse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
primaryAddrJurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse.
address	No	Collection of Object	This List holds all the warehouse addresses.
cacheTimestamp	No	dateTime	This field specifies the date and time when the warehouse record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1199 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).

Table 5-1199 (Cont.) CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1200 Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
addressKey	No	Number(11,0)	This field contains the unique address key.
addressType	No	String (2)	This field specifies the type for the address. Valid values are: 01 - Business, 02 - Postal, 03 - Returns, 04 - Order, 05 - Invoice, 06 - Remittance.
addressTypeDescription	No	String (40)	This field specifies the address type description.
primaryAddressInd	No	String (1)	This field indicates whether the address is the primary address for the address type.
add1	No	String (240)	This field contains the first line of the address.
add2	No	String (240)	This field contains the second line of the address.
add3	No	String (240)	This field contains the third line of the address.
city	No	String (120)	This field contains the name of the city that is associated with the address.
state	No	String (3)	This field contains the state abbreviation that is associated with the address.
stateName	No	String (120)	This field contains the state description that is associated with the address.
countryId	No	String (3)	This field contains the country where the address exists.

Table 5-1200 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
countryName	No	String (120)	This field contains the country description that is associated with the address.
post	No	String (30)	This field contains the zip code for the address.
contactName	No	String (120)	This field contains the name of the contact person at this address.
contactPhone	No	String (20)	This field contains the phone number of the contact person at this address.
contactTelex	No	String (20)	This field contains the telex number of the contact person at this address.
contactFax	No	String (20)	This field contains the fax number of the contact person at this address.
contactEmail	No	String (100)	This field contains the email address of the contact person at this address.
phone2Type	No	String (6)	This field contains the type of the phone number stored in the phone2 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone2	No	String (20)	This field contains an additional phone number associated with the address.

Table 5-1200 (Cont.) Address - Object. See list of elements for detail

Element Name	Required	Data Type	Description
phone3Type	No	String (6)	This field contains the type of the phone number stored in the phone3 field. Valid values are AS - Assistant, CM - Company Main, HF - Home Fax, HO - Home, MA - Main, PA - Pager, WF - Work Fax, WO - Work, and can also be found on the codes table with a code type of PHTP.
phone3	No	String (20)	This field contains a second additional phone number associated with the address.
latitude	No	String (30)	This field contains the north-south geographic coordinate of the address, represented in decimal degrees.
longitude	No	String (30)	This field contains the east-west geographic coordinate of the address, represented in decimal degrees.
county	No	String (250)	This field contains the county where the address exists.
jurisdictionCode	No	String (10)	This field contains the ID associated with the tax jurisdiction of the country-state relationship.
createDateTime	No	dateTime	This field holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This List holds custom flex attributes for the warehouse's address.

Table 5-1201 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "warehouse": 6000,
      "warehouseName": null,
      "whNameSecondary": null,
      "email": null,
      "stockholdingInd": "Y",
      "channelId": 1,
      "channelDescription": null,
      "currencyCode": "USD",
      "physicalWarehouse": 6000,
      "primaryVirtualWarehouse": 1,
      "dunsNo": null,
      "dunsLocation": "1055",
      "breakPackInd": null,
      "redistributionWarehouseInd": null,
      "deliveryPolicy": "NEXT",
      "pricingLocation": 1,
      "pricingLocationCurrency": null,
      "orgUnitId": 1111111111,
      "vatRegion": 1000,
      "orgHierarchyType": 1,
      "orgHierarchyValue": 1,
      "restrictedInd": null,
      "protectedInd": null,
      "forecastWarehouseInd": null,
      "transferEntityId": 1,
      "finisherInd": null,
      "inboundHandlingDays": 1,
      "virtualWarehouseType": null,
      "orgEntityType": null,
      "customerOrderLocationInd": "Y",
      "defaultWarehouse": 5,
      "giftWrappingInd": "Y",
      "customerOrderShippingInd": "Y",
      "investmentBuyInd": null,
    }
  ]
}
```

```

"investmentBuyWarehouseLink": 1,
"replenishableInd": null,
"replenishmentSourceOrder": 1,
"replenishableWarehouseLink": 1,
"taxId": null,
"primaryAddressType": "01",
"primaryAddressTypeDescription": null,
"primaryAddressAdd1": "101 First Street",
"primaryAddressAdd2": "201 Second Street",
"primaryAddressAdd3": null,
"primaryAddressCity": "Minneapolis",
"primaryAddressState": "MN",
"primaryAddressStateName": "MN",
"primaryAddressCountryId": "US",
"primaryAddressCountryName": "US",
"primaryAddressPost": "55555-1234",
"primaryAddressContactName": "Laura Johnson",
"primaryAddressContactPhone": "1800800800",
"primaryAddressContactTelex": null,
"primaryAddressContactFax": "617-897-0902",
"primaryAddressContactEmail": "someone@email.com",
"primaryAddressPhone2": "5551234567",
"primaryAddressPhone2Type": "W0",
"primaryAddressPhone3": "5551234567",
"primaryAddressPhone3Type": "W0",
"primaryAddressLatitude": "44.38129",
"primaryAddressLongitude": "-92.03521",
"primaryAddressCounty": null,
"primaryAddrJurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"address": [
{
"addressKey": 114772,
"addressType": "01",
"addressTypeDescription": null,
"primaryAddressInd": "Y",
"add1": "101 First Street",
"add2": "201 Second Street",
"add3": null,
"city": "Minneapolis",
"state": "MN",
"stateName": "MN",
"countryId": "US",
"countryName": "US",
"post": "55555-1234",
"contactName": "Laura Johnson",

```

```

"contactPhone": "1800800800",
"contactTelex": null,
"contactFax": "617-897-0902",
"contactEmail": "someone@email.com",
"phone2Type": "W0",
"phone2": "5551234567",
"phone3Type": "W0",
"phone3": "5551234567",
"latitude": "44.38129",
"longitude": "-92.03521",
"county": null,
"jurisdictionCode": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
},
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offsetkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
WH	Yes	No	No	No
WH_CFA_EXT	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_EXT_WH	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_WH_JSON	Yes	No	No	No
ICL_RMS_WH	Yes	No	Yes	Yes
ICL_RMS_ADDR	Yes	No	Yes	Yes

Purchase Orders

The following services are included in this functional area:

- [Mobile Workflow Purchase Order Download Service](#)
 - [Create Purchase Order - Get Items for Purchase Orders](#)
 - [Create Purchase Order - Get Locations for Purchase Order](#)
 - [Create Purchase Order - Get Next Purchase Order Numbers](#)
 - [Create Purchase Order - Get Purchase Order Supplier Terms](#)
 - [Create Purchase Order - Get Suppliers for Purchase Order](#)
 - [Create Purchase Order - Refresh Purchase Order Item Locations](#)
 - [Create Purchase Order - Refresh Purchase Order Items](#)
 - [Create Purchase Order - Refresh Purchase Order Suppliers](#)
 - [Get Purchase Order Detail - Simple](#)
 - [Manage Purchase Orders - Get Open Purchase Orders](#)
 - [Manage Purchase Orders - Get Purchase Order Cancel Reasons](#)
 - [Manage Purchase Orders - Get Purchase Order Header](#)
 - [Manage Purchase Orders - Get Purchase Order Items for an Order](#)
 - [Manage Purchase Orders - Get Purchase Order Locations for an Order/Item](#)
 - [Manage Purchase Orders - Get Purchase Order Origins](#)
 - [Manage Purchase Orders - Get Purchase Order Statuses](#)
 - [Manage Purchase Orders - Get Users Who Have Created Purchase Orders](#)
- [Mobile Workflow Purchase Order Upload Service](#)
 - [Create Purchase Order - Simple](#)
 - [Manage Purchase Orders - Approve Purchase Orders](#)
 - [Manage Purchase Orders - Cancel Purchase Orders](#)
 - [Manage Purchase Orders - Reject Purchase Orders](#)
 - [Manage Purchase Orders - Update Purchase Order Dates](#)
- [Purchase Order Download Service](#)
 - [Get Purchase Order Details](#)
 - [Get Purchase Order Details by Order Number](#)
- [Purchase Order Upload Service](#)
 - [Create Pre-issued Order Numbers](#)

- [Create Purchase Order Details](#)
- [Create Purchase Order Expenses](#)
- [Create Purchase Order HTS](#)
- [Create Purchase Order HTS Assessments](#)
- [Create Purchase Order Letters of Credit](#)
- [Create Purchase Orders](#)
- [Delete Purchase Order Details](#)
- [Delete Purchase Order Expenses](#)
- [Delete Purchase Order HTS](#)
- [Delete Purchase Order HTS Assessments](#)
- [Delete Purchase Order Letters of Credit](#)
- [Delete Purchase Orders](#)
- [Update Purchase Order Details](#)
- [Update Purchase Order Expenses](#)
- [Update Purchase Order HTS](#)
- [Update Purchase Order HTS Assessments](#)
- [Update Purchase Order Letters of Credit](#)
- [Update Purchase Orders](#)

Mobile Workflow Purchase Order Download Service

The following services are included in this functional area:

- [Create Purchase Order - Get Items for Purchase Orders](#)
- [Create Purchase Order - Get Locations for Purchase Order](#)
- [Create Purchase Order - Get Next Purchase Order Numbers](#)
- [Create Purchase Order - Get Purchase Order Supplier Terms](#)
- [Create Purchase Order - Get Suppliers for Purchase Order](#)
- [Create Purchase Order - Refresh Purchase Order Item Locations](#)
- [Create Purchase Order - Refresh Purchase Order Items](#)
- [Create Purchase Order - Refresh Purchase Order Suppliers](#)
- [Get Purchase Order Detail - Simple](#)
- [Manage Purchase Orders - Get Open Purchase Orders](#)
- [Manage Purchase Orders - Get Purchase Order Cancel Reasons](#)
- [Manage Purchase Orders - Get Purchase Order Header](#)
- [Manage Purchase Orders - Get Purchase Order Items for an Order](#)
- [Manage Purchase Orders - Get Purchase Order Locations for an Order/Item](#)
- [Manage Purchase Orders - Get Purchase Order Origins](#)
- [Manage Purchase Orders - Get Purchase Order Statuses](#)
- [Manage Purchase Orders - Get Users Who Have Created Purchase Orders](#)

Create Purchase Order - Get Items for Purchase Orders

This section describes the Search Items service.

Business Overview

This service retrieves items applicable for Purchase Order. Item can be searched by either Item or VPN. Enter an item number, a partial item description, or a VPN in the search string.

1. When search type is ITEM, the search string can be an item number, a partial item number, an item description, or partial item description
2. When search type is VPN, the search string can be a VPN or partial VPN.

The items returned are constrained by the following criteria:

- Approved status.
- Transaction-level items.
- Orderable items.
- Pack items with Order Type as Each are filtered out.
- Only items belonging to Normal Merchandise Purchase Type as Department are returned.
- When a supplier is sent as input then:
 - Only items supplied by the input supplier are returned.
 - The item information is based on the Item/Supplier/Primary Origin Country.
- When supplier is not sent as input, then item information is based on the primary supplier and primary origin country.
- If the system_options.dept_level_orders is set to "Y" and the Department ID is sent as input, then only the input department items are returned.
- Items set for deletion are filtered out.

Service Type

Get

ReST URL

```
/PurchaseOrders/item?  
itemSearchType={itemSearchType}&searchString={searchString}&dept={dept}&supplier={supplier}&locations={locations}&pageSize={pageSize}&pageNumber={pageNumber}  
}
```

Input Parameters

Parameter Name	Required	Description	Valid values
itemSearchType	Yes	Search Type item or VPN.	ITEM, VPN
searchString	Yes	Search string for items Id or Name.	NA
dept	No	Selected items' department ID.	NA
supplier	No	Selected Supplier ID.	NA

Parameter Name	Required	Description	Valid values
Locations	No	Comma Separated values for selected locations' ID.	NA
PageSize	No	Maximum number of items to retrieve per page.	NA
PageNumber	No	Result page to retrieve.	NA

Output

PoltemSearchResultRDO

Parameter Name	Data Type
item	String
itemDesc	String
supplier	BigDecimal
originCountry	String
suppPackSize	BigDecimal
unitCost	BigDecimal
supplierCurrency	String
baseUnitRetail	BigDecimal
retailCurrency	String
baseRetailUnitOfMeasure	String
itemImageUrl	String
dept	BigDecimal
itemSearchLocRDO	List<PoltemSearchRstLocRDO>

PoltemSearchRstLocRDO

Parameter Name	Data Type
location	BigDecimal
locationType	String
unitRetail	BigDecimal
retailCurrency	String
unitRetailUnitOfMeasure	String
itemLocStatus	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```

{
  "links": [],
  "totalRecordCount": 1,
  "results": [
    {
      "links": [],
      "item": "100001887",
      "itemDesc": "DIT Test 12:Black:Medium",
      "supplier": 2200,
      "originCountry": "US",
      "suppPackSize": 1,
      "unitCost": 5,
      "supplierCurrency": "USD",
      "baseUnitRetail": 7.26,
      "retailCurrency": "USD",
      "baseRetailUnitOfMeasure": "EA",
      "itemImageUrl": null,
      "dept": 1102,
      "itemSearchLocRDO": [
        {
          "links": [],
          "location": 363640301,
          "locationType": "W",
          "unitRetail": 7.26,
          "retailCurrency": "USD",
          "unitRetailUnitOfMeasure": "CKG",
          "itemLocStatus": "A",
          "hyperMediaContent": {
            "linkRDO": []
          }
        }
      ],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
DAILY_PURGE	Yes	No	No	No
DEPS	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_IMAGE	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No
RPM_MERCH_RETAIL_DEF_EXPL	Yes	No	No	No
RPM_ZONE	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
V_SUPS	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Purchase Order - Get Locations for Purchase Order

This section describes the Search Location service.

Business Overview

The Web service enables location search applicable for PO. Location can be searched by either 'S'tore or 'W'arehouse. Enter a location number, a partial location number, a location description, or a partial location description in the search string.

The locations returned are constrained by the following criteria:

- Only stockholding locations are returned.
- When search type is Warehouse then:
 - Only virtual warehouses are returned.
 - Internal finishers are filtered out.
- When search type is store then only the following stores are returned:
 - Company stores.
 - Open stores.
- When system_options.org_unit_ind is set as 'Y' then:
 - When supplier is sent as input then only locations with same org_unit_id are returned.
 - When Org Unit ID is sent as input then only locations with same org_unit_id are returned.

Service Type

Get

ReST URL

```
/PurchaseOrders/location?
locationType={locationType}&searchString={searchString}&supplier={supplier}&or
gUnitId={orgUnitId}&pageSize={pageSize}&pageNumber={pageNumber}
```

Input Parameters

Parameter Name	Required	Description	Valid values
LocationType	Yes	Location type Store or warehouse.	S, W
SearchString	Yes	Search string for locations Id or Name.	NA
Supplier	No	Selected Supplier ID.	NA
OrgUnitId	No	Selected locations' Org unit ID.	NA
PageSize	No	Maximum number of locations to retrieve per page.	NA
PageNumber	No	Result page to retrieve.	NA

Output

PoLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locType	String
locName	String
locationCurrency	String
orgUnitId	BigDecimal

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "links": [
    {
      "href": "/PurchaseOrders/location?
searchString=e&pageSize=1&pageNumber=2",
      "rel": "next",
      "type": "GET",
      "methodType": null
    }
  ]
}
```

```

    }
  ],
  "totalRecordCount": 100,
  "results": [
    {
      "links": [],
      "location": 292919862,
      "locType": "S",
      "locName": "ALLOC_FD_Store_1_292919862",
      "locationCurrency": "USD",
      "orgUnitId": 1111111111,
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "hyperMediaContent": {
    "linkRDO": [
      {
        "href": "/PurchaseOrders/location?searchString=e&pageSize=1&pageNumber=2",
        "rel": "next",
        "type": "GET",
        "methodType": null
      }
    ]
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PARTNER_ORG_UNIT	Yes	No	No	No
V_STORE	Yes	No	No	No
V_WH	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Purchase Order - Get Next Purchase Order Numbers

Business Overview

This service retrieves next available order numbers from Merchandising. This service takes an optional query parameter of count to retrieve multiple (up to 999) order numbers. The count defaults to 1 if the query parameter is not provided.

Service Type

Get

ReST URL

/PurchaseOrders/order/id

Input Parameters

Parameter Name	Required	Description
count	No	Count (1-999) of order numbers to be fetched. Default value is 1.

Output

Parameter Name	Data Type	Details
order_no	Number	Deprecated (Use orderNumbers) as this service has been enhanced to return multiple order numbers. This field will contain the first order number from the orderNumbers list.
orderNumbers	Array	Contains the list of available order numbers.

JSON Structure

```
{  
  "order_no": 123456,  
  "orderNumbers": [123456, 123457]  
}
```

Table Impact

N/A

Create Purchase Order - Get Purchase Order Supplier Terms

This section describes the valid terms.

Business Overview

Retrieves all valid terms; valid terms are enabled with flag set to Yes and within the start and end active date.

Service Type

Get

ReST URL

/PurchaseOrders/supplier/terms

Input Parameters

N/A

Output

PoSupTermsRDO

Parameter Name	Data Type
terms	String
terms_code	String
terms_desc	String

JSON Structure

```
{
  "links": [],
  "terms": "108",
  "terms_code": "108",
  "terms_desc": "02 001.00% 010 000",
  "rank": null,
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
TERMS_HEAD	Yes	No	No	No
TERMS_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Purchase Order - Get Suppliers for Purchase Order

This section describes the Search Supplier service.

Business Overview

Supplier search can be, by entering either full or partial supplier site ID (numeric) or by a full or partial supplier site description in the search string.

Returned suppliers are constrained by the following criteria:

- Only active supplier sites are returned.
- When items are sent as input, then only supplier sites that are common amongst the items are returned.
- When locations are sent as input, then only suppliers that are valid for the Org Units associated with the input locations are returned.

Service Type

Get

ReST URL

```
/PurchaseOrders/supplier?  
supplierSearchString={supplierSearchString}&locations={locations}&items={items  
&pageSize={pageSize}&pageNumber={pageNumber}
```

Input Parameters

Parameter Name	Required	Description
SupplierSearchString	Yes	Search string for Supplier's ID or Name.
Item	No	Comma Separated values for items.
Locations	No	Comma Separated values for locations.
PageSize	No	Maximum number of suppliers to retrieve per page.
PageNumber	No	Result page to retrieve.

Output

PoSupSearchResultRDO

Parameter Name	Data Type
supplier	BigDecimal
supplierName	String
supplierCurrency	String
terms	String
defaultItemLeadTime	BigDecimal
supplierSearchItemRDO	List<PoSupItemResultRDO>
supplierSearchItemLocRDO	List<PoSupItemLocResultRDO>

PoSupItemResultRDO

Parameter Name	Data Type
item	String
originCountryId	String
leadTime	BigDecimal

PoSupItemLocResultRDO

Parameter Name	Data Type
item	String
location	BigDecimal

Parameter Name	Data Type
pickupLeadTime	BigDecimal

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "links": [],
  "totalRecordCount": 1,
  "results": [
    {
      "links": [],
      "supplier": 2200,
      "supplierName": "Our Supplier",
      "supplierCurrency": "USD",
      "terms": "04",
      "defaultItemLeadTime": 2,
      "supplierSearchItemRDO": [
        {
          "links": [],
          "item": "100001887",
          "originCountryId": "US",
          "leadTime": 2,
          "hyperMediaContent": {
            "linkRDO": []
          }
        }
      ],
      "supplierSearchItemLocRDO": [
        {
          "links": [],
          "item": "100001887",
          "location": 363640301,
          "pickupLeadTime": null,
          "hyperMediaContent": {
            "linkRDO": []
          }
        }
      ],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```



```
}  
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ITEM_SUPP_COUNTRY_LOC	Yes	No	No	No
STORE	Yes	No	No	No
SUPS	Yes	No	No	No
V_SUPS	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Purchase Order - Refresh Purchase Order Item Locations

This section describes the Load Locations service.

Business Overview

This Web service allows the user to refresh already selected PO locations records.

Service Type

Get

ReST URL

/PurchaseOrders/location/load?locations={locations}&supplier={supplier}

Input Parameters

Parameter Name	Required	Description
Locations	Yes	Comma Separated values for selected locations' ID.
Supplier	No	Selected Supplier ID.

Output

PoLocSearchResultRDO

Parameter Name	Data Type
location	BigDecimal
locType	String
locName	String
locationCurrency	String

Parameter Name	Data Type
orgUnitId	BigDecimal

JSON Structure

```
{
  "links": [],
  "location": 292919862,
  "locType": "S",
  "locName": "ALLOC_FD_Store_1_292919862",
  "locationCurrency": "USD",
  "orgUnitId": 1111111111,
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PARTNER_ORG_UNIT	Yes	No	No	No
V_STORE	Yes	No	No	No
V_WH	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Purchase Order - Refresh Purchase Order Items

This section describes the load items.

Business Overview

The primary use of loading items Web service is to refresh already selected PO items records.

Service Type

Get

ReST URL

/PurchaseOrders/item/load?item=item&supplier={supplier}&locations={locations}

Input Parameters

Parameter Name	Required	Description
Items	Yes	Comma Separated values for selected items' ID.

Parameter Name	Required	Description
Supplier	No	Selected Supplier ID.
Locations	No	Comma Separated values for selected locations' ID.

Output

PoltemSearchResultRDO

Parameter Name	Data Type
item	String
itemDesc	String
supplier	BigDecimal
originCountry	String
suppPackSize	BigDecimal
unitCost	BigDecimal
supplierCurrency	String
baseUnitRetail	BigDecimal
retailCurrency	String
baseRetailUnitOfMeasure	String
itemImageUrl	String
dept	BigDecimal
itemSearchLocRDO	List<PoltemSearchRstLocRDO>

PoltemSearchRstLocRDO

Parameter Name	Data Type
location	BigDecimal
locationType	String
unitRetail	BigDecimal
retailCurrency	String
unitRetailUnitOfMeasure	String
itemLocStatus	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "links": [],
```

```

"totalRecordCount": 1,
"results": [
  {
    "links": [],
    "item": "100001887",
    "itemDesc": "DIT Test 12:Black:Medium",
    "supplier": 2200,
    "originCountry": "US",
    "suppPackSize": 1,
    "unitCost": 5,
    "supplierCurrency": "USD",
    "baseUnitRetail": 7.26,
    "retailCurrency": "USD",
    "baseRetailUnitOfMeasure": "EA",
    "itemImageUrl": null,
    "dept": 1102,
    "itemSearchLocRDO": [
      {
        "links": [],
        "location": 363640301,
        "locationType": "W",
        "unitRetail": 7.26,
        "retailCurrency": "USD",
        "unitRetailUnitOfMeasure": "CKG",
        "itemLocStatus": "A",
        "hyperMediaContent": {
          "linkRDO": []
        }
      }
    ],
    "hyperMediaContent": {
      "linkRDO": []
    }
  },
  {
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
]
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
DAILY_PURGE	Yes	No	No	No
DEPS	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_IMAGE	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
RPM_MERCH_RETAIL_DEF_EXPL	Yes	No	No	No
RPM_ZONE	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
V_SUPS	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Create Purchase Order - Refresh Purchase Order Suppliers

This section describes the load supplier service.

Business Overview

Loading supplier Web service allows a user to refresh the selected supplier records.

Service Type

Get

ReST URL

```
/PurchaseOrders/supplier/load?
suppliers={suppliers}&locations={locations}&items={items}
```

Input Parameters

Parameter Name	Required	Description
Supplier	Yes	Supplier's ID.
Item	No	Comma Separated values for items.
Locations	No	Comma Separated values for locations.

Output

PoSupSearchResultRDO

Parameter Name	Data Type
supplier	BigDecimal
supplierName	String
supplierCurrency	String
terms	String
defaultItemLeadTime	BigDecimal
supplierSearchItemRDO	List<PoSupItemResultRDO>
supplierSearchItemLocRDO	List<PoSupItemLocResultRDO>

PoSupItemResultRDO

Parameter Name	Data Type
item	String
originCountryId	String
leadTime	BigDecimal

PoSupItemLocResultRDO

Parameter Name	Data Type
item	String
location	BigDecimal
pickupLeadTime	BigDecimal

JSON Structure

```
{
  "links": [],
  "totalRecordCount": 1,
  "results": [
    {
      "links": [],
      "supplier": 2200,
      "supplierName": "Our Supplier",
      "supplierCurrency": "USD",
      "terms": "04",
      "defaultItemLeadTime": 2,
      "supplierSearchItemRDO": [
        {
          "links": [],
          "item": "100001887",
          "originCountryId": "US",
          "leadTime": 2,
          "hyperMediaContent": {
            "linkRDO": []
          }
        }
      ],
      "supplierSearchItemLocRDO": [
        {
          "links": [],
          "item": "100001887",
          "location": 363640301,
          "pickupLeadTime": null,
          "hyperMediaContent": {
            "linkRDO": []
          }
        }
      ],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ]
}
```

```
    }  
  ],  
  "hyperMediaContent": {  
    "linkRDO": []  
  }  
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPPLIER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ITEM_SUPP_COUNTRY_LOC	Yes	No	No	No
STORE	Yes	No	No	No
SUPS	Yes	No	No	No
V_SUPS	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Get Purchase Order Detail - Simple

This section describes the Purchase Order Detail Service.

Business Overview

Purchase Order Detail service allows user to retrieve purchase order information for a selected order.

Service Type

Get

ReST URL

Po/poDetail?orderNumber={orderNumber}

Input Parameters

Parameter Name	Required	Description
Order Number	Yes	Order Number

Output

RestPoRecRDO

Parameter Name	Data Type
orderNumber	BigDecimal

Parameter Name	Data Type
orderType	String
group	BigDecimal
division	BigDecimal
dept	BigDecimal
buyer	BigDecimal
supplier	BigDecimal
supplierStatus	String
locationType	String
location	BigDecimal
writtenDate	Date
notBeforeDate	Date
notAfterDate	Date
otbEndOfWeekDate	Date
earliestShipDate	Date
latestShipDate	Date
closeDate	Date
terms	String
freightTerms	String
originIndicator	BigDecimal
shipmentMethod	String
purchaseType	String
status	String
currencyCode	String
masterPurchaseOrderNumber	BigDecimal
poltemTbl	List<RestPoltemRecRDO>

RestPoltemRecRDO

Parameter Name	Data Type
item	String
refernceltem	String
packItem	String
originCountryId	String
earliestShipDate	Date
latestShipDate	Date
supplierPackSize	BigDecimal
location	BigDecimal
locationType	String
physicalWarehouse	BigDecimal
unitRetail	BigDecimal
quantityOrdered	BigDecimal

Parameter Name	Data Type
quantityPrescaled	BigDecimal
quantityReceived	BigDecimal
lastReceivedQuantity	BigDecimal
lastRoundQuantity	BigDecimal
lastGroupRoundedQunatity	BigDecimal
quantityCancelled	BigDecimal
cancelCode	String
cancelDate	Date
unitCost	BigDecimal
costSource	String
nonScaleIndicator	String
estimatedStockDate	Date
restPolItemExpTbl	List<RestPolItemExpRecRDO>

RestPolItemExpRecRDO

Parameter Name	Data Type
item	String
packItem	String
location	BigDecimal
locationType	String
componentId	String
componentDecsption	String
alwaysDefaultIndicator	String
componentRate	BigDecimal
componentCurrency	String
exchangeRate	BigDecimal
estimatedExpenValue	BigDecimal

JSON Structure

```
{
  "orderNumber": null,
  "orderType": null,
  "group": null,
  "division": null,
  "dept": null,
  "buyer": null,
  "supplier": null,
  "supplierStatus": null,
  "locationType": null,
  "location": null,
  "writtenDate": null,
  "notBeforeDate": null,
  "notAfterDate": null,
```

```

"otbEndOfWeekDate": null,
"earliestShipDate": null,
"latestShipDate": null,
"closeDate": null,
"terms": null,
"freightTerms": null,
"originIndicator": null,
"shipmentmethod": null,
"purchaseType": null,
"status": null,
"currencyCode": null,
"masterPurchaseOrderNumber": null,
"poItemTbl": [
  {
    "item": null,
    "referenceItem": null,
    "originCountryId": null,
    "earliestShipDate": null,
    "latestShipDate": null,
    "supplierPackSize": null,
    "location": null,
    "locationType": null,
    "physicalWarehouse": null,
    "unitRetail": null,
    "quantityOrdered": null,
    "quantityPrescaled": null,
    "quantityReceived": null,
    "lastReceivedQuantity": null,
    "lastRoundQuantity": null,
    "lastGroupRoundedQuantity": null,
    "quantityCancelled": null,
    "cancelCode": null,
    "unitCost": null,
    "costSource": null,
    "nonScaleIndicator": null,
    "estimatedStockDate": null,
    "poItemExpTbl": [
      {
        "item": null,
        "packItem": null,
        "location": null,
        "locationType": null,
        "componentId": null,
        "componentDescription": null,
        "alwaysDefaultIndicator": null,
        "componentRate": null,
        "componentCurrency": null,
        "exchangeRate": null,
        "estimatedExpenditureValue": null,
        "links": [],
        "hyperMediaContent": {
          "linkRDO": []
        }
      }
    ]
  },
  "links": [],

```

```
        "hyperMediaContent": {
          "linkRDO": []
        }
      },
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ORDHEAD	Yes	No	No	No
ORDLOC	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC_EXPENSES	Yes	No	No	No
V_DEPS	Yes	No	No	No
SUPS	Yes	No	No	No
WH	Yes	No	No	No
ELC_COMP	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Open Purchase Orders

This section describes the Purchase Order Search service.

Functional Area

Procurement

Business Overview

This service retrieves summary information on all none closed purchase orders that match input criteria.

Service Type

Get

ReST URL

```
/PurchaseOrders/recent/purchaseOrderSearch?
statuses={statuses}&createIds={createIds}&startCreateDate={startCreateDate}&en
dCreateDate={endCreateDate}&orderNumber={orderNumber}&suppliers={suppliers}&or
iginCodes={originCodes}&departments={departments}&pageSize={pageSize}&pageNumb
er={pageNumber}
```

Input Parameters

Parameter Name	Required	Description	Valid values
statuses	No	List of order status	A, S, W
createIds	No	List of user IDs who created the PO	
startCreateDate	No	Long format date for starting period	
endCreateDate	No	Long format date for end period	
orderNumber	No	Order number to retrieve	
suppliers	No	List of order suppliers	
originCodes	No	List of valid Origin codes	
departments	No	List of valid order/item departments	
pageSize		Maximum number of orders to retrieve per page	
pageNumber		Result page to retrieve	

Output

RpoSearchResRDO

Parameter Name	Data Type
orderNumber	BigDecimal
status	String
supplier	BigDecimal
supplierName	String
notBeforeDate	Long
notAfterDate	Long
totalCost	BigDecimal
currency	String
previouslyApprovedIndicator	String
editableIndicator	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "type": "paginationRDO",
  "totalRecordCount": 252,
  "hyperMediaContent": {},
  "links": [],
```

```
"results": [{
  "orderNumber": 12453253,
  "statusId" : "W",
  "supplierId": 124121,
  "supplierName": "Some Supplier Site",
  "notBeforeDate": 35235252,
  "notAfterDate": 325235252351,
  "totalCost": 243.231,
  "currencyCode": "USD"
}]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_MASTER	Yes	No	No	No
PRODUCT_CONFIG_OPTIONS	Yes	No	No	No
V_ORDHEAD	Yes	No	No	No
V_ORDSKU	Yes	No	No	No
V_SUPS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Purchase Order Cancel Reasons

This section describes the Cancel Reason Code List service.

Functional Area

Procurement

Business Overview

Retrieves all purchase order cancel reason codes.

Service Type

Get

ReST URL

/PurchaseOrders/recent/cancelReasonCodeList

Input Parameters

No input.

Output

CodeDetailRDO

Parameter Name	Data Type
code	String
codeDescription	String
codeSequence	BigDecimal

JSON Structure

```
[
  {
    "code": null,
    "codeDescription": null,
    "codeSequence": null,
    "links": [ ],
    "hyperMediaContent": {
      "linkRDO": [ ]
    }
  }
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Purchase Order Header

This section describes the Get Purchase Order Summary service.

Functional Area

Procurement

Business Overview

This service retrieves purchase order header detail with open to buy information.

Service Type

Get

ReST URL

/PurchaseOrders/recent/PurchaseOrderSummary?orderNumber={orderNumber}

Input Parameters

Order Number-Required

Output

RpoOrderSumRDO

Parameter Name	Data Type
orderNumber	BigDecimal
status	String
supplier	BigDecimal
supplierName	String
notBeforeDate	Long
notAfterDate	Long
otbEowDate	Long
terms	String
termsCode	String
termsDescription	String
totalCost	BigDecimal
totalRetail	BigDecimal
Currency	String
createId	String
writtenDate	Long
defaultDisplayLevel	String
previouslyApprovedIndicator	String
editableIndicator	String
otbTable	List<RpoOrderSumOtbRDO>

RpoOrderSumOtbRDO

Parameter Name	Data Type
department	BigDecimal
classId	BigDecimal
subclassId	BigDecimal
subclassName	String
orderAmount	BigDecimal
budgetAmount	BigDecimal
receivedAmount	BigDecimal
approvedAmount	BigDecimal
outstandingAmount	BigDecimal

JSON Structure

```
{
  "orderNumber":12345,
  "statusId":"W",
  "supplierId":12345,
  "supplierName": "Supplier 12345",
```

```

    "notBeforeDate": 1234567,
    "notAfterDate": 236573,
    "terms": "01",
    "termsCode": "01234",
    "termsDescription": "Letter Of Credit",
    "totalCost": 123.45,
    "totalRetail": 456.78,
    "currencyCode": "CAD",
    "createdBy": "BUYER",
    "writtenDate": 1234567,
    "otbResults":
    [{
        "department" : 12345,
        "classId": 12345,
        "subClassId" : 12345,
        "subClassName": "subClassName"
        "budgetAmount": 12345.545,
        "orderAmount": 12345.545,
        "receivedAmount": 12345.545,
        "approvedAmount": 12345.545
    }]
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
COST_ZONE_GROUP_LOC	Yes	No	No	No
COUNTRY_ATTRIB	Yes	No	No	No
DEPS	Yes	No	No	No
ELC_COMP	Yes	No	No	No
ITEM_COST_HEAD	Yes	No	No	No
ITEM_EXP_DETAIL	Yes	No	No	No
ITEM_EXP_HEAD	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORDLOC	Yes	No	No	No
ORDLOC_EXP	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDSKU HTS	Yes	No	No	No
ORDSKU HTS_ASSESS	Yes	No	No	No
OTB	Yes	No	No	No
PERIOD	Yes	No	No	No
PRODUCT_CONFIG_OPTIONS	Yes	No	No	No
STORE	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
SUPS	Yes	No	No	No
V_ORDHEAD	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
V_SUBCLASS_TL	Yes	No	No	No
V_SUPS	Yes	No	No	No
V_TERMS_HEAD_TL	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Purchase Order Items for an Order

This section describes the Get Purchase Order Items service.

Functional Area

Procurement

Business Overview

This service retrieves items details for an order number. Based on the display level, the items record aggregates to the level specified when applicable.

Service Type

Get

ReST URL

/PurchaseOrders/recent/PurchaseOrderItems?
orderNumber={orderNumber}&itemDisplayLevel={itemDisplayLevel}&pageSize={pageSi
ze}&pageNumber={pageNumber}

Input Parameters

- Order Number - Required
- Item Display Level - Optional - valid values PARENT_LEVEL, PARENT_DIFF_LEVEL, or TRAN_LEVEL
- Page Size - Optional
- Page Number - Optional

Output

RpoOrderSumItemRDO

Parameter Name	Data Type
item	String
ItemDescription	String

Parameter Name	Data Type
diff1	String
diff1Description	String
diff2	String
diff2Description	String
diff3	String
diff3Description	String
diff4	String
diff4Description	String
quantityOrdered	BigDecimal
totalCost	BigDecimal
currency	String
itemImageUrl	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "type": "paginationRDO",
  "totalRecordCount": 252,
  "hyperMediaContent": {},
  "links": [],
  "orderNumber": 1212131,
  "results": [{
    "itemId": 1234
    "itemDescription": "some item",
    "firstDiffId": 123424,
    "firstDiffDescription": "desc",
    "secondDiffId": 12345
    "secondDiffDescription" : "desc",
    "thirdDiffId": 1234324
    "thirdDiffDescription" : "desc",
    "fourthDiffId" : 1324,
    "fourthDiffDescription" : "desc",
    "quantityOrdered": 100,
    "totalCost" : 12345.353,
    "currencyCode": "USD",
    "itemImageUrl": "http://..."
  }]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_IMAGE	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ORDLOC	Yes	No	No	No
ORDSKU	Yes	No	No	No
ORDLOC_WKSHT	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Purchase Order Locations for an Order/Item

This section describes the Get Purchase Order Item Locations service.

Functional Area

Procurement

Business Overview

This service retrieves item location details for an order number. The location record aggregates based on the display level when applicable.

Service Type

Get

ReST URL

```
/PurchaseOrders/recent/PurchaseOrderItemLocations?
orderNumber={orderNumber}&item={item}&itemDisplayLevel={itemDisplayLevel}&diff
1={diff1}&diff2={diff2}&diff3={diff3}&diff4={diff4}&pageSize={pageSize}&pageNu
mber={pageNumber}
```

Input Parameters

Parameter Name	Required	Description	Valid values
orderNumber	Yes	Order number	
item	Yes	Item Id	
itemDisplayLevel	No	Item display level	PARENT_LEVEL, PARENT_DIFF_LEVEL, TRAN_LEVEL
diff1	No	Diff1 Id	
diff2	No	Diff2 Id	
diff3	No	Diff3 Id	
diff4	No	Diff4 Id	

Parameter Name	Required	Description	Valid values
pageSize	No	Maximum number of items to retrieve per page	
pageNumber	No	Result page to retrieve	

Output

RpoOrderItemLocRDO

Parameter Name	Data Type
location	BigDecimal
locationName	String
quantityOrdered	BigDecimal
totalCost	BigDecimal
currency	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "locations" : [
    {
      "locationId" : 12345,
      "locationName" : "some location",
      "orderedQuantity" : 1000,
      "totalCost" : 12345.234,
      "currencyCode" : "USD"
    },
    {
      "locationId" : 12345,
      "locationName" : "some location",
      "orderedQuantity" : 1000,
      "totalCost" : 12345.234,
      "currencyCode" : "USD"
    }
  ]
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_MASTER	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
ORDLOC	Yes	No	No	No
V_STORE_TL	Yes	No	No	No
V_WH_TL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Purchase Order Origins

This section describes the Origin Code List service.

Functional Area

Procurement

Business Overview

Retrieves all purchase order origin codes.

Service Type

Get

ReST URL

/PurchaseOrders/recent/originCodeList

Input Parameters

No input.

Output

CodeDetailRDO

Parameter Name	Data Type
code	String
codeDescription	String
codeSequence	BigDecimal

JSON Structure

```
[
  {
    "code": null,
    "codeDescription": null,
    "codeSequence": null,
    "links": [ ],
    "hyperMediaContent": {
      "linkRDO": [ ]
    }
  }
]
```

```
    }  
  }  
]
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Purchase Order Statuses

This section describes the Purchase Order Status list.

Functional Area

Procurement

Business Overview

Retrieves all valid purchase order statuses.

Service Type

Get

ReST URL

/PurchaseOrders/recent/purchaseOrderStatusList

Input Parameters

No input.

Output

CodeDetailRDO

Parameter Name	Data Type
code	String
codeDescription	String
codeSequence	BigDecimal

JSON Structure

```
[  
  {  
    "code": null,  
    "codeDescription": null,  
    "codeSequence": null,  
  },  
  ...  
]
```

```

        "codeSequence": null,
        "links": [ ],
        "hyperMediaContent": {
            "linkRDO": [ ]
        }
    }
]

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Get Users Who Have Created Purchase Orders

This section describes the Search Purchase Order User ID.

Functional Area

Procurement

Business Overview

This service retrieves a list of user IDs associated with creating a purchase order.

Service Type

Get

ReST URL

```

/PurchaseOrders/recent/searchUserIds?
searchString={searchString}&pageSize={pageSize}&pageNumber={pageNumber}

```

Input Parameters

- Search String - Required
- Page Size - Optional
- Page Number - Optional

Output

VarcharIdRDO

Parameter Name	Data Type
id	String

PagedResultsRDO

Parameter Name	Data Type
totalRecordCount	BigDecimal
Next Page URL	String
Previous Page URL	String

JSON Structure

```
{
  "totalRecordCount": null,
  "results": [
    {
      "id": null,
      "links": [ ],
      "hyperMediaContent": {
        "linkRDO": [ ]
      }
    }
  ],
  "links": [ ],
  "hyperMediaContent": {
    "linkRDO": [ ]
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
V_ORDHEAD	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Mobile Workflow Purchase Order Upload Service

The following services are included in this functional area:

- [Create Purchase Order - Simple](#)
- [Manage Purchase Orders - Approve Purchase Orders](#)
- [Manage Purchase Orders - Cancel Purchase Orders](#)
- [Manage Purchase Orders - Reject Purchase Orders](#)
- [Manage Purchase Orders - Update Purchase Order Dates](#)

Create Purchase Order - Simple**Business Overview**

This Web service calls the existing Merchandising XOrder API directly with input parameters. For more information on Merchandising XOrder API, see the sections addressing both the Store Order Subscription API and the PO Subscription API.

Service Type

Post

ReST URL

/PurchaseOrders

Input Parameters

Example json RDO input:

```
{
  "links" : [ ],
  "itemRDOs" : [ {
    "links" : [ ],
    "item" : null,
    "location" : null,
    "unitCost" : null,
    "referenceItem" : null,
    "originCountryId" : null,
    "suppPackSize" : null,
    "qtyOrdered" : null,
    "locationType" : null,
    "cancelInd" : null,
    "reInstateInd" : null,
    "hyperMediaContent" : {
      "linkRDO" : [ ]
    }
  } ],
  "orderNo" : null,
  "supplier" : null,
  "currencyCode" : null,
  "terms" : null,
  "notBeforeDate" : null,
  "notAfterDate" : null,
  "status" : "A",
  "writtenDate" : null,
  "origInd" : null,
  "user_id" : null,
  "dept" : null,
  "exchangeRate" : null,
  "includeOnOrdInd" : null,
  "ediPoInd" : null,
  "preMarkInd" : null,
  "comment" : null,
  "otbEowDate" : null,
  "hyperMediaContent" : {
    "linkRDO" : [ ]
  }
}
```

Output

NA

Table Impact

For more information on Merchandising XOrder API, see the Store Order Subscription API and the PO Subscription API sections.

Manage Purchase Orders - Approve Purchase Orders

This section describes the Approve Purchase Orders service.

Functional Area

Procurement

Business Overview

This service approves a list of purchase orders.

Service Type

Post

ReST URL

/PurchaseOrders/recent/approvePurchaseOrders?orderNumbers={orderNumbers}

Input Parameters

Order Number -Required-comma separated list

Output

RpoStatusRDO

Parameter Name	Data Type
successOrdersCount	BigDecimal
successOrdersTable	List<BigDecimal>
failOrdersCount	BigDecimal
failOrdersTable	List<RpoFailRDO>

RpoFailRDO

Parameter Name	Data Type
orderNumber	BigDecimal
errorMessage	String

JSON Structure:

```
{
  "successOrdersCount": 0,
  "successOrdersTable": [],
  "failOrdersCount": 2,
  "failOrdersTable": [
    {
      "orderNumber": 123,
      "errorMessage": " Invalid Order Number.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
}
```

```

    {
      "orderNumber": 987,
      "errorMessage": "Invalid Order Number.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALC_HEAD_TEMP	No	No	No	Yes
ALLOC_CHRG_TEMP	No	No	No	Yes
ALLOC_DETAIL	Yes	No	Yes	No
ALLOC_DETAIL_TEMP	No	No	No	Yes
ALLOC_HEADER	Yes	No	Yes	No
ALLOC_HEADER_TEMP	No	No	No	Yes
CONTRACT_COST_HIST	Yes	Yes	No	No
CONTRACT_DETAIL	Yes	No	Yes	No
CONTRACT_HEADER	Yes	No	Yes	No
DEAL_ACTUALS_FORECAST	No	No	No	Yes
DEAL_ACTUALS_ITEM_LOC	No	No	No	Yes
DEAL_COMP_PROM	No	No	No	Yes
DEAL_DETAIL	No	No	No	Yes
DEAL_HEAD	No	No	No	Yes
DEAL_HEAD_CFA_EXT	No	No	No	Yes
DEAL_ITEMLOC_DCS	No	No	No	Yes
DEAL_ITEMLOC_DIV_GRP	No	No	No	Yes
DEAL_ITEMLOC_ITEM	No	No	No	Yes
DEAL_ITEMLOC_PARENT_DIFF	No	No	No	Yes
DEAL_QUEUE	No	No	No	Yes
DEAL_THRESHOLD	No	No	No	Yes
DEAL_THRESHOLD_REV	No	No	No	Yes
DOC	Yes	No	No	No
DOC_LINK	Yes	No	No	No
ITEM_LOC	Yes	No	Yes	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ITEM_TICKET	Yes	No	No	No
LC_ACTIVITY	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
LC_AMENDMENTS	Yes	Yes	No	No
LC_DETAIL	Yes	Yes	No	No
LC_HEAD	Yes	No	Yes	No
LC_ORDAPPLY	No	Yes	No	Yes
ORD_INV_MGMT	Yes	No	No	Yes
ORD_LC_AMENDMENTS	Yes	No	No	No
ORDCUST	Yes	No	No	No
ORDCUST_DETAIL	Yes	Yes	No	Yes
ORDDIST_ITEM_TEMP	No	No	No	Yes
ORDHEAD	Yes	No	No	No
ORDHEAD_REV	No	Yes	No	No
ORDLC	Yes	No	Yes	No
ORDLOC	Yes	No	Yes	No
ORDLOC_DISCOUNT	No	No	No	Yes
ORDLOC_DISCOUNT_TEMP	No	No	No	Yes
ORDLOC_EXP_TEMP	No	No	No	Yes
ORDLOC_REV	No	Yes	No	No
ORDLOC_TEMP	No	No	No	Yes
ORDLOC_WKSHT	Yes	No	No	Yes
ORDSKU	Yes	No	No	No
ORDSKU HTS	Yes	No	No	No
ORDSKU HTS_ASSESS_TEMP	No	No	No	Yes
ORDSKU HTS_TEMP	No	No	No	Yes
ORDSKU_REV	No	Yes	No	No
ORDSKU_TEMP	No	No	No	Yes
OTB	Yes	Yes	Yes	No
OTB_CASCADE_STG	No	Yes	No	No
PARTNER_ORG_UNIT	Yes	No	No	No
POP_TERMS_DEF	No	No	No	Yes
POP_TERMS_FULFILLMENT	No	No	No	Yes
PROCUREMENT_UNIT_OPTIONS	Yes	No	No	No
REPL_RESULTS_TEMP	No	No	No	Yes
REQ_DOC	Yes	Yes	No	No
REQ_DOC_TEMP	No	No	No	Yes
REV_ORDERS	No	No	No	Yes
RTM_UNIT_OPTIONS	Yes	No	No	No
STORE	Yes	No	No	No
SUP_AVAIL	Yes	No	Yes	No
SUPS	Yes	No	No	No
SYSTEM_CONFIG_OPTIONS	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
TAX_CALC_EVENT	Yes	Yes	No	No
TAX_EVENT_RUN_TYPE	Yes	No	No	No
TICKET_REQUEST	No	Yes	No	No
TIMELINE_TEMP	No	No	No	Yes
TRANSIT_TIMES	Yes	No	No	No
V_PACKSKU_QTY	Yes	No	No	No
WH	Yes	No	No	No
WO_DETAIL_TEMP	No	No	No	Yes
WO_HEAD_TEMP	No	No	No	Yes
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Cancel Purchase Orders

This section describes the Cancel Purchase Orders service.

Functional Area

Procurement

Business Overview

This service cancels a list of purchase order.

Service Type

Post

ReST URL

/PurchaseOrders/recent/cancelPurchaseOrders?orderNumbers={orderNumbers}

Input Parameters

Order Number -Required-comma separated list

Output

RpoStatusRDO

Parameter Name	Data Type
successOrdersCount	BigDecimal
successOrdersTable	List<BigDecimal>
failOrdersCount	BigDecimal
failOrdersTable	List<RpoFailRDO>

RpoFailRDO

Parameter Name	Data Type
orderNumber	BigDecimal
errorMessage	String

JSON Structure:

```
{
  "successOrdersCount": 0,
  "successOrdersTable": [],
  "failOrdersCount": 2,
  "failOrdersTable": [
    {
      "orderNumber": 123,
      "errorMessage": "Invalid Reason Code.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    },
    {
      "orderNumber": 987,
      "errorMessage": "Invalid Order Number.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_DETAIL	Yes	No	Yes	No
ALLOC_HEADER	Yes	No	Yes	No
APPT_DETAIL	Yes	No	No	No
APPT_HEAD	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
DEAL_CALC_QUEUE	No	No	No	Yes
ORDHEAD	Yes	No	Yes	No
ORDLOC	Yes	No	Yes	No
OTB	No	No	Yes	No
SHIPMENT	Yes	No	Yes	No
SHIPSKU	Yes	No	Yes	No
SYSTEM_OPTIONS	Yes	No	No	No
WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Reject Purchase Orders

This section describes the Reject Purchase Orders service.

Functional Area

Procurement

Business Overview

This service rejects a list of purchase order.

Service Type

Post

ReST URL

/PurchaseOrders/recent/rejectPurchaseOrders?orderNumbers={orderNumbers}

Input Parameters

Order Numbers - Required - comma separated list

Output

RpoStatusRDO

Parameter Name	Data Type
successOrdersCount	BigDecimal
successOrdersTable	List<BigDecimal>
failOrdersCount	BigDecimal
failOrdersTable	List<RpoFailRDO>

RpoFailRDO

Parameter Name	Data Type
orderNumber	BigDecimal
errorMessage	String

JSON Structure:

```
{
  "successOrdersCount": 0,
  "successOrdersTable": [],
  "failOrdersCount": 2,
  "failOrdersTable": [
    {
      "orderNumber": 123,
      "errorMessage": " Invalid Order Number.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
}
```

```

    {
      "orderNumber": 987,
      "errorMessage": "Invalid Order Number.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_DETAIL	No	No	Yes	No
ALLOC_HEADER	Yes	No	Yes	No
CONTRACT_DETAIL	Yes	No	Yes	No
CONTRACT_HEADER	Yes	No	Yes	No
ITEM_MASTER	Yes	No	No	No
LC_ORDAPPLY	No	No	No	Yes
ORDHEAD	Yes	No	Yes	No
ORDLOC	Yes	No	No	No
OTB	No	No	Yes	No
SHIPMENT	Yes	No	No	No
SHIPSKU	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Purchase Orders - Update Purchase Order Dates

This section describes the Update Purchase Orders Date service.

Functional Area

Procurement

Business Overview

This service update list of purchase order dates. If no date is sent or sent as null then the assumption is there is no change on the current record date.

Service Type

Post

ReST URL

/PurchaseOrders/recent/updatePurchaseOrderDate?
notBeforeDate={notBeforeDate}&notAfterDate={notAfterDate}&otbEowDate={otbEowDate}&orderNumbers={orderNumbers}

Input Parameters

- Order Numbers - Required - comma separated list
- Not Before Date - Optional - in a long format
- Not After Date - Optional - in a long format
- OTB EWO Date - Optional - in a long format

Output

RpoStatusRDO

Parameter Name	Data Type
successOrdersCount	BigDecimal
successOrdersTable	List<BigDecimal>
failOrdersCount	BigDecimal
failOrdersTable	List<RpoFailRDO>

RpoFailRDO

Parameter Name	Data Type
orderNumber	BigDecimal
errorMessage	String

JSON Structure:

```
{
  "successOrdersCount": 0,
  "successOrdersTable": [],
  "failOrdersCount": 2,
  "failOrdersTable": [
    {
      "orderNumber": 123,
      "errorMessage": "Invalid Reason Code.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    },
    {
      "orderNumber": 987,
      "errorMessage": "Invalid Reason Code.",
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
```

```

    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ALLOC_HEADER	No	No	Yes	No
CONTRACT_HEADER	Yes	No	No	No
DEAL_HEAD	Yes	No	Yes	No
ORDHEAD	Yes	No	Yes	No
OTB	No	No	Yes	No
SHIPMENT	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Purchase Order Download Service

The following services are included in this functional area:

- [Get Purchase Order Details](#)
- [Get Purchase Order Details by Order Number](#)

Get Purchase Order Details

Endpoints

MerchIntegrations/services/procurement/order

MerchIntegrations/services/procurement/order/{orderNo}

Functional Area

Purchase Orders

Business Overview

This service publishes external purchase orders to downstream consuming systems. The payload includes order header and line-location detail, along with supplier-sourcing, location, and customer-order enrichment. The JSON view returns orders only when ORDHEAD.STATUS is A, W, S, or C, ORDHEAD.Orig_Approval_Date is not null, and ORDHEAD.ORDER_TYPE <> 'DSD', so DSD orders and orders that have never crossed an approval event are not included. Worksheet and submitted statuses can appear in the payload only after the order has already been approved at least once, and orderLocationType is derived as S, W, or SW depending on whether the order locations are stores, warehouses, or a mix of both.

The publication for order based on order detail line is only when unit cost or estimated in stock date or quantity ordered is changed. This service will not publish orders when shipping or receiving of the order has started.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ORDHEAD, ORDLOC

Additional business tables: ORDSKU, ORDCUST, ORDCUST_DETAIL, ITEM_SUPP_COUNTRY, STORE, WH, DEPS, BUYER, RPM_PROMO, CODE_DETAIL, V_TERMS_HEAD_TL, PO_TYPE, V_PARTNER_TL, V_OUTLOC_TL, ORDHEAD_CFA_EXT, ORDSKU_CFA_EXT, ORDLOC_CFA_EXT

JSON cache table: MERCHAPI_EXT_ORDER

JSON generation view: V_MERCHAPI_EXT_ORDER_JSON

Builds the header from ORDHEAD, the detail array from ORDLOC and ORDSKU, OMS linkage from ORDCUST and ORDCUST_DETAIL, item-supplier sourcing attributes from ITEM_SUPP_COUNTRY, location enrichment from WH and STORE, descriptive lookup values from DEPS, BUYER, RPM_PROMO, CODE_DETAIL, V_TERMS_HEAD_TL, PO_TYPE, V_PARTNER_TL, and V_OUTLOC_TL, and header, item, and item-location custom flex attributes from MERCHAPI_EXT_ORDER_CFA, MERCHAPI_EXT_ORDER_ITEM_CFA, and MERCHAPI_EXT_ORD_ITEM_LOC_CFA.

The view only publishes orders where ORDHEAD.STATUS is one of A, W, S, or C, ORDHEAD.ORIG_APPROVAL_DATE is not null, and ORDHEAD.ORDER_TYPE <> 'DSD'.

This means DSD orders and orders that have never crossed an approval event are not returned, even though worksheet and submitted statuses can still appear later in the payload after the order has already been approved once.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_ORDER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the order no longer exists in V_MERCHAPI_EXT_ORDER_JSON.

ICL (Integration Change Log) table: ICL_RMS_ORDER (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on ORDHEAD, ORDLOC, ORDHEAD_CFA_EXT, ORDLOC_CFA_EXT, and ORDSKU_CFA_EXT.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ORDER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ORDER_ADHOC_PROCESS ->
MERCHAPI_EXT_ORDER_ADHOC_JOB

Webhook configuration api name: procurement/order

Special handling

Additional GET query parameters: supplier, orderOriginatedInd, notBeforeDate, notAfterDate, importOrderInd, locType.

These filters are not backed by dedicated cache-table indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. The available configuration for this service should therefore be reviewed and validated against the customer's projected production data profile, including expected volume, message size, and access pattern, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through API_NO_OF_THREADS, and commit sizing is controlled through API_COMMIT_MAX_COUNTER.

For initial seeding, thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the message from multiple base and supporting tables. Common starting points for large-volume seeding are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment.

API_COMMIT_MAX_COUNTER should be sized together with the thread setting. Typical values can range from about 100 to 1,000 rows, with higher values such as 10,000 used in some cases after testing. Where payloads are larger or the source SQL is heavier because of the number of contributing tables, a lower thread count with a higher commit value may perform better.

For delta processing, the same parameters apply. The seeding commit setting is usually a reasonable starting point, but where qualifying business events can create short-term spikes, the commit value may need to remain on the lower side so backlog drains in smaller units of work.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, DROP_INDEX_ON_REFRESH can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes.

This can improve data-load throughput, but the overall refresh plan should also account for the additional time needed for index rebuild after the seeding step finishes.

GATHER_STATS_ON_REFRESH should normally remain enabled unless testing shows a need to change it.

REST fetch considerations

REST page size should be selected based on average message size so that the average response remains below about 10 MB per call. Common page-limit values are 500, 1,000, 5,000, or 10,000 rows, and very small limits, especially below 100, should generally be avoided.

For full downloads, do not provide the since query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the LAST_UPDATE_DATETIME index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping BUILD_HISTOGRAM enabled allows the service to support recovery through smaller, more selective polling windows using the optimalBefore query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides optimalBefore and leaves before empty. The client then uses the returned before timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups LAST_UPDATE_DATETIME values into buckets of 10,000 rows in MERCHAPI_TIMESTAMP_HISTOGRAM using the shared HISTOGRAM_ROW_COUNT configuration.

Service Type

GET

ReST URL

MerchIntegrations/services/procurement/order

MerchIntegrations/services/procurement/order/{orderNo}

Input Parameters

Parameter Name	Required	Data Type	Description
since	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
before	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)
supplier	No	String (10)	Supplier
orderOriginatedInd	No	Number	Order Originated Indicator
notBeforeDate	No	String	Not Before Date (Example: 2023-12-31)
notAfterDate	No	String	Not After Date (Example: 2023-12-31)

Parameter Name	Required	Data Type	Description
importOrderInd	No	String	Import Order Indicator - must be Y or N
locType	No	String	Order Location Type - must be S, W or SW
offsetkey	No	String (12)	Offset key (Order Number)
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1202 Input Parameter for "MerchIntegrations/services/procurement/order/{orderNo}"

Parameter Name	Required	Data Type	Description
orderNo	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-1203 PageResultsItemsMerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1204 MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for purchase order details.
docType	No	String (1)	This field indicates the type of document this message is for. Valid value is P for purchase order.
orderNo	No	Number(12,0)	This field contains the number to uniquely identify an order within the system.
status	No	String (1)	This field indicates the current status of the order.
orderType	No	String (3)	This field indicates the type of order.
orderTypeDescription	No	String (250)	This field contains the order type description.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dept	No	Number(4,0)	This field contains the department number for orders limited to a single department and will be null for orders involving items in more than one department.
deptName	No	String (120)	This field contains the name of the department.
buyer	No	Number(4,0)	This field contains the number associated with the buyer for the order.
buyerName	No	String (120)	This field contains the name of the authorized buyer.
supplier	No	Number(10,0)	This field contains the vendor number who will provide the merchandise specified in the order.
promotion	No	Number(10,0)	This field contains the promotion number associated with the order to provide a link between the order dialog and the promotions dialog.
promotionDescription	No	String (1000)	This field contains a description of the promotion. This value comes from Pricing.
qualityControlInd	No	String (1)	This field determines whether or not quality control will be required when items for this order are received.
notBeforeDate	No	date	This field contains the first date that delivery of the order will be accepted.
notAfterDate	No	date	This field contains the last date that delivery of the order will be accepted.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbEowDate	No	date	This field contains the OTB budget bucket the order amount should be placed into.
earliestShipDate	No	date	This field contains the earliest date when the items on the purchase order can be shipped by the supplier. This represents the earliest ship date of all the items on the order.
latestShipDate	No	date	This field contains the date after which the items on the purchase order can no longer be shipped by the supplier. This represents the greatest latest ship date of all the items on the order.
closeDate	No	date	This field contains the date when the order is closed.
terms	No	String (15)	This field is an indicator identifying the sales terms for the order. These terms specify when payment is due and if any discounts exist for early payment.
termsCode	No	String (240)	This field contains the description associated with the terms code.
freightTerms	No	String (30)	This field contains an indicator that references what freight terms are related to the order. Valid values are in the FREIGHT_TERMS table.
paymentMethod	No	String (6)	This field indicates how the purchase order will be paid.
paymentMethodDescription	No	String (250)	This field contains the description of the payment method.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order.
backhaulTypeDescription	No	String (250)	This field contains the description of the backhaul type.
backhaulAllowance	No	Number(20,4)	This field contains the backhaul allowance value.
shipMethod	No	String (6)	This field contains the method used to ship the items on the purchase order from the country of origin to the country of import.
shipMethodDescription	No	String (250)	This field contains the description for the shipping method.
purchaseType	No	String (6)	This field indicates what is included in the supplier's cost of the item.
purchaseTypeDescription	No	String (250)	This field contains the description for the purchase type.
shipPayMethod	No	String (2)	This field contains the code indicating the payment terms for freight charges associated with the order.
shipPayMethodDescription	No	String (250)	This field contains the description of the given shipment payment method.
fobTransactionResponsibility	No	String (2)	This field contains the code indicating the type of the location that is responsible for the transportation of the order.
fobTransactionResponsibilityCodeDescription	No	String (250)	This field contains the description of the FOB transaction responsibility code.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fobTransactionResponsibilityDescription	No	String (250)	This field contains a user-entered field describing the code for the location responsible for the transportation of the order.
fobTitlePass	No	String (2)	This field contains an indicator used to determine where the title for goods is passed from the vendor to the purchaser. Examples include city, factory, or origin.
fobTitlePassCodeDescription	No	String (250)	This field contains the description of the FOB title pass code.
fobTitlePassDescription	No	String (250)	This field contains a user-entered field describing the code where the title of the merchandise is to be passed. It could be a city name, factory name, or place of origin.
vendorOrderNo	No	String (15)	This field contains the vendor's unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated with a Merchandising order when the order is created online.
exchangeRate	No	Number(20,10)	This field contains the rate of exchange used for the purchase order between the order and primary currencies.
factory	No	String (10)	This field contains the factory at which the items on the purchase order are made. This field is only available when the system is running Import functionality.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
factoryDescription	No	String (240)	This field contains the description of the factory.
agent	No	String (10)	This field contains the agent that is used in the purchase of the items on the purchase order. This field is only available when the system is running Import functionality.
agentDesc	No	String (240)	This field contains the description of the agent.
dischargePort	No	String (5)	This field contains the port at which the items on the purchase order will enter the country of import. This field is only available when the system is running Import functionality.
dischargePortDescription	No	String (150)	This field contains the description of the discharge port.
ladingPort	No	String (5)	This field contains the port from which the items on the purchase order are shipped. This field is only available when the system is running Import functionality.
ladingPortDescription	No	String (150)	This field contains the description of the lading port.
freightContractNo	No	String (10)	This field contains the number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
poType	No	String (4)	This field contains the value associated with the PO_TYPE for the order. Valid values are in the PO_TYPE table.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
poTypeDescription	No	String (120)	This field contains the description of the PO type.
preMarkInd	No	String (1)	This field indicates whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores.
currencyCode	No	String (3)	This field contains the currency code for the order. When creating an order within Merchandising, this field will always default to the currency of the supplier, but it can be overwritten.
contractNo	No	Number(6,0)	This field contains the contract number associated with this order.
pickupLocation	No	String (250)	This field contains the location at which the order will be picked up, if the order is a pickup order.
pickupNo	No	String (25)	This field contains the reference number for the pickup order.
pickupDate	No	date	This field contains the date when the order can be picked up from the supplier. This field is only required if the purchase type of the order is Pickup.
appointmentDatetime	No	date	This field contains the date of the receiving appointment at the warehouse.
comments	No	String (2000)	This field contains the miscellaneous comments attached to the purchase order.
writtenDate	No	date	This field specifies the date the order was created within the system.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderOriginatedInd	No	Number(2,0)	This field indicates where the order originated.
originalApprovalDate	No	date	This field specifies the date that the order was originally approved.
originalApprovalId	No	String (254)	This field indicates where the order was approved. It will be the user ID of the person approving the order for all orders manually approved online and will contain Replen for orders automatically approved in the replenishment process.
ediSentInd	No	String (1)	This field indicates whether or not the order has been sent via EDI.
ediOrderInd	No	String (1)	This field indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction.
importOrderInd	No	String (1)	This field indicates if the purchase order is an import order.
importCountryId	No	String (3)	This field specifies the identifier of the country into which the items on the order are being imported.
partnerType1	No	String (6)	This field contains partner 1 type.
partner1	No	String (10)	This field contains additional partner 1.
partnerType2	No	String (6)	This field contains partner 2 type.
partner2	No	String (10)	This field contains additional partner 2.
partnerType3	No	String (6)	This field contains partner 3 type.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
partner3	No	String (10)	This field contains additional partner 3.
importType	No	String (1)	This field specifies the importer/exporter assigned to the supplier of the purchase order.
importId	No	Number(10,0)	This field identifies the importer/exporter assigned to the supplier. This should reference the WH.WH column.
clearingZoneId	No	String (5)	This field contains the clearing zone ID.
routingLocationId	No	String (5)	This field contains the routing location for the import order.
deliverySupplier	No	Number(10,0)	This field holds the supplier or supplier site from where the goods are delivered.
wfOrderNo	No	Number(10,0)	This field contains the franchise order number for which the purchase order was created.
exchangeRateInd	No	String (1)	This field contains the exchange rate indicator on a document.
customerOrderNo	No	String (48)	This field holds the master customer order number from the Order Management System.
fullfillmentOrderNo	No	String (48)	This field holds the number from the Order Management System (OMS) related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS.
customerOrderInd	No	String (1)	This field indicates whether the purchase order is linked to a customer order.

Table 5-1204 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the purchase order.
details	No	Collection of Object	This list holds details for the purchase order.
cacheTimestamp	No	dateTime	This field specifies the date and time when the record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1205 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1206 OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	This field contains the unique alphanumeric value to identify the item included in the purchase order.
referenceItem	No	String (25)	This field contains the unique alphanumeric value to identify the reference item.
physicalLocationType	No	String (1)	This field contains the type of location in the location field.
physicalLocation	No	Number(10,0)	This field contains the actual physical location that the item will be ordered to. This field may contain a store or a physical warehouse.
virtualWarehouse	No	Number(10,0)	This field contains the virtual warehouse for the order location when the physical location type is warehouse.
physicalStoreType	No	String (6)	This field indicates the store type of the physical location store. This is only populated if physical location type is store.
physicalStockholdingInd	No	String (1)	This field indicates whether the physical location store is stockholding or not.
physicalQuantityOrdered	No	Number(12,4)	This field indicates the total quantity ordered to the physical location.
unitCost	No	Number(20,4)	This field contains the unit cost for the item/ location on the order, including discounts (deal/bracket/ allowance), but not landed cost components. This field is stored in order currency.

Table 5-1206 (Cont.) OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierInitialUnitCost	No	Number(20,4)	Initial supplier unit cost of the item/ location on the order, before deals were applied. When discounts must be recalculated on an approved order, this cost will be used as the starting point. This value is captured as the current item/ location/supplier/ origin country cost at the time of order approval.
unitRetail	No	Number(20,4)	This field contains the current retail price in local currency for the SKU at this location.
prescaledQuantity	No	Number(12,4)	This field contains the order quantity (system generated or manually generated) for the item/location before order scaling processing was performed.
originalReplenishment Quantity	No	Number(12,4)	This field contains the original order quantity generated by the replenishment process.
costSource	No	String (4)	This field indicates the basis for the cost of the item/location on the order and whether deals or brackets have been applied.
lastRoundedQuantity	No	Number(12,4)	This field holds the most recent system rounded order quantity value for the line item.
lastGroupRoundedQuantity	No	Number(12,4)	This field holds the last system rounded order quantity for the rounding group the location is a part of.

Table 5-1206 (Cont.) OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountryId	No	String (3)	This field contains the identifier of the country from which the item is being sourced.
supplierPackSize	No	Number(12,4)	This field contains the supplier pack size for this order/item.
earliestShipDate	No	date	This field contains the earliest date that the item can be shipped by the supplier.
latestShipDate	No	date	This field contains the date after which the item can not be shipped by the supplier.
pickupLocation	No	String (250)	This field contains the location at which the order will be picked up, if the order is a pickup order.
pickupNo	No	String (25)	This field is not used in Merchandising.
packingMethod	No	String (6)	This field indicates whether the packing method of the item in the container is flat or hanging. Valid values are defined under code type PKMT.
roundLevel	No	String (6)	This field will be used to determine how order quantities will be rounded to case, layer, and pallet.
transferOrderLinkId	No	Number(12,0)	This field contains a reference number to link the item on the transfer to any purchase orders that have been created to allow the from location (i.e. warehouse) on the transfer to fulfill the transfer quantity to the to location (i.e. store) on the transfer.

Table 5-1206 (Cont.) OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
estimatedInStockDate	No	date	This field contains the date that the item on the PO is expected to be available to ship from the PO location to another location.
transactionUom	No	String (4)	This field holds the original transaction unit of measure to be passed on to the store system. It is only used for customer direct orders coming from the order management system. All quantities on the payload are expressed in the item's standard unit of measure.
itemLineNo	No	Number(10,0)	This field indicates the item line number from customer orders.
orderItemCreateDateTime	No	dateTime	This column holds the record order detail item creation date and time (UTC).
orderItemUpdateDateTime	No	dateTime	This column holds the date and time when the order detail item was last updated (UTC).
orderItemLocCreateDateTime	No	dateTime	This column holds the record order detail item location creation date and time (UTC).
orderItemLocUpdateDateTime	No	dateTime	This column holds the date and time when the order detail item location was last updated (UTC).
orderItemCustomFlexAttribute	No	Collection of Object	This list holds custom flex attributes for purchase order item details.
orderItemLocationCustomFlexAttribute	No	Collection of Object	This list holds custom flex attributes for purchase order item location details.

Table 5-1207 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "docType": "P",
      "orderNo": 464690012,
      "status": "A",
      "orderType": "N/B",
      "orderTypeDescription": null,
      "dept": 1,
      "deptName": null,
      "buyer": 1000,
      "buyerName": null,
      "supplier": 2400,
      "promotion": 41,
      "promotionDescription": null,
      "qualityControlInd": "Y",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "earliestShipDate": "2001-12-31",
      "latestShipDate": "2001-12-31",
      "closeDate": "2001-12-31",
      "terms": "02",
      "termsCode": null,
      "freightTerms": "03",
      "paymentMethod": "LC",
      "paymentMethodDescription": null,
      "backhaulType": null,
      "backhaulTypeDescription": null,
      "backhaulAllowance": 1.0,
      "shipMethod": "32",
      "shipMethodDescription": null,
      "purchaseType": "FOB",
      "purchaseTypeDescription": null,
      "shipPayMethod": "P0",
      "shipPayMethodDescription": null,
      "fobTransactionResponsibility": "CA",
    }
  ]
}
```

```

"fobTransactionResponsibilityCodeDescription": null,
"fobTransactionResponsibilityDescription": "US",
"fobTitlePass": "CC",
"fobTitlePassCodeDescription": null,
"fobTitlePassDescription": "US",
"vendorOrderNo": "9987001",
"exchangeRate": 2,
"factory": "100",
"factoryDescription": null,
"agent": "007",
"agentDesc": null,
"dischargePort": "480",
"dischargePortDescription": null,
"ladingPort": "7",
"ladingPortDescription": null,
"freightContractNo": "111",
"poType": "4000",
"poTypeDescription": null,
"preMarkInd": "Y",
"currencyCode": "USD",
"contractNo": 1,
"pickupLocation": null,
"pickupNo": null,
"pickupDate": "2001-12-31",
"appointmentDatetime": "2001-12-31",
"comments": "Create Order",
"writtenDate": "2001-12-31",
"orderOriginatedInd": 2,
"originalApprovalDate": "2001-12-31",
"originalApprovalId": null,
"ediSentInd": null,
"ediOrderInd": "Y",
"importOrderInd": null,
"importCountryId": "US",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocationId": null,
"deliverySupplier": 1,
"wfOrderNo": 1,
"exchangeRateInd": null,
"customerOrderNo": null,
"fullfillmentOrderNo": null,
"customerOrderInd": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",

```

```

"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
},
],
"details": [
{
"item": "103900095",
"referenceItem": null,
"physicalLocationType": "W",
"physicalLocation": 6000,
"virtualWarehouse": 1,
"physicalStoreType": null,
"physicalStockholdingInd": null,
"physicalQuantityOrdered": 25,
"unitCost": 45.2,
"supplierInitialUnitCost": 1.0,
"unitRetail": 1.0,
"prescaledQuantity": 1.0,
"originalReplenishmentQuantity": 1.0,
"costSource": null,
"lastRoundedQuantity": 1.0,
"lastGroupRoundedQuantity": 1.0,
"originCountryId": "IN",
"supplierPackSize": 1,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"pickupLocation": null,
"pickupNo": null,
"packingMethod": null,
"roundLevel": null,
"transferOrderLinkId": 1,
"estimatedInStockDate": "2001-12-31",
"transactionUom": null,
"itemLineNo": 1,
"orderItemCreateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemUpdateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemLocCreateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemLocUpdateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemCustomFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"orderItemLocationCustomFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
}
]
}

```

```

]
}
],
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
}
],
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORDHEAD_CFA_EXT	Yes	No	No	No
ORDSKU_CFA_EXT	Yes	No	No	No
ORDLOC_CFA_EXT	Yes	No	No	No
MERCHAPI_EXT_ORDER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ORDER_JSON	Yes	No	No	No
ICL_RMS_ORDER	Yes	No	Yes	Yes

Get Purchase Order Details by Order Number

Endpoints

MerchIntegrations/services/procurement/order

MerchIntegrations/services/procurement/order/{orderNo}

Functional Area

Purchase Orders

Business Overview

This service publishes external purchase orders to downstream consuming systems. The payload includes order header and line-location detail, along with supplier-sourcing, location, and customer-order enrichment. The JSON view returns orders only when ORDHEAD.STATUS is A, W, S, or C, ORDHEAD.Orig_Approval_Date is not null, and ORDHEAD.ORDER_TYPE <> 'DSD', so DSD orders and orders that have never crossed an approval event are not included. Worksheet and submitted statuses can appear in the payload only after the order has already been approved at least once, and orderLocationType is derived as S, W, or SW depending on whether the order locations are stores, warehouses, or a mix of both.

The publication for order based on order detail line is only when unit cost or estimated in stock date or quantity ordered is changed. This service will not publish orders when shipping or receiving of the order has started.

Technical Details

This service follows the Publish API cache architecture in which the REST service serves pre-generated JSON from an integration cache table that is initially seeded and then maintained through asynchronous delta processing. For overall enablement, scheduling, and guidance on how publish api works and how webhook for this api can be implemented to stream the data out, refer to the white paper *Publish APIs - Customer Enablement Overview*.

Core business tables: ORDHEAD, ORDLOC

Additional business tables: ORDSKU, ORDCUST, ORDCUST_DETAIL, ITEM_SUPP_COUNTRY, STORE, WH, DEPS, BUYER, RPM_PROMO, CODE_DETAIL, V_TERMS_HEAD_TL, PO_TYPE, V_PARTNER_TL, V_OUTLOC_TL, ORDHEAD_CFA_EXT, ORDSKU_CFA_EXT, ORDLOC_CFA_EXT

JSON cache table: MERCHAPI_EXT_ORDER

JSON generation view: V_MERCHAPI_EXT_ORDER_JSON

Builds the header from ORDHEAD, the detail array from ORDLOC and ORDSKU, OMS linkage from ORDCUST and ORDCUST_DETAIL, item-supplier sourcing attributes from ITEM_SUPP_COUNTRY, location enrichment from WH and STORE, descriptive lookup values from DEPS, BUYER, RPM_PROMO, CODE_DETAIL, V_TERMS_HEAD_TL, PO_TYPE, V_PARTNER_TL, and V_OUTLOC_TL, and header, item, and item-location custom flex attributes from MERCHAPI_EXT_ORDER_CFA, MERCHAPI_EXT_ORDER_ITEM_CFA, and MERCHAPI_EXT_ORD_ITEM_LOC_CFA.

The view only publishes orders where ORDHEAD.STATUS is one of A, W, S, or C, ORDHEAD.Orig_Approval_Date is not null, and ORDHEAD.ORDER_TYPE <> 'DSD'.

This means DSD orders and orders that have never crossed an approval event are not returned, even though worksheet and submitted statuses can still appear later in the payload after the order has already been approved once.

Initial Seeding and post enablement rebuild/repair

Initial seeding and rebuild pull published rows from V_MERCHAPI_EXT_ORDER_JSON. No additional filter criteria were identified beyond the cache-build logic.

Mark cache rows deleted when the order no longer exists in V_MERCHAPI_EXT_ORDER_JSON.

ICL (Integration Change Log) table: ICL_RMS_ORDER (ICL consumer name = MERCHAPI)

ICL entries are created using triggers on ORDHEAD, ORDLOC, ORDHEAD_CFA_EXT, ORDLOC_CFA_EXT, and ORDSKU_CFA_EXT.

Batch Configuration. Asynchronous delta processing is supported using Background engine and also via POM scheduling. Choose one.

Process configuration name: API_EXT_ORDER

To enable via Background engine set BACKGROUND_RUN_FLAG to Y and review the BACKGROUND_RUN_INTERVAL_SECOND parameter.

POM Batch schedule process and job:
MERCHAPI_DELTA_EXT_ORDER_ADHOC_PROCESS ->
MERCHAPI_EXT_ORDER_ADHOC_JOB

Webhook configuration api name: procurement/order

Special handling

Additional GET query parameters: supplier, orderOriginatedInd, notBeforeDate, notAfterDate, importOrderInd, locType.

These filters are not backed by dedicated cache-table indexes. At high volume, use them only as additional criteria on top of a small-window delta fetch and not for full downloads.

Performance Considerations

For customers expecting high to very high publish volume from this API, sizing and operational tuning should be addressed early in the implementation lifecycle. Initial cache seeding, ongoing delta processing, full extracts, and incremental downloads or webhook-driven delivery have different runtime characteristics and should be evaluated separately. The available configuration for this service should therefore be reviewed and validated against the customer's projected production data profile, including expected volume, message size, and access pattern, with the key performance-related configuration for this service covering the following:

Multi-threading and commit control

This service supports multi-threaded execution for both initial seeding and delta processing. Thread count is controlled through API_NO_OF_THREADS, and commit sizing is controlled through API_COMMIT_MAX_COUNTER.

For initial seeding, thread and commit settings should be chosen together based on expected row volume, average message size, and the complex SQL query required to build the message from multiple base and supporting tables. Common starting points for large-volume seeding are 8, 16, or 32 threads. Values of 48 or 64 threads should be considered only after testing and with awareness of competing workload in the environment.

API_COMMIT_MAX_COUNTER should be sized together with the thread setting. Typical values can range from about 100 to 1,000 rows, with higher values such as 10,000 used in some cases after testing. Where payloads are larger or the source SQL is heavier because of the number of contributing tables, a lower thread count with a higher commit value may perform better.

For delta processing, the same parameters apply. The seeding commit setting is usually a reasonable starting point, but where qualifying business events can create short-term spikes, the commit value may need to remain on the lower side so backlog drains in smaller units of work.

Dropping indexes and constraints during seeding

This option applies only to initial seeding, including truncate-and-load execution. When thread and commit settings are high, concurrent inserts can increase index maintenance overhead and slow the load.

To reduce that overhead, `DROP_INDEX_ON_REFRESH` can be enabled so indexes and constraints are dropped before seeding and rebuilt after the load completes.

This can improve data-load throughput, but the overall refresh plan should also account for the additional time needed for index rebuild after the seeding step finishes.

`GATHER_STATS_ON_REFRESH` should normally remain enabled unless testing shows a need to change it.

REST fetch considerations

REST page size should be selected based on average message size so that the average response remains below about 10 MB per call. Common page-limit values are 500, 1,000, 5,000, or 10,000 rows, and very small limits, especially below 100, should generally be avoided.

For full downloads, do not provide the `since` query parameter so the database can favor the primary-key access path. For delta downloads, use a small time window, typically a few minutes and at most a few hours, so the eligible row set on the `LAST_UPDATE_DATETIME` index remains small.

Wider delta windows, especially multi-day windows that can arise during outages or data recovery, generally do not perform well. For such scenarios, keeping `BUILD_HISTOGRAM` enabled allows the service to support recovery through smaller, more selective polling windows using the `optimalBefore` query parameter.

In this catch-up pattern, for requests up to 30 days in the past, the client provides `optimalBefore` and leaves `before` empty. The client then uses the returned `before` timestamp to continue polling incrementally until the backlog is cleared, and then resumes the normal short-window delta pattern. Hourly histogram processing groups `LAST_UPDATE_DATETIME` values into buckets of 10,000 rows in `MERCHAPI_TIMESTAMP_HISTOGRAM` using the shared `HISTOGRAM_ROW_COUNT` configuration.

Service Type

GET

ReST URL

`MerchIntegrations/services/procurement/order`

`MerchIntegrations/services/procurement/order/{orderNo}`

Input Parameters

Parameter Name	Required	Data Type	Description
<code>since</code>	No	String	Since Date (Example: 2010-12-31T23:59:59.999Z)
<code>before</code>	No	String	Before Date (Example: 2040-12-31T23:59:59.999Z)

Parameter Name	Required	Data Type	Description
supplier	No	String (10)	Supplier
orderOriginatedInd	No	Number	Order Orginated Indicator
notBeforeDate	No	String	Not Before Date (Example: 2023-12-31)
notAfterDate	No	String	Not After Date (Example: 2023-12-31)
importOrderInd	No	String	Import Order Indicator - must be Y or N
locType	No	String	Order Location Type - must be S, W or SW
offsetkey	No	String (12)	Offset key (Order Number)
limit	No	String (6)	Pagination limit. Default value is 1000.
optimalBefore	No	String	Enables histogram-based calculation of the before timestamp when before is not provided. Valid values are Y or N. Default value is N.
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Table 5-1208 Input Parameter for "MerchIntegrations/services/procurement/order/{orderNo}"

Parameter Name	Required	Data Type	Description
orderNo	Yes	Number	
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)

Table 5-1208 (Cont.) Input Parameter for "MerchIntegrations/services/procurement/order/{orderNo}"

Parameter Name	Required	Data Type	Description
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Output

Table 5-1209 PageResultsItemsMerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count for pagination supported request
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1210 MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
action	No	String (6)	Action for purchase order details.
docType	No	String (1)	This field indicates the type of document this message is for. Valid value is P for purchase order.
orderNo	No	Number(12,0)	This field contains the number to uniquely identify an order within the system.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	No	String (1)	This field indicates the current status of the order.
orderType	No	String (3)	This field indicates the type of order.
orderTypeDescription	No	String (250)	This field contains the order type description.
dept	No	Number(4,0)	This field contains the department number for orders limited to a single department and will be null for orders involving items in more than one department.
deptName	No	String (120)	This field contains the name of the department.
buyer	No	Number(4,0)	This field contains the number associated with the buyer for the order.
buyerName	No	String (120)	This field contains the name of the authorized buyer.
supplier	No	Number(10,0)	This field contains the vendor number who will provide the merchandise specified in the order.
promotion	No	Number(10,0)	This field contains the promotion number associated with the order to provide a link between the order dialog and the promotions dialog.
promotionDescription	No	String (1000)	This field contains a description of the promotion. This value comes from Pricing.
qualityControlInd	No	String (1)	This field determines whether or not quality control will be required when items for this order are received.
notBeforeDate	No	date	This field contains the first date that delivery of the order will be accepted.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
notAfterDate	No	date	This field contains the last date that delivery of the order will be accepted.
otbEowDate	No	date	This field contains the OTB budget bucket the order amount should be placed into.
earliestShipDate	No	date	This field contains the earliest date when the items on the purchase order can be shipped by the supplier. This represents the earliest ship date of all the items on the order.
latestShipDate	No	date	This field contains the date after which the items on the purchase order can no longer be shipped by the supplier. This represents the greatest latest ship date of all the items on the order.
closeDate	No	date	This field contains the date when the order is closed.
terms	No	String (15)	This field is an indicator identifying the sales terms for the order. These terms specify when payment is due and if any discounts exist for early payment.
termsCode	No	String (240)	This field contains the description associated with the terms code.
freightTerms	No	String (30)	This field contains an indicator that references what freight terms are related to the order. Valid values are in the FREIGHT_TERMS table.
paymentMethod	No	String (6)	This field indicates how the purchase order will be paid.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
paymentMethodDescription	No	String (250)	This field contains the description of the payment method.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order.
backhaulTypeDescription	No	String (250)	This field contains the description of the backhaul type.
backhaulAllowance	No	Number(20,4)	This field contains the backhaul allowance value.
shipMethod	No	String (6)	This field contains the method used to ship the items on the purchase order from the country of origin to the country of import.
shipMethodDescription	No	String (250)	This field contains the description for the shipping method.
purchaseType	No	String (6)	This field indicates what is included in the supplier's cost of the item.
purchaseTypeDescription	No	String (250)	This field contains the description for the purchase type.
shipPayMethod	No	String (2)	This field contains the code indicating the payment terms for freight charges associated with the order.
shipPayMethodDescription	No	String (250)	This field contains the description of the given shipment payment method.
fobTransactionResponsibility	No	String (2)	This field contains the code indicating the type of the location that is responsible for the transportation of the order.
fobTransactionResponsibilityCodeDescription	No	String (250)	This field contains the description of the FOB transaction responsibility code.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fobTransactionResponsibilityDescription	No	String (250)	This field contains a user-entered field describing the code for the location responsible for the transportation of the order.
fobTitlePass	No	String (2)	This field contains an indicator used to determine where the title for goods is passed from the vendor to the purchaser. Examples include city, factory, or origin.
fobTitlePassCodeDescription	No	String (250)	This field contains the description of the FOB title pass code.
fobTitlePassDescription	No	String (250)	This field contains a user-entered field describing the code where the title of the merchandise is to be passed. It could be a city name, factory name, or place of origin.
vendorOrderNo	No	String (15)	This field contains the vendor's unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated with a Merchandising order when the order is created online.
exchangeRate	No	Number(20,10)	This field contains the rate of exchange used for the purchase order between the order and primary currencies.
factory	No	String (10)	This field contains the factory at which the items on the purchase order are made. This field is only available when the system is running Import functionality.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
factoryDescription	No	String (240)	This field contains the description of the factory.
agent	No	String (10)	This field contains the agent that is used in the purchase of the items on the purchase order. This field is only available when the system is running Import functionality.
agentDesc	No	String (240)	This field contains the description of the agent.
dischargePort	No	String (5)	This field contains the port at which the items on the purchase order will enter the country of import. This field is only available when the system is running Import functionality.
dischargePortDescription	No	String (150)	This field contains the description of the discharge port.
ladingPort	No	String (5)	This field contains the port from which the items on the purchase order are shipped. This field is only available when the system is running Import functionality.
ladingPortDescription	No	String (150)	This field contains the description of the lading port.
freightContractNo	No	String (10)	This field contains the number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
poType	No	String (4)	This field contains the value associated with the PO_TYPE for the order. Valid values are in the PO_TYPE table.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
poTypeDescription	No	String (120)	This field contains the description of the PO type.
preMarkInd	No	String (1)	This field indicates whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores.
currencyCode	No	String (3)	This field contains the currency code for the order. When creating an order within Merchandising, this field will always default to the currency of the supplier, but it can be overwritten.
contractNo	No	Number(6,0)	This field contains the contract number associated with this order.
pickupLocation	No	String (250)	This field contains the location at which the order will be picked up, if the order is a pickup order.
pickupNo	No	String (25)	This field contains the reference number for the pickup order.
pickupDate	No	date	This field contains the date when the order can be picked up from the supplier. This field is only required if the purchase type of the order is Pickup.
appointmentDatetime	No	date	This field contains the date of the receiving appointment at the warehouse.
comments	No	String (2000)	This field contains the miscellaneous comments attached to the purchase order.
writtenDate	No	date	This field specifies the date the order was created within the system.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderOriginatedInd	No	Number(2,0)	This field indicates where the order originated.
originalApprovalDate	No	date	This field specifies the date that the order was originally approved.
originalApprovalId	No	String (254)	This field indicates where the order was approved. It will be the user ID of the person approving the order for all orders manually approved online and will contain Replen for orders automatically approved in the replenishment process.
ediSentInd	No	String (1)	This field indicates whether or not the order has been sent via EDI.
ediOrderInd	No	String (1)	This field indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction.
importOrderInd	No	String (1)	This field indicates if the purchase order is an import order.
importCountryId	No	String (3)	This field specifies the identifier of the country into which the items on the order are being imported.
partnerType1	No	String (6)	This field contains partner 1 type.
partner1	No	String (10)	This field contains additional partner 1.
partnerType2	No	String (6)	This field contains partner 2 type.
partner2	No	String (10)	This field contains additional partner 2.
partnerType3	No	String (6)	This field contains partner 3 type.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
partner3	No	String (10)	This field contains additional partner 3.
importType	No	String (1)	This field specifies the importer/exporter assigned to the supplier of the purchase order.
importId	No	Number(10,0)	This field identifies the importer/exporter assigned to the supplier. This should reference the WH.WH column.
clearingZoneId	No	String (5)	This field contains the clearing zone ID.
routingLocationId	No	String (5)	This field contains the routing location for the import order.
deliverySupplier	No	Number(10,0)	This field holds the supplier or supplier site from where the goods are delivered.
wfOrderNo	No	Number(10,0)	This field contains the franchise order number for which the purchase order was created.
exchangeRateInd	No	String (1)	This field contains the exchange rate indicator on a document.
customerOrderNo	No	String (48)	This field holds the master customer order number from the Order Management System.
fullfillmentOrderNo	No	String (48)	This field holds the number from the Order Management System (OMS) related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS.
customerOrderInd	No	String (1)	This field indicates whether the purchase order is linked to a customer order.

Table 5-1210 (Cont.) MerchApiOrder - Object. See list of elements for detail

Element Name	Required	Data Type	Description
createDateTime	No	dateTime	This column holds the record creation date and time (UTC).
updateDateTime	No	dateTime	This column holds the date and time when the record was last updated (UTC).
customFlexAttribute	No	Collection of Object	This list holds custom flex attributes for the purchase order.
details	No	Collection of Object	This list holds details for the purchase order.
cacheTimestamp	No	dateTime	This field specifies the date and time when the record was fetched into the cache for web service publication (UTC).
cacheCreateTimestamp	No	dateTime	This field specifies the date and time when the record was first created for web service publication (UTC).

Table 5-1211 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	No	String (30)	This field holds the custom flex attribute name.
value	No	String (250)	This field holds the numeric or character value of the custom flex attribute.
valueDate	No	date	This field holds the date value of the custom flex attribute.
createDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was created (UTC).
updateDateTime	No	dateTime	This field holds the date and time when the custom flex attribute was last updated (UTC).

Table 5-1212 OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	This field contains the unique alphanumeric value to identify the item included in the purchase order.
referenceItem	No	String (25)	This field contains the unique alphanumeric value to identify the reference item.
physicalLocationType	No	String (1)	This field contains the type of location in the location field.
physicalLocation	No	Number(10,0)	This field contains the actual physical location that the item will be ordered to. This field may contain a store or a physical warehouse.
virtualWarehouse	No	Number(10,0)	This field contains the virtual warehouse for the order location when the physical location type is warehouse.
physicalStoreType	No	String (6)	This field indicates the store type of the physical location store. This is only populated if physical location type is store.
physicalStockholdingInd	No	String (1)	This field indicates whether the physical location store is stockholding or not.
physicalQuantityOrdered	No	Number(12,4)	This field indicates the total quantity ordered to the physical location.
unitCost	No	Number(20,4)	This field contains the unit cost for the item/ location on the order, including discounts (deal/bracket/allowance), but not landed cost components. This field is stored in order currency.

Table 5-1212 (Cont.) OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplierInitialUnitCost	No	Number(20,4)	Initial supplier unit cost of the item/ location on the order, before deals were applied. When discounts must be recalculated on an approved order, this cost will be used as the starting point. This value is captured as the current item/ location/supplier/ origin country cost at the time of order approval.
unitRetail	No	Number(20,4)	This field contains the current retail price in local currency for the SKU at this location.
prescaledQuantity	No	Number(12,4)	This field contains the order quantity (system generated or manually generated) for the item/location before order scaling processing was performed.
originalReplenishment Quantity	No	Number(12,4)	This field contains the original order quantity generated by the replenishment process.
costSource	No	String (4)	This field indicates the basis for the cost of the item/location on the order and whether deals or brackets have been applied.
lastRoundedQuantity	No	Number(12,4)	This field holds the most recent system rounded order quantity value for the line item.
lastGroupRoundedQuantity	No	Number(12,4)	This field holds the last system rounded order quantity for the rounding group the location is a part of.

Table 5-1212 (Cont.) OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
originCountryId	No	String (3)	This field contains the identifier of the country from which the item is being sourced.
supplierPackSize	No	Number(12,4)	This field contains the supplier pack size for this order/item.
earliestShipDate	No	date	This field contains the earliest date that the item can be shipped by the supplier.
latestShipDate	No	date	This field contains the date after which the item can not be shipped by the supplier.
pickupLocation	No	String (250)	This field contains the location at which the order will be picked up, if the order is a pickup order.
pickupNo	No	String (25)	This field is not used in Merchandising.
packingMethod	No	String (6)	This field indicates whether the packing method of the item in the container is flat or hanging. Valid values are defined under code type PKMT.
roundLevel	No	String (6)	This field will be used to determine how order quantities will be rounded to case, layer, and pallet.
transferOrderLinkId	No	Number(12,0)	This field contains a reference number to link the item on the transfer to any purchase orders that have been created to allow the from location (i.e. warehouse) on the transfer to fulfill the transfer quantity to the to location (i.e. store) on the transfer.

Table 5-1212 (Cont.) OrderDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
estimatedInStockDate	No	date	This field contains the date that the item on the PO is expected to be available to ship from the PO location to another location.
transactionUom	No	String (4)	This field holds the original transaction unit of measure to be passed on to the store system. It is only used for customer direct orders coming from the order management system. All quantities on the payload are expressed in the item's standard unit of measure.
itemLineNo	No	Number(10,0)	This field indicates the item line number from customer orders.
orderItemCreateDateTime	No	dateTime	This column holds the record order detail item creation date and time (UTC).
orderItemUpdateDateTime	No	dateTime	This column holds the date and time when the order detail item was last updated (UTC).
orderItemLocCreateDateTime	No	dateTime	This column holds the record order detail item location creation date and time (UTC).
orderItemLocUpdateDateTime	No	dateTime	This column holds the date and time when the order detail item location was last updated (UTC).
orderItemCustomFlexAttribute	No	Collection of Object	This list holds custom flex attributes for purchase order item details.
orderItemLocationCustomFlexAttribute	No	Collection of Object	This list holds custom flex attributes for purchase order item location details.

Table 5-1213 LinkItems - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetch and contains the offsetkey which uniquely identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "action": "INSERT",
      "docType": "P",
      "orderNo": 464690012,
      "status": "A",
      "orderType": "N/B",
      "orderTypeDescription": null,
      "dept": 1,
      "deptName": null,
      "buyer": 1000,
      "buyerName": null,
      "supplier": 2400,
      "promotion": 41,
      "promotionDescription": null,
      "qualityControlInd": "Y",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "earliestShipDate": "2001-12-31",
      "latestShipDate": "2001-12-31",
      "closeDate": "2001-12-31",
      "terms": "02",
      "termsCode": null,
      "freightTerms": "03",
      "paymentMethod": "LC",
      "paymentMethodDescription": null,
      "backhaulType": null,
      "backhaulTypeDescription": null,
      "backhaulAllowance": 1.0,
      "shipMethod": "32",
      "shipMethodDescription": null,
      "purchaseType": "FOB",
      "purchaseTypeDescription": null,
      "shipPayMethod": "P0",
      "shipPayMethodDescription": null,
      "fobTransactionResponsibility": "CA",
    }
  ]
}
```

```

"fobTransactionResponsibilityCodeDescription": null,
"fobTransactionResponsibilityDescription": "US",
"fobTitlePass": "CC",
"fobTitlePassCodeDescription": null,
"fobTitlePassDescription": "US",
"vendorOrderNo": "9987001",
"exchangeRate": 2,
"factory": "100",
"factoryDescription": null,
"agent": "007",
"agentDesc": null,
"dischargePort": "480",
"dischargePortDescription": null,
"ladingPort": "7",
"ladingPortDescription": null,
"freightContractNo": "111",
"poType": "4000",
"poTypeDescription": null,
"preMarkInd": "Y",
"currencyCode": "USD",
"contractNo": 1,
"pickupLocation": null,
"pickupNo": null,
"pickupDate": "2001-12-31",
"appointmentDatetime": "2001-12-31",
"comments": "Create Order",
"writtenDate": "2001-12-31",
"orderOriginatedInd": 2,
"originalApprovalDate": "2001-12-31",
"originalApprovalId": null,
"ediSentInd": null,
"ediOrderInd": "Y",
"importOrderInd": null,
"importCountryId": "US",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocationId": null,
"deliverySupplier": 1,
"wfOrderNo": 1,
"exchangeRateInd": null,
"customerOrderNo": null,
"fullfillmentOrderNo": null,
"customerOrderInd": null,
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z",
"customFlexAttribute": [
{
"name": "attribute1",
"value": "value1",

```

```

"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
},
],
"details": [
{
"item": "103900095",
"referenceItem": null,
"physicalLocationType": "W",
"physicalLocation": 6000,
"virtualWarehouse": 1,
"physicalStoreType": null,
"physicalStockholdingInd": null,
"physicalQuantityOrdered": 25,
"unitCost": 45.2,
"supplierInitialUnitCost": 1.0,
"unitRetail": 1.0,
"prescaledQuantity": 1.0,
"originalReplenishmentQuantity": 1.0,
"costSource": null,
"lastRoundedQuantity": 1.0,
"lastGroupRoundedQuantity": 1.0,
"originCountryId": "IN",
"supplierPackSize": 1,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"pickupLocation": null,
"pickupNo": null,
"packingMethod": null,
"roundLevel": null,
"transferOrderLinkId": 1,
"estimatedInStockDate": "2001-12-31",
"transactionUom": null,
"itemLineNo": 1,
"orderItemCreateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemUpdateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemLocCreateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemLocUpdateDateTime": "2001-12-31T23:59:59.000Z",
"orderItemCustomFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
],
"orderItemLocationCustomFlexAttribute": [
{
"name": "attribute1",
"value": "value1",
"valueDate": "2001-12-31",
"createDateTime": "2001-12-31T23:59:59.000Z",
"updateDateTime": "2001-12-31T23:59:59.000Z"
}
]
}
]
}

```

```

]
},
"cacheTimestamp": "2001-12-31T23:59:59.000Z",
"cacheCreateTimestamp": "2001-12-31T23:59:59.000Z"
},
"hasMore": true,
"limit": 1000,
"count": 1000,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?
limit=1000&since=1970-01-01T00:00:00.001Z&before=2024-01-01T00:00:00.001Z&offs
etkey=1234,XYZ",
"rel": "next"
}
]
}

```

Table Impact

The list of table includes the data pre-processing for this service along with the table/views queried during the ReST call.

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCHAPI_CONFIG	Yes	No	No	No
MERCHAPI_CONSUMER_CONFIG	Yes	No	No	No
MERCHAPI_ASYNC_REQUEST	Yes	No	No	No
MERCH_BATCH_PARAM	Yes	No	No	No
ORDHEAD	Yes	No	No	No
ORDHEAD_CFA_EXT	Yes	No	No	No
ORDSKU_CFA_EXT	Yes	No	No	No
ORDLOC_CFA_EXT	Yes	No	No	No
MERCHAPI_EXT_ORDER	Yes	Yes	Yes	Yes
V_MERCHAPI_EXT_ORDER_JSON	Yes	No	No	No
ICL_RMS_ORDER	Yes	No	Yes	Yes

Purchase Order Upload Service

The following services are included in this functional area:

- [Create Pre-issued Order Numbers](#)
- [Create Purchase Order Details](#)
- [Create Purchase Order Expenses](#)
- [Create Purchase Order HTS](#)
- [Create Purchase Order HTS Assessments](#)
- [Create Purchase Order Letters of Credit](#)
- [Create Purchase Orders](#)

- [Delete Purchase Order Details](#)
- [Delete Purchase Order Expenses](#)
- [Delete Purchase Order HTS](#)
- [Delete Purchase Order HTS Assessments](#)
- [Delete Purchase Order Letters of Credit](#)
- [Delete Purchase Orders](#)
- [Update Purchase Order Details](#)
- [Update Purchase Order Expenses](#)
- [Update Purchase Order HTS](#)
- [Update Purchase Order HTS Assessments](#)
- [Update Purchase Order Letters of Credit](#)
- [Update Purchase Orders](#)

Create Pre-issued Order Numbers

Functional Area

Purchase Orders

Business Overview

This webservice generates pre-issued order numbers for the calling application with a set expiry date to be used in orders that will later be created and integrated to Merchandising.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrder/preIssuedOrderNumber/create

Input Payload Details

Table 5-1214 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Number (10)	Contains the supplier to which the order number is assigned.
quantity	Yes	Number (4)	Contains the number of order numbers to be generated.
expiryDays	Yes	Number (4)	Contains the number of days before the pre-issued order numbers expire.

Sample Input Message

```
{
  "supplier": 2400,
```

```
"quantity": 5,  
"expiryDays": 14  
}
```

Response Code: 200 (Success)

Table 5-1215 CreateResponse - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNumbers	No	Collection of Object	References a collection of order numbers generated.

Table 5-1216 OrderNumbers - Object. See list of elements for detail

Element Name	Required	Data Type	Description
supplier	No	Number (10)	Contains the supplier to which the order number is assigned.
orderNo	Yes	Number (12)	Contains the order number generated in Merchandising.
expiryDate	Yes	date	The date when the pre-issued order number will expire.

Sample Response Message

```
{  
  "orderNumbers": [  
    {  
      "supplier": 2400,  
      "orderNo": 90989990,  
      "expiryDate": "2001-12-31"  
    }  
  ]  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```



```
    ],  
    "businessError": [  
      "Error message"  
    ]  
  }  
}
```

Create Purchase Order Details

Functional Area

Purchase Orders

Business Overview

This webservice is used to create new detail level attributes of the purchase order such as the addition of items and locations in order to keep Merchandising in sync with an external system that is responsible for maintaining purchase orders. Order details can be created for orders in Approved, Worksheet, Submitted or Closed status. The only information needed at the header level is the order number, which if not provided, will cause the message to be rejected. If an item/location combination has the Purchase Type as 'Consignment / Concession' for the primary supplier, then orders can only be created from the primary supplier for that location. In case of an order being raised for a parent item which is owned at a location but has consignment / concession child items under it, and the supplier is a non-primary one, then the child items which are not owned at that location will be omitted from the order.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrder/details/create

Input Payload Details

Table 5-1217 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	

Table 5-1218 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.

Table 5-1218 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
details	No	Collection of Object	This references the order detail node.
itemDetails	No	Collection of Object	This references the collection of items in the order.

Table 5-1219 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	An approved, transaction level item. If a pack, it must be an orderable pack (with an order_as_type of 'P'). This is required if the ref_item field is not populated. An item/location may be added to the order in worksheet or approved status.
location	Yes	Number (10)	An active store or warehouse. If multichannel is on, and a warehouse is being order to, a virtual warehouse is expected. An item/location may be added to the order in worksheet or approved status.
locationType	No	String (1)	The location type of the location. Valid values are 'S' store and 'W' warehouse. This field cannot be modified.
unitCost	No	Number (20,4)	The cost of the item from the supplier in the order's currency. This can be modified if the status is worksheet or approved, but not if any portion of the item's order quantity is in transit or has been received. If it is not specified on the message, it will default from the item supplier cost for this location.
referenceItem	No	String (25)	The id of a reference item which can be used instead of using the item field. If the item field is not populated this field is required. A ref item/location may be added to the order in worksheet or approved status.
originCountry	No	String (3)	The identifier of the country from which the item is being sourced. This field cannot be modified.
supplierPackSize	No	Number (12,4)	The supplier pack size for the item on the order. This may be updated in worksheet or approved status.

Table 5-1219 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
quantityOrdered	No	Number (12,4)	The quantity ordered of item. This can be updated in worksheet or approved status to adjust the quantity of an item/location or cancel an item/location from the order.
cancelInd	No	String (1)	Indicates if the detail record's quantity should be cancelled. Valid values are 'Y'es and 'N'o.
reinstateInd	No	String (1)	Indicates if a detail record which was previously cancelled should be reinstated. This will reinstate in the order quantity the amount in the cancelled bucket. Valid values are 'Y'es and 'N'o.
deliveryDate	No	date	The date by which goods are to be delivered. There can be multiple item-locations on the order with different delivery dates.
quantityCancelled	No	Number (12,4)	This field contains the quantity that was left to be ordered when the line item was cancelled.
cancelCode	No	String (1)	This field contains the reason that the line item was cancelled. This field is required if a line item is cancelled.
estimatedInstockDate	No	date	Date that the item on the PO is expected to be available to ship from the PO location to another location. It is calculated as the Order Creation or Approval Date + Supplier Lead Time + Transit Days between the Supplier and the PO/location + Item/Warehouse Inbound Handling Days (if PO location is a warehouse).
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes for the order/location. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Table 5-1220 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1221 ItemDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	An approved, transaction level item. If a pack, it must be an orderable pack (with an order_as_type of 'P'). This is required if the ref_item field is not populated. An item/location may be added to the order in worksheet or approved status.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes for ordsku. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL

Sample Input Message

```
{
  "items": [
    {
      "orderNo": 464690012,
      "approvedBy": null,
      "dataLoadingDestination": "RMS",
      "details": [
        {
          "item": "103900095",
          "location": 6000,
          "locationType": "W",
          "unitCost": 45.2,
          "referenceItem": null,
          "originCountry": "IN",
          "supplierPackSize": 1,
          "quantityOrdered": 25,
          "cancelInd": null,
          "reinstateInd": null,
          "deliveryDate": "2001-12-31",
          "quantityCancelled": null,
          "cancelCode": null,
          "estimatedInstockDate": "2001-12-31",
          "earliestShipDate": "2001-12-31",
          "latestShipDate": "2001-12-31",

```

```

        "customFlexAttribute": [
          {
            "name": "REVIEW_DATE",
            "value": null,
            "valueDate": "2001-12-31"
          }
        ]
      },
    ],
    "itemDetails": [
      {
        "item": "103900095",
        "customFlexAttribute": [
          {
            "name": "REVIEW_DATE",
            "value": null,
            "valueDate": "2001-12-31"
          }
        ]
      }
    ]
  }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS",
  "message": "Service call is successful"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}

```

```
    ]  
  }
```

Create Purchase Order Expenses

Functional Area

Purchase Orders

Business Overview

This webservice is used to create expense information on the purchase order within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. Expenses may be created alongside the creation of a new order or added to an existing order that has location records defined. Internally, the service sets the status of the order to Worksheet in order to accommodate the modification of expense information for purchase orders in the system that have been submitted or approved. Validation of the new expense information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrder/expenses/create

Input Payload Details

Table 5-1222 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order expenses.

Table 5-1223 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.

Table 5-1223 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.

Table 5-1223 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.

Table 5-1223 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	String (6)	This field indicates what's included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.

Table 5-1223 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.

Table 5-1223 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
expenses	No	Collection of Object	References the expense associated with a particular item/location of the purchase order.

Table 5-1224 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1225 Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains alphanumeric value that uniquely identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
location	Yes	Number (10)	This field contains the location that item will be ordered to. This field may contain a store or a warehouse.
locationType	Yes	String (1)	This field contains the type of location in the location field. Valid values are: S - Store W - Warehouse.
component	Yes	String (10)	This field contains a user specified code representing a specific expense found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the expense.
calculationBasis	No	String (1)	This field contains the basis the expense will be calculated against if there is no CVB code attached to the expense. Valid values are S (Supplier) or O (Order). If the Cost Basis is S then when calculating the expense, the items supplier cost will be used. If the Cost Basis is O then the order unit cost of the item on the order will be used to calculate the expense. The code type for this field is EXCB.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base of the component (either value or specific).
componentCurrency	Yes	String (3)	This field contains the currency the expense or assessment is to be entered in. If the Calculation Basis is V (Value), then comp_currency will be NULL.
exchangeRate	Yes	Number (20,10)	This field contains the exchange rate of the individual expense.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.

Table 5-1225 (Cont.) Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-1225 (Cont.) Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.
inAlc	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
```

```
"currencyCode": "USD",
"terms": "02",
"notBeforeDate": "2001-12-31",
"notAfterDate": "2001-12-31",
"otbEowDate": "2001-12-31",
"dept": null,
"status": "A",
"includeOnOrderInd": "Y",
"writtenDate": "2001-12-31",
"origin": "2",
"ediPoInd": "Y",
"preMarkInd": "Y",
"approvedBy": null,
"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
```

```

    "earliestShipDate": "2001-12-31",
    "latestShipDate": "2001-12-31",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ],
    "expenses": [
      {
        "item": "103900095",
        "packItem": null,
        "location": 10001,
        "locationType": "W",
        "component": "OCFRT",
        "computationValueBase": null,
        "calculationBasis": null,
        "componentRate": 17,
        "componentCurrency": "INR",
        "exchangeRate": 29,
        "perCount": 1,
        "perCountUom": "M3",
        "nominationFlag1": "N",
        "inDuty": "N",
        "nominationFlag3": "N",
        "inExpense": "+",
        "inAlc": "N"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {

```



```

        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
    }
  ]
}

```

Create Purchase Order HTS

Functional Area

Purchase Orders

Business Overview

This webservice is used to create Harmonized Tariff Schedules and related assessments within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order has not yet been submitted or approved. HTS and Assessments may be created together with the creation of a new import order or added to an existing order. Internally, the service sets the status of the order to Worksheet in order to accommodate the addition of HTS information for purchase orders in the system that have been submitted or approved. Validation of the new HTS information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrder/hts/create

Input Payload Details

Table 5-1226 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order HTS records.

Table 5-1227 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.

Table 5-1227 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.

Table 5-1227 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.

Table 5-1227 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	String (6)	This field indicates what's included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.

Table 5-1227 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.

Table 5-1227 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1228 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1229 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains an unique alphanumeric value that identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
hts	Yes	String (25)	This field contains an identifier for the Harmonized Tariff Schedule code.
status	Yes	String (1)	This field contains a flag to indicate the status of the Purchase Order/Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
originCountry	Yes	String (3)	This field contains the country of manufacture when the HTS tracking level is M. When the HTS tracking level is S, this stores the country of sourcing.
importCountry	No	String (3)	This field contains the country that the item will be imported in to.
assessments	No	Collection of Object	References the HTS code assessments associated with a particular item of the purchase order.

Table 5-1230 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains a user specified code representing the specific assessment component found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the assessment.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount in the currency defined for the assessment (usually the import country's currency) if the Calculation Basis is Specific.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.

Table 5-1230 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-1230 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.
inAlc	Yes	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
    }
  ]
}
```

```
"status": "A",
"includeOnOrderInd": "Y",
"writtenDate": "2001-12-31",
"origin": "2",
"ediPoInd": "Y",
"preMarkInd": "Y",
"approvedBy": null,
"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
```

```

        "valueDate": "2001-12-31"
      }
    ],
    "hts": [
      {
        "item": "103900095",
        "packItem": null,
        "hts": "901411112",
        "status": "W",
        "originCountry": "IN",
        "importCountry": "US",
        "assessments": [
          {
            "component": "DTYCAUS",
            "computationValueBase": null,
            "componentRate": 12,
            "perCount": 1,
            "perCountUom": "EA",
            "nominationFlag1": "N",
            "inDuty": "+",
            "nominationFlag3": "-",
            "inExpense": "N",
            "inAlc": "N"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

```
}  
]  
}
```

Create Purchase Order HTS Assessments

Functional Area

Purchase Orders

Business Overview

This webservice is used to create Harmonized Tariff Schedules assessments within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. Assessments may be created together with the creation of a new import order or added to an existing order. Internally, the service sets the status of the order to Worksheet in order to accommodate the addition of HTS information for purchase orders in the system that have been submitted or approved. Validation of the new and updated information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrder/hts/assessments/create

Input Payload Details

Table 5-1231 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order HTS assessment records.

Table 5-1232 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.

Table 5-1232 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.

Table 5-1232 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.

Table 5-1232 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	String (6)	This field indicates whats included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.

Table 5-1232 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.

Table 5-1232 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1233 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1234 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains an unique alphanumeric value that identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
hts	Yes	String (25)	This field contains an identifier for the Harmonized Tariff Schedule code.
status	Yes	String (1)	This field contains a flag to indicate the status of the Purchase Order/Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
originCountry	Yes	String (3)	This field contains the country of manufacture when the HTS tracking level is M. When the HTS tracking level is S, this stores the country of sourcing.
importCountry	No	String (3)	This field contains the country that the item will be imported in to.
assessments	No	Collection of Object	References the HTS code assessments associated with a particular item of the purchase order.

Table 5-1235 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains a user specified code representing the specific assessment component found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the assessment.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount in the currency defined for the assessment (usually the import country's currency) if the Calculation Basis is Specific.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.

Table 5-1235 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-1235 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.
inAlc	Yes	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
    }
  ]
}
```

```
"status": "A",
"includeOnOrderInd": "Y",
"writtenDate": "2001-12-31",
"origin": "2",
"ediPoInd": "Y",
"preMarkInd": "Y",
"approvedBy": null,
"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
```

```

        "valueDate": "2001-12-31"
      }
    ],
    "hts": [
      {
        "item": "103900095",
        "packItem": null,
        "hts": "9014111112",
        "status": "W",
        "originCountry": "IN",
        "importCountry": "US",
        "assessments": [
          {
            "component": "DTYCAUS",
            "computationValueBase": null,
            "componentRate": 12,
            "perCount": 1,
            "perCountUom": "EA",
            "nominationFlag1": "N",
            "inDuty": "+",
            "nominationFlag3": "-",
            "inExpense": "N",
            "inAlc": "N"
          }
        ]
      }
    ]
  }
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

```
}  
  ]  
}
```

Create Purchase Order Letters of Credit

Functional Area

Purchase Orders

Business Overview

This webservice is used to create letter of credit information within Merchandising when the order payment type is Letter of Credit in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order has not yet been submitted or approved. For an existing order that has a payment type of Letter of Credit, the modification of header level order information is not necessary; however header level fields, particularly the payment type can be updated in case a letter of credit needs to be attached to the worksheet order that has an existing payment type other than Letter of Credit. Internally, the service sets the status of the order to Worksheet in order to accommodate the modification of letter of credit information for purchase orders in the system that have been submitted or approved. Validation of updated information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrder/lettersOfCredit/create

Input Payload Details

Table 5-1236 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of letter of credit records.

Table 5-1237 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.

Table 5-1237 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.

Table 5-1237 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.

Table 5-1237 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	String (6)	This field indicates what's included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.

Table 5-1237 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.

Table 5-1237 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
letterOfCredit	No	Record	References the letter of credit for the purchase order.

Table 5-1238 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1239 LetterOfCredit - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lcReferenceId	No	Number (8)	This field contains the ID of the LC to which the order is attached.
lcGroupId	No	String (10)	This field contains an ID that can be used to group Orders together so that they can be attached to a single Letter of Credit.
applicant	Yes	String (10)	This field contains the ID of the applicant being used for the LC process.
beneficiary	Yes	Number (10)	This field contains the ID of the beneficiary being used for the LC process.
merchandiseDescription	Yes	String (2000)	This field holds a description of the merchandise on the order.
transshipmentInd	Yes	String (1)	This field indicates whether or not the LC to which the order is attached will allow the order to be transshipped. Valid values are 'Y'es and 'N'o.
partialShipmentInd	Yes	String (1)	This field indicates whether or not the LC to which the order is attached will allow the order to be partially shipped. Valid values are 'Y'es and 'N'o.
lcAttachedInd	Yes	String (1)	This field indicates whether or not the order has been attached to a Letter of Credit. Valid values are Y and N.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
      "status": "A",
      "includeOnOrderInd": "Y",
      "writtenDate": "2001-12-31",
      "origin": "2",
      "ediPoInd": "Y",
      "preMarkInd": "Y",
      "approvedBy": "RMS_ADMIN",
    }
  ]
}
```

```

"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"letterOfCredit": {
  "lcReferenceId": 10015000,
  "lcGroupId": null,
  "applicant": "500",

```

```

        "beneficiary": 2400,
        "merchandiseDescription": "Bath Essentials",
        "transshipmentInd": "Y",
        "partialShipmentInd": "Y",
        "lcAttachedInd": "Y"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Create Purchase Orders

Functional Area

Purchase Orders

Business Overview

This webservice is used to create purchase orders in order to keep Merchandising in sync with an external system that is responsible for maintaining purchase orders. It creates orders in Worksheet, Approved or Closed status. It also creates other data associated with a purchase order, including letter of credit, expenses, harmonized tariff schedules (HTS) and assessments, and custom flex attributes (CFAS). It also will apply rounding rules, default inventory management parameters, apply bracket costs, update open to buy buckets, and insert a record into the deals queue for deals to be applied to the order, if applicable. If the location specified

on a purchase order is a franchise store, a corresponding franchise order is also created along with the PO.

It is assumed that the source of the orders sent in this service is not the supplier, as vendor managed inventory (VMI) POs can be sent using the Upload Purchase Order and Purchase Order Change Acknowledgements from Suppliers to Merchandising (ediupack) batch upload. It also does not support creating customer order POs or contract POs. Customer order POs are assumed to be sent using the Customer Order Fulfillment Subscription API and contract orders are created using replenishment processes in Merchandising or manually using the UI.

Service Type

POST

ReST URL

MerchIntegrations/services/purchaseOrders/create

Input Payload Details

Table 5-1240 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	Collection of orders.

Table 5-1241 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.
currencyCode	No	String (3)	The code of the order's currency. If this is not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If this is not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.

Table 5-1241 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
exchangeRate	No	Number (20,10)	The rate of exchange for the PO used between the order and primary currencies. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If this is not defined, the default is STG.

Table 5-1241 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.
purchaseType	No	String (6)	This field indicates what is included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).

Table 5-1241 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.

Table 5-1241 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to an Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.

Table 5-1241 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH column with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
fileId	No	String (20)	In order to support supplier pooling via the buyer worksheet, the recommended order quantities for the individual suppliers are linked together for logistical purposes while generating orders from the worksheet based on this identifier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
details	No	Collection of Object	This references the order detail node.
itemDetails	No	Collection of Object	This references the collection of items in the order.
letterOfCredit	No	Record	References the letter of credit for the purchase order.
expenses	No	Collection of Object	References the expense associated with a particular item/location of the purchase order.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1242 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	An approved, transaction level item. If a pack, it must be an orderable pack (with an order_as_type of 'P'). This is required if the ref_item field is not populated. An item/location may be added to the order in worksheet or approved status.
location	Yes	Number (10)	An active store or warehouse. If multichannel is on, and a warehouse is being order to, a virtual warehouse is expected. An item/location may be added to the order in worksheet or approved status.
locationType	No	String (1)	The location type of the location. Valid values are 'S' store and 'W' warehouse. This field cannot be modified.
unitCost	No	Number (20,4)	The cost of the item from the supplier in the order's currency. This can be modified if the status is worksheet or approved, but not if any portion of the item's order quantity is in transit or has been received. If it is not specified on the message, it will default from the item supplier cost for this location.
referenceItem	No	String (25)	The id of a reference item which can be used instead of using the item field. If the item field is not populated this field is required. A ref item/location may be added to the order in worksheet or approved status.
originCountry	No	String (3)	The identifier of the country from which the item is being sourced. This field cannot be modified.
supplierPackSize	No	Number (12,4)	The supplier pack size for the item on the order. This may be updated in worksheet or approved status.
quantityOrdered	No	Number (12,4)	The quantity ordered of item. This can be updated in worksheet or approved status to adjust the quantity of an item/location or cancel an item/location from the order.
cancelInd	No	String (1)	Indicates if the detail record's quantity should be cancelled. Valid values are 'Y'es and 'N'o.
reinstateInd	No	String (1)	Indicates if a detail record which was previously cancelled should be reinstated. This will reinstate in the order quantity the amount in the cancelled bucket. Valid values are 'Y'es and 'N'o.

Table 5-1242 (Cont.) Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
deliveryDate	No	date	The date by which goods are to be delivered. There can be multiple item-locations on the order with different delivery dates.
quantityCancelled	No	Number (12,4)	This field contains the quantity that was left to be ordered when the line item was cancelled.
cancelCode	No	String (1)	This field contains the reason that the line item was cancelled. This field is required if a line item is cancelled.
estimatedInstockDate	No	date	Date that the item on the PO is expected to be available to ship from the PO location to another location. It is calculated as the Order Creation or Approval Date + Supplier Lead Time + Transit Days between the Supplier and the PO/location + Item/Warehouse Inbound Handling Days (if PO location is a warehouse).
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes at the order header level. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-1243 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1244 ItemDetails - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	An approved, transaction level item. If a pack, it must be an orderable pack (with an order_as_type of 'P'). This is required if the ref_item field is not populated. An item/location may be added to the order in worksheet or approved status.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured, named attribute is missing from the input collection, its value will be set to NULL.

Table 5-1245 LetterOfCredit - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lcReferenceId	No	Number (8)	This field contains the ID of the LC to which the order is attached.
lcGroupId	No	String (10)	This field contains an ID that can be used to group Orders together so that they can be attached to a single Letter of Credit.
applicant	Yes	String (10)	This field contains the ID of the applicant being used for the LC process.
beneficiary	Yes	Number (10)	This field contains the ID of the beneficiary being used for the LC process.
merchandiseDescription	Yes	String (2000)	This field holds a description of the merchandise on the order.
transshipmentInd	Yes	String (1)	This field indicates whether or not the LC to which the order is attached will allow the order to be transshipped. Valid values are 'Y'es and 'N'o.
partialShipmentInd	Yes	String (1)	This field indicates whether or not the LC to which the order is attached will allow the order to be partially shipped. Valid values are 'Y'es and 'N'o.
lcAttachedInd	Yes	String (1)	This field indicates whether or not the order has been attached to a Letter of Credit. Valid values are Y and N.

Table 5-1246 Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains alphanumeric value that uniquely identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
location	Yes	Number (10)	This field contains the location that item will be ordered to. This field may contain a store or a warehouse.
locationType	Yes	String (1)	This field contains the type of location in the location field. Valid values are: S - Store W - Warehouse.
component	Yes	String (10)	This field contains a user specified code representing a specific expense found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the expense.
calculationBasis	No	String (1)	This field contains the basis the expense will be calculated against if there is no CVB code attached to the expense. Valid values are S (Supplier) or O (Order). If the Cost Basis is S then when calculating the expense, the items supplier cost will be used. If the Cost Basis is O then the order unit cost of the item on the order will be used to calculate the expense. The code type for this field is EXCB.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base of the component (either value or specific).
componentCurrency	Yes	String (3)	This field contains the currency the expense or assessment is to be entered in. If the Calculation Basis is V (Value), then comp_currency will be NULL.
exchangeRate	No	Number (20,10)	This field contains the exchange rate of the individual expense.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.

Table 5-1246 (Cont.) Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.

Table 5-1246 (Cont.) Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.
inAlc	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.

Table 5-1247 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains an unique alphanumeric value that identifies the item.

Table 5-1247 (Cont.) Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
hts	Yes	String (25)	This field contains an identifier for the Harmonized Tariff Schedule code.
status	Yes	String (1)	This field contains a flag to indicate the status of the Purchase Order/Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
originCountry	Yes	String (3)	This field contains the country of manufacture when the HTS tracking level is M. When the HTS tracking level is S, this stores the country of sourcing.
importCountry	No	String (3)	This field contains the country that the item will be imported in to.
assessments	No	Collection of Object	References the HTS code assessments associated with a particular item of the purchase order.

Table 5-1248 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains a user specified code representing the specific assessment component found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the assessment.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount in the currency defined for the assessment (usually the import country's currency) if the Calculation Basis is Specific.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.

Table 5-1248 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.

Table 5-1248 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.
inAlc	Yes	String (1)	See description for nom_flag_1. If ALC is turned on (i. e. system_options.import_ind = Y), nom_flag_2 is used to indicate the status of a component in duty calculation nom_flag_4 is used to indicate the status of a component in ELC calculation nom_flag_5 is used to indicate the status of a component in ALC calculation i. e. these 3 flags can not be used for any other purposes if ALC is turned on.

Sample Input Message

```
{
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
```

```
"terms": "02",
"notBeforeDate": "2001-12-31",
"notAfterDate": "2001-12-31",
"otbEowDate": "2001-12-31",
"dept": null,
"status": "A",
"exchangeRate": 2,
"includeOnOrderInd": "Y",
"writtenDate": "2001-12-31",
"origin": "2",
"ediPoInd": "Y",
"preMarkInd": "Y",
"approvedBy": null,
"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
```

```

"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"fileId": "String",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"details": [
  {
    "item": "103900095",
    "location": 10001,
    "locationType": "W",
    "unitCost": 45.2,
    "referenceItem": null,
    "originCountry": "IN",
    "supplierPackSize": 1,
    "quantityOrdered": 25,
    "cancelInd": null,
    "reinstateInd": null,
    "deliveryDate": "2001-12-31",
    "quantityCancelled": null,
    "cancelCode": null,
    "estimatedInstockDate": "2001-12-31",
    "earliestShipDate": "2001-12-31",
    "latestShipDate": "2001-12-31",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ]
  }
],
"itemDetails": [
  {
    "item": "103900095",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ]
  }
],
"letterOfCredit": {
  "lcReferenceId": 10015000,
  "lcGroupId": null,
  "applicant": "500",
  "beneficiary": 2400,
  "merchandiseDescription": "Bath Essentials",
  "transshipmentInd": "Y",

```



```

        "partialShipmentInd": "Y",
        "lcAttachedInd": "Y"
    },
    "expenses": [
        {
            "item": "103900095",
            "packItem": null,
            "location": 10001,
            "locationType": "W",
            "component": "OCFRT",
            "computationValueBase": null,
            "calculationBasis": null,
            "componentRate": 17,
            "componentCurrency": "INR",
            "exchangeRate": 29,
            "perCount": 1,
            "perCountUom": "M3",
            "nominationFlag1": "N",
            "inDuty": "N",
            "nominationFlag3": "N",
            "inExpense": "+",
            "inAlc": "N"
        }
    ],
    "hts": [
        {
            "item": "103900095",
            "packItem": null,
            "hts": "9014111112",
            "status": "W",
            "originCountry": "IN",
            "importCountry": "US",
            "assessments": [
                {
                    "component": "DTYCAUS",
                    "computationValueBase": null,
                    "componentRate": 12,
                    "perCount": 1,
                    "perCountUom": "EA",
                    "nominationFlag1": "N",
                    "inDuty": "+",
                    "nominationFlag3": "-",
                    "inExpense": "N",
                    "inAlc": "N"
                }
            ]
        }
    ]
}

```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Delete Purchase Order Details

Functional Area

Purchase Orders

Business Overview

This webservice is used to delete line items from purchase orders in the worksheet status in order to keep Merchandising in sync with an external system that is responsible for maintaining purchase orders. For orders that are not in worksheet status, when a detail delete is requested, it will update the quantities to cancelled quantities and will be subject for re-approval.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/purchaseOrder/details/delete

Input Payload Details

Table 5-1249 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	

Table 5-1250 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
details	No	Collection of Object	This references the order detail node.

Table 5-1251 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	An approved, transaction level item. If a pack, it must be an orderable pack (with an order_as_type of 'P').
location	Yes	Number (10)	An active store or warehouse. If multichannel is on, and a warehouse is being order to, a virtual warehouse is expected.
referenceItem	No	String (25)	The id of a reference item which can be used instead of using the item field. If the item field is not populated this field is required.

Sample Input Message

```
{
  "items": [
    {
      "orderNo": 464690012,
      "dataLoadingDestination": "RMS",
      "details": [
        {
          "item": "100000323",
          "location": 10001,
          "referenceItem": null
        }
      ]
    }
  ]
}
```

```
    }  
  ]  
}  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS",  
  "message": "Service call is successful"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ],  
  "businessError": [  
    "Error message"  
  ]  
}
```

Delete Purchase Order Expenses

Functional Area

Purchase Orders

Business Overview

This webservice is used to delete expense information from the purchase order within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. Internally, the service sets the status of the order to Worksheet in order to accommodate the modification of expense information for purchase orders in the system that have been submitted or approved. Validation of the updated information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/purchaseOrder/expenses/delete

Input Payload Details

Table 5-1252 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order expenses.

Table 5-1253 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
expenses	No	Collection of Object	References the expense associated with a particular item/location of the purchase order.

Table 5-1254 Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Unique alphanumeric value that identifies the item.
packItem	No	String (25)	Alphanumeric value that uniquely identifies the pack.
location	Yes	Number (10)	This field contains the location that the item was ordered to.
component	Yes	String (10)	This field contains the component id.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "dataLoadingDestination": "RMS",
```

```
    "expenses": [
      {
        "item": "103900095",
        "packItem": null,
        "location": 10001,
        "component": "OCFRT"
      }
    ]
  }
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Purchase Order HTS

Functional Area

Purchase Orders

Business Overview

This webservice is used to delete Harmonized Tariff Schedules and related assessments within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. When deleting HTS information from a purchase order that has been submitted or approved, internally, the service sets the status of the order to Worksheet in order to accommodate the change in HTS information.

Validation of the new HTS information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/purchaseOrder/hts/delete

Input Payload Details

Table 5-1255 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order HTS records.

Table 5-1256 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1257 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Unique alphanumeric value that identifies the item.
packItem	No	String (25)	Alphanumeric value that uniquely identifies the pack.
hts	Yes	String (25)	The unique identifier for the Harmonized Tariff Schedule code.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
```

```

        "dataLoadingDestination": "RMS",
        "hts": [
            {
                "item": "103900095",
                "packItem": null,
                "hts": "901411112"
            }
        ]
    }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Delete Purchase Order HTS Assessments

Functional Area

Purchase Orders

Business Overview

This webservice is used to delete Harmonized Tariff Schedule assessments within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. When deleting assessment information from a purchase order that has been submitted or approved, the service internally sets the status of the order to Worksheet in order to accommodate the change in information.

Validation of the new HTS information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/purchaseOrder/hts/assessments/delete

Input Payload Details

Table 5-1258 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order HTS assessment records.

Table 5-1259 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1260 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Unique alphanumeric value that identifies the item.
packItem	No	String (25)	Alphanumeric value that uniquely identifies the pack.
hts	Yes	String (25)	The unique identifier for the Harmonized Tariff Schedule code.
assessments	No	Collection of Object	References the assessment associated with the HTS of a particular item in the purchase order.

Table 5-1261 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains the component id.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "dataLoadingDestination": "RMS",
      "hts": [
        {
          "item": "103900095",
          "packItem": null,
          "hts": "9014111112",
          "assessments": [
            {
              "component": "MPFUS"
            }
          ]
        }
      ]
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
    }
  ]
}
```

```

        "inputValue": "X"
      }
    ]
  }

```

Delete Purchase Order Letters of Credit

Functional Area

Purchase Orders

Business Overview

This webservice is used to delete letter of credit information within Merchandising when the order payment type is Letter of Credit in order to keep it in sync with an external system that is responsible for maintaining purchase orders.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/purchaseOrder/lettersOfCredit/delete

Input Payload Details

Table 5-1262 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of letter of credit records.

Table 5-1263 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.

Sample Input Message

```

{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,

```

```
        "dataLoadingDestination": "RMS"
      }
    ]
  }
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Delete Purchase Orders

Functional Area

Purchase Orders

Business Overview

This webservice is used to delete purchase orders from Merchandising in sync with an external system that is responsible for maintaining purchase orders. If you are deleting the entire purchase order, this service will first validate that the order number is valid since an existing order number is the only required field for a header delete message. Deleting an order will in effect set the status of the order at the header level to D. Only worksheet orders can be deleted and deletion will not be allowed if the order is submitted, approved or has been approved, or if allocations exist for the order. Delete messages will still be processed, however it will be treated as an update of cancelled quantity and the quantity ordered will be reduced to the quantity available to be cancelled. If this results in all line items being cancelled or if the delete is made at header level, the status of the order will become Closed.

If an order is still in worksheet status, the entire order will be deleted. If the order involves any franchise stores, then any franchise order or return created with the order will also be cancelled or deleted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

DELETE

ReST URL

MerchIntegrations/services/purchaseOrders/delete

Input Payload Details

Table 5-1264 Delete - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of orders.

Table 5-1265 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "dataLoadingDestination": "RMS"
    }
  ]
}
```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}
```

Update Purchase Order Details

Functional Area

Purchase Orders

Business Overview

This webservice is used to update detail level attributes of the purchase order such as order quantities, unit cost and line item cancellation information in order to keep Merchandising in sync with an external system that is responsible for maintaining purchase orders. Order details can be updated for orders in Approved, Worksheet, Submitted or Closed status. The only information needed at the header level is the order number, which if not provided, will cause the message to be rejected. All other details provided at the header level will be ignored. Modifying order quantity, as well as supplier pack size or unit cost on an approved or submitted order will in effect set the order status to worksheet and subject it for automatic re-approval. When modifying order quantities, the full amount should be provided, not just the difference in the old and new values. Validations are also done on quantity changes, such as the ordered quantity should not go below the allocated quantity or replenishment quantity, quantity ordered cannot not be less than quantity received.

Fields that can be modified in worksheet, submitted and approved status:

Supplier Pack Size

Unit Cost – for items with no received quantities

Quantity Ordered

Fields that cannot be modified in statuses other than worksheet:

Origin Country ID

Location

Fields that can be modified only in approved status:

Quantity Cancelled

Cancel Code

Reinstating order lines

To reinstate orders, the reinstate indicator should be set to Y. In effect, this will set the cancelled quantities of the line items to 0 and reinstate the ordered quantities.

Cancelling a line item in an approved order

In order to cancel a line item on the order, you can set the cancel indicator at the detail level to Y and at the same time, the quantity ordered for that line item must be set to 0. For partial cancellations, either reduce the quantity of an approved order or populate the quantity cancelled field making sure the cancel indicator is blank or set to N. This will allow for the automatic re-approval of the entire order, if there are line items still on the order once processed by the API. The cancel indicator and reinstate indicator cannot both be set to Y at the same time.

For more details on Purchase Orders, see the Create Purchase Order service description.

Reinstating Order Lines

To reinstate orders, the reinstate indicator should be set to Y. In effect, this will set the cancelled quantities of the line items to 0 and reinstate the ordered quantities. This will set the status of the reinstated order to Worksheet.

Cancelling a Line Item in an Approved Order

In order to cancel a line item on the order, you can set the cancel indicator at the detail level to Y and at the same time, the quantity ordered for that line item must be set to 0. For partial cancellations, either reduce the quantity of an approved order or populate the quantity cancelled field making sure the cancel indicator is blank or set to N. This will allow for the automatic re-approval of the entire order, if there are line items still on the order once processed by the API. The cancel indicator and reinstate indicator cannot both be set to Y at the same time.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/purchaseOrder/details/update

Input Payload Details

Update - Object. See list of elements for detail

Items - Object. See list of elements for detail

Details - Object. See list of elements for detail

CustomFlexAttribute - Object. See list of elements for detail

ItemDetails - Object. See list of elements for detail

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Update Purchase Order Expenses

Functional Area

Purchase Orders

Business Overview

This webservice is used to update expense information on the purchase order within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. Internally, the service sets the status of the order to Worksheet in order to accommodate the modification of expense information for purchase orders in the system that have been submitted or approved. Validation of the new expense information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/purchaseOrder/expenses/update

Input Payload Details

Table 5-1266 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order expenses.

Table 5-1267 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.

Table 5-1267 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.

Table 5-1267 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.

Table 5-1267 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
purchaseType	No	String (6)	This field indicates what's included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.

Table 5-1267 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.

Table 5-1267 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
expenses	No	Collection of Object	References the expense associated with a particular item/location of the purchase order.

Table 5-1268 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1269 Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains alphanumeric value that uniquely identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
location	Yes	Number (10)	This field contains the location that item will be ordered to. This field may contain a store or a warehouse.
locationType	Yes	String (1)	This field contains the type of location in the location field. Valid values are: S - Store W - Warehouse.
component	Yes	String (10)	This field contains a user specified code representing a specific expense found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the expense.
calculationBasis	No	String (1)	This field contains the basis the expense will be calculated against if there is no CVB code attached to the expense. Valid values are S (Supplier) or O (Order). If the Cost Basis is S then when calculating the expense, the items supplier cost will be used. If the Cost Basis is O then the order unit cost of the item on the order will be used to calculate the expense. The code type for this field is EXCB.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base of the component (either value or specific).
componentCurrency	Yes	String (3)	This field contains the currency the expense or assessment is to be entered in. If the Calculation Basis is V (Value), then comp_currency will be NULL.
exchangeRate	Yes	Number (20,10)	This field contains the exchange rate of the individual expense.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.

Table 5-1269 (Cont.) Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.

Table 5-1269 (Cont.) Expenses - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.
inAlc	Yes	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
    }
  ]
}
```



```
"status": "A",
"includeOnOrderInd": "Y",
"writtenDate": "2001-12-31",
"origin": "2",
"ediPoInd": "Y",
"preMarkInd": "Y",
"approvedBy": null,
"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
```

```

        "valueDate": "2001-12-31"
      }
    ],
    "expenses": [
      {
        "item": "103900095",
        "packItem": null,
        "location": 10001,
        "locationType": "W",
        "component": "OCFRT",
        "computationValueBase": null,
        "calculationBasis": null,
        "componentRate": 17,
        "componentCurrency": "INR",
        "exchangeRate": 29,
        "perCount": 1,
        "perCountUom": "M3",
        "nominationFlag1": "N",
        "inDuty": "N",
        "nominationFlag3": "N",
        "inExpense": "+",
        "inAlc": "N"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

```
]
}
```

Update Purchase Order HTS

Functional Area

Purchase Orders

Business Overview

This webservice is used to update Harmonized Tariff Schedules and related assessments within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. When updating HTS information for a purchase order that has been submitted or approved, internally, the service sets the status of the order to Worksheet in order to accommodate the change in HTS information. Validation of the new HTS information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/purchaseOrder/hts/update

Input Payload Details

Table 5-1270 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order HTS records.

Table 5-1271 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.

Table 5-1271 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.

Table 5-1271 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.
purchaseType	No	String (6)	This field indicates whats included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).

Table 5-1271 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.

Table 5-1271 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.

Table 5-1271 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1272 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1273 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains an unique alphanumeric value that identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.

Table 5-1273 (Cont.) Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hts	Yes	String (25)	This field contains an identifier for the Harmonized Tariff Schedule code.
status	Yes	String (1)	This field contains a flag to indicate the status of the Purchase Order/Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
originCountry	Yes	String (3)	This field contains the country of manufacture when the HTS tracking level is M. When the HTS tracking level is S, this stores the country of sourcing.
importCountry	No	String (3)	This field contains the country that the item will be imported in to.
assessments	No	Collection of Object	References the HTS code assessments associated with a particular item of the purchase order.

Table 5-1274 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains a user specified code representing the specific assessment component found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the assessment.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount in the currency defined for the assessment (usually the import country's currency) if the Calculation Basis is Specific.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.

Table 5-1274 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.
inDuty	Yes	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.

Table 5-1274 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inAlc	Yes	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
      "status": "A",
      "includeOnOrderInd": "Y",
      "writtenDate": "2001-12-31",
      "origin": "2",
      "ediPoInd": "Y",
      "preMarkInd": "Y",
      "approvedBy": null,
      "commentDesc": "Create Order",
      "dataLoadingDestination": "RMS",
      "masterOrderNo": null,
      "ladingPort": "7",
      "dischargePort": "480",
      "factory": "100",
      "agent": "007",
      "shipMethod": "32",
      "partnerType1": "AG",
      "partner1": "007",
      "partnerType2": "E",
      "partner2": "4678",
      "partnerType3": null,
      "partner3": null,
      "importCountry": "US",
      "purchaseType": "FOB",
      "paymentMethod": "LC",
      "titlePassLocationType": "FF",

```

```

"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"hts": [
  {
    "item": "103900095",
    "packItem": null,
    "hts": "901411112",
    "status": "W",
    "originCountry": "IN",
    "importCountry": "US",
    "assessments": [
      {
        "component": "DTYCAUS",
        "computationValueBase": null,
        "componentRate": 12,
        "perCount": 1,
        "perCountUom": "EA",
        "nominationFlag1": "N",
        "inDuty": "+",
        "nominationFlag3": "-",
        "inExpense": "N",
        "inAlc": "N"
      }
    ]
  }
]

```

```
    }  
  ]  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Update Purchase Order HTS Assessments

Functional Area

Purchase Orders

Business Overview

This webservice is used to update Harmonized Tariff Schedules assessments within Merchandising in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order is not yet submitted or approved. When updating assessment information for a purchase order that has been submitted or approved, the service internally sets the status of the order to Worksheet in order to accommodate the change in assessment information. Validation of the new information is performed before automatic order re-approval is attempted.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/purchaseOrder/hts/assessments/update

Input Payload Details

Table 5-1275 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of order HTS assessment records.

Table 5-1276 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.

Table 5-1276 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.

Table 5-1276 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.
purchaseType	No	String (6)	This field indicates whats included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.

Table 5-1276 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).

Table 5-1276 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Puchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.

Table 5-1276 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
hts	No	Collection of Object	References the HTS code associated with a particular item of the purchase order.

Table 5-1277 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1278 Hts - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	This field contains an unique alphanumeric value that identifies the item.
packItem	No	String (25)	This field contains alphanumeric value that uniquely identifies the pack.
hts	Yes	String (25)	This field contains an identifier for the Harmonized Tariff Schedule code.
status	Yes	String (1)	This field contains a flag to indicate the status of the Purchase Order/Item/HTS relationship. Valid values are (W)orksheet and (A)pproved.
originCountry	Yes	String (3)	This field contains the country of manufacture when the HTS tracking level is M. When the HTS tracking level is S, this stores the country of sourcing.
importCountry	No	String (3)	This field contains the country that the item will be imported in to.
assessments	No	Collection of Object	References the HTS code assessments associated with a particular item of the purchase order.

Table 5-1279 Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
component	Yes	String (10)	This field contains a user specified code representing the specific assessment component found on the ELC_COMP table.
computationValueBase	No	String (10)	This field contains a code representing the base value for calculating the assessment.
componentRate	Yes	Number (20,4)	This field contains the rate to be charged against the calculation base. This field will hold a percentage value if the Calculation Basis of the Component is Value or a monetary amount in the currency defined for the assessment (usually the import country's currency) if the Calculation Basis is Specific.
perCount	No	Number (12,4)	This field contains a count indicating the amount of the Per Count Unit of Measure to which the rate applies. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count would be 10.
perCountUom	No	String (4)	This field contains the unit of measure in which the Per Count is specified. When creating an expense that is defined as \$. 20 for every 10 kilograms of an item ordered, the Per Count UOM would be the Unit of Measure code for Kilograms.
nominationFlag1	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 1), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 1), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 1). For example, if cost component ABC has a + in Nomination Flag 1, then all CVBs that include Nomination Flag 1 will include this cost component value in their calculation basis.

Table 5-1279 (Cont.) Assessments - Object. See list of elements for detail

Element Name	Required	Data Type	Description
inDuty	Yes	String (1)	If Import Management is enabled in Merchandising (i. e. system_options.import_ind = Y), the inDuty flag is used to indicate if the value of the cost component should be included in total duty.
nominationFlag3	Yes	String (1)	This field contains a flag that is used to indicate if the value of the cost component should be included in the calculation of another. Nomination flags are used to calculate the value of a Computation Value Base (CVB) which has the corresponding nomination flag set to Yes (Y). Valid values are: N/A (N) (the value of this component will not be included in CVBs which look at Nomination Flag 3), Add (+) (the value of this component will be added to CVBs which look at Nomination Flag 3), and Subtract (-) (the value of this component will be deducted from CVBs which look at Nomination Flag 3). For example, if cost component ABC has a + in Nomination Flag 3, then all CVBs that include Nomination Flag 3 will include this cost component value in their calculation basis.
inExpense	Yes	String (1)	The inExpense flag is used to indicate if the value of the cost component should be included in total expenses.
inAlc	Yes	String (1)	The inALC flag is used for cost components where the retailer does not expect to receive an actual value for the charge via an obligation or customs entry. The inALC flag allows the value of the cost component to be included in ALC via direct nomination and indicates to the system that the estimated value on the order should be used as the actual value as well because no other actual will be provided.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
```

```
"terms": "02",
"notBeforeDate": "2001-12-31",
"notAfterDate": "2001-12-31",
"otbEowDate": "2001-12-31",
"dept": null,
"status": "A",
"includeOnOrderInd": "Y",
"writtenDate": "2001-12-31",
"origin": "2",
"ediPoInd": "Y",
"preMarkInd": "Y",
"approvedBy": null,
"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
```

```

    "latestShipDate": "2001-12-31",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ],
    "hts": [
      {
        "item": "103900095",
        "packItem": null,
        "hts": "901411112",
        "status": "W",
        "originCountry": "IN",
        "importCountry": "US",
        "assessments": [
          {
            "component": "DTYCAUS",
            "computationValueBase": null,
            "componentRate": 12,
            "perCount": 1,
            "perCountUom": "EA",
            "nominationFlag1": "N",
            "inDuty": "+",
            "nominationFlag3": "-",
            "inExpense": "N",
            "inAlc": "N"
          }
        ]
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
}

```

```
    "validationErrors": [
      {
        "error": "must be one of Y, N",
        "field": "createRecord.arg0.approveInd",
        "inputValue": "X"
      }
    ]
  }
}
```

Update Purchase Order Letters of Credit

Functional Area

Purchase Orders

Business Overview

This webservice is used to update letter of credit information within Merchandising when order payment type is Letter of Credit in order to keep it in sync with an external system that is responsible for maintaining purchase orders. It also updates order header information with the values provided in the request if the order has not yet been submitted or approved. For an existing order that has a payment type of Letter of Credit, the modification of header level order information is not necessary; however header level fields, particularly the payment type can be updated in case a letter of credit needs to be attached to the worksheet order that has an existing payment type other than Letter of Credit. Internally, the service sets the status of the order to Worksheet in order to accommodate the modification of letter of credit information for purchase orders in the system that have been submitted or approved. Validation of updated information is performed before automatic order re-approval is attempted without having to use the separate Update Purchase Order Service. If order has been previously approved and the order is attached to a Letter of Credit, only the Transshipment and Partial Shipment can be changed in the letter of credit details.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/purchaseOrder/lettersOfCredit/update

Input Payload Details

Table 5-1280 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of letter of credit records.

Table 5-1281 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.
currencyCode	No	String (3)	The code of the order's currency. If not populated, the supplier's currency will be used. This field can be modified if the order status is worksheet or approved and the order has not been shipped.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.

Table 5-1281 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.

Table 5-1281 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
importCountry	No	String (3)	The identifier of the country into which the items on the order are being imported.
purchaseType	No	String (6)	This field indicates whats included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.

Table 5-1281 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to a Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.

Table 5-1281 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH folumn with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item can not be shipped by the supplier.
latestShipDate	No	date	The date after which the item can not be shipped by the supplier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.
letterOfCredit	No	Record	References the letter of credit for the purchase order.

Table 5-1282 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Table 5-1283 LetterOfCredit - Object. See list of elements for detail

Element Name	Required	Data Type	Description
lcReferenceId	No	Number (8)	This field contains the ID of the LC to which the order is attached.
lcGroupId	No	String (10)	This field contains an ID that can be used to group Orders together so that they can be attached to a single Letter of Credit.
applicant	Yes	String (10)	This field contains the ID of the applicant being used for the LC process.
beneficiary	Yes	Number (10)	This field contains the ID of the beneficiary being used for the LC process.
merchandiseDescription	Yes	String (2000)	This field holds a description of the merchandise on the order.
transshipmentInd	Yes	String (1)	This field indicates whether or not the LC to which the order is attached will allow the order to be transshipped. Valid values are 'Y'es and 'N'o.
partialShipmentInd	Yes	String (1)	This field indicates whether or not the LC to which the order is attached will allow the order to be partially shipped. Valid values are 'Y'es and 'N'o.
lcAttachedInd	Yes	String (1)	This field indicates whether or not the order has been attached to a Letter of Credit. Valid values are Y and N.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "currencyCode": "USD",
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
      "status": "A",
      "includeOnOrderInd": "Y",
      "writtenDate": "2001-12-31",
      "origin": "2",
      "ediPoInd": "Y",
      "preMarkInd": "Y",
      "approvedBy": null,
    }
  ]
}
```

```

"commentDesc": "Create Order",
"dataLoadingDestination": "RMS",
"masterOrderNo": null,
"ladingPort": "7",
"dischargePort": "480",
"factory": "100",
"agent": "007",
"shipMethod": "32",
"partnerType1": "AG",
"partner1": "007",
"partnerType2": "E",
"partner2": "4678",
"partnerType3": null,
"partner3": null,
"importCountry": "US",
"purchaseType": "FOB",
"paymentMethod": "LC",
"titlePassLocationType": "FF",
"titlePassLocation": "US",
"purchaseOrderType": "4000",
"orderType": "N/B",
"buyer": 1000,
"location": null,
"locationType": null,
"promotion": 41,
"qualityControlInd": "Y",
"freightTerms": "03",
"backhaulType": null,
"backhaulAllowance": null,
"shipPayMethod": "PO",
"transportationResponsibilityType": "FF",
"transportationResponsibility": "US",
"vendorOrderNo": "9987001",
"freightContractNo": "111",
"pickupLocation": null,
"pickupNo": null,
"pickupDate": null,
"appointmentDateTime": "2001-12-31T23:59:59.123Z",
"importType": "W",
"importId": 10001,
"clearingZoneId": null,
"routingLocation": null,
"reApproveInd": null,
"earliestShipDate": "2001-12-31",
"latestShipDate": "2001-12-31",
"customFlexAttribute": [
  {
    "name": "REVIEW_DATE",
    "value": null,
    "valueDate": "2001-12-31"
  }
],
"letterOfCredit": {
  "lcReferenceId": 10015000,
  "lcGroupId": null,
  "applicant": "500",

```

```
        "beneficiary": 2400,  
        "merchandiseDescription": "Bath Essentials",  
        "transshipmentInd": "Y",  
        "partialShipmentInd": "Y",  
        "lcAttachedInd": "Y"  
      }  
    }  
  ]  
}
```

Response Code: 200 (Success)

Sample Response Message

```
{  
  "status": "SUCCESS"  
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```
{  
  "status": "ERROR",  
  "message": "Error found in validation of input payload",  
  "validationErrors": [  
    {  
      "error": "must be one of Y, N",  
      "field": "createRecord.arg0.approveInd",  
      "inputValue": "X"  
    }  
  ]  
}
```

Update Purchase Orders

Functional Area

Purchase Orders

Business Overview

This webservice is used to update purchase orders in order to keep Merchandising in sync with an external system that is responsible for maintaining purchase orders. It can be used to modify only certain header level attributes of existing purchase orders. There are certain fields that are not allowed to be updated at header level depending on the status, and if these are still provided in the message, they will simply be ignored and no error message will be returned. Modification of the following header level fields is allowed while the order is submitted or approved, without having to set the order in worksheet status: status, not before date, not after date, terms, include on-order indicator, and comments.

For more details on Purchase Orders, see the Create Purchase Order service description.

Service Type

PUT

ReST URL

MerchIntegrations/services/purchaseOrders/update

Input Payload Details

Table 5-1284 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	Yes	Collection of Object	

Table 5-1285 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
orderNo	Yes	Number (12)	The unique identifier for the order.
supplier	No	Number (10)	The identifier of the supplier from which the order will be sourced. This cannot be modified if details exist for the PO.
terms	No	String (15)	The sales terms of the order. If not populated on the create message, will default to the supplier's terms. This can be modified in worksheet or approved status.
notBeforeDate	No	date	The first date that delivery will be accepted. This can be modified in worksheet or approved status.
notAfterDate	No	date	The last date that delivery will be accepted. This can be modified in worksheet or approved status.
otbEowDate	No	date	The end of week date of the OTB bucket used. This can be modified in worksheet or approved status.
dept	No	Number (4)	The department in which are all the items on the order. This field is required if the system enforces single department ordering. This cannot be modified if details exist for the PO.
status	No	String (1)	The code for the status of the order. Valid values are 'W' worksheet and 'A' approved for PO creation. It is also possible to modify the status to 'C' closed.

Table 5-1285 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
exchangeRate	No	Number (20,10)	The rate of exchange for the PO used between the order and primary currencies. This field can be modified if the order status is worksheet or approved and the order has not been shipped..
includeOnOrderInd	No	String (1)	Indicates if the order should be included in on-order calculations. This can be modified in worksheet or approved status. Valid values are Y and N.
writtenDate	No	date	The date the order was created. If this field is not populated on the message it will default to the time of creation in RMS. It cannot be modified.
origin	No	String (2)	Indicates where the order originated. Valid values include: 2 - Manual, 6 - AIP generated order, 7, 8, 12 - Consignment Vendor Order.
ediPoInd	No	String (1)	Indicates whether or not the order will be transmitted to the supplier via an Electronic Data Exchange transaction. Valid values are: Y = Submit via EDI, N = Do not use EDI.
preMarkInd	No	String (1)	This field indicated whether or not a supplier has agreed to break an order into separate boxes so that the boxes can be sent directly to stores. Valid values are Y and N.
approvedBy	No	String (254)	Indicates where the order was approved. It will be the user ID of the person approving the order.
commentDesc	No	String (2000)	Any comments pertaining to the order.
dataLoadingDestination	No	String (6)	This field indicates if the order will be created in RMS or the staging tables. Valid values are RMS or STG. If not defined, the default is STG.
masterOrderNo	No	Number (12)	This field indicates the master order number from which child records were created.
ladingPort	No	String (5)	This field indicates the port from which the items on the purchase order are shipped.
dischargePort	No	String (5)	This field indicates the port at which the items on the purchase order will enter the country of import.
factory	No	String (10)	This field indicates the factory at which the items on the purchase order are made.

Table 5-1285 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
agent	No	String (10)	This field indicates the agent that is used in the purchase of the items on the purchase order.
shipMethod	No	String (6)	This field indicates the method used to ship the items on the purchase order from the country of origin to the country of import.
partnerType1	No	String (6)	This field indicates Partner 1 Type.
partner1	No	String (10)	This field indicates the Additional Partner 1.
partnerType2	No	String (6)	This field indicates Partner 2 Type.
partner2	No	String (10)	This field indicates the Additional Partner 2.
partnerType3	No	String (6)	This field indicates Partner 3 Type.
partner3	No	String (10)	This field indicates the Additional Partner 3.
purchaseType	No	String (6)	This field indicates what is included in the suppliers cost of the item. Valid values include C (Cost), CI (Cost and Insurance), CIF (Cost, Insurance and Freight) and FOB (Free on Board).
paymentMethod	No	String (6)	This field indicates the default method for how purchase orders for this site will be paid. Valid values are in the code type PYMT(Payment Method). Examples are Open Account(OA), Letter of Credit(LC) and Wire Transfer(WT).
titlePassLocationType	No	String (2)	Contains the code indicating the type of location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and City (CI).
titlePassLocation	No	String (250)	Contains the description of the location or point where ownership of merchandise changes hands from seller/supplier to buyer/retailer.
purchaseOrderType	No	String (4)	This field contains the value associated with the PO_TYPE for the order.
orderType	No	String (3)	Indicates the type of order and which Open To Buy bucket will be updated. Valid values include: N/B - Non Basic, ARB - Automatic Reorder of Basic, BRB - Buyer Reorder of Basic, DSD - Direct Store Delivery

Table 5-1285 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
buyer	No	Number (4)	Contains the number associated with the buyer for the order.
location	No	Number (10)	This field contains the location all items on the order will be delivered to if populated. i. e. It will mean a single location order.
locationType	No	String (1)	This field contains the type of location in the location field. Valid values are: Valid values are S (Store) or W (Warehouse).
promotion	No	Number (10)	Contains the RPCS offer ID associated with the order to provide a link between the order dialog and the promotions dialog.
qualityControlInd	No	String (1)	Determines whether or not quality control will be required when items for this order are received. Valid values are Y and N.
freightTerms	No	String (30)	Indicator that references what freight terms are related to the order.
backhaulType	No	String (6)	This field contains the type of backhaul allowance that will be applied to the order. Some examples are Calculated or Flat rate.
backhaulAllowance	No	Number (20,4)	This field will contain the backhaul allowance value.
shipPayMethod	No	String (2)	Code indicating the payment terms for freight charges associated with the order. Valid values include: CC - Collect, CF - Collect Freight Credited Back to Customer, DF - Defined by Buyer and Seller, MX - Mixed, PC - Prepaid but Charged to Customer, PO - Prepaid Only, PP - Prepaid by Seller.
transportationResponsibility Type	No	String (2)	Contains the code indicating the type of location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer. Valid values are found under the FOBT code type. Examples are City and State (AC), Country Of Sourcing (CA) and Country (CC).
transportationResponsibility	No	String (250)	Contains the description of the location or point at which the responsibility for transportation transfers from the seller/supplier to the buyer/retailer.

Table 5-1285 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
vendorOrderNo	No	String (15)	Contains the vendors unique identifying number for an order. These orders may have originated by the vendor through the EDI process or this number can be associated to an Oracle Retail order when the order is created on-line.
freightContractNo	No	String (10)	The number of the contract with a shipper that will give specific freight rates. This field is only available when the system is running Import functionality.
pickupLocation	No	String (250)	Contains the location at which the order will be picked up, if the order is a Pickup order.
pickupNo	No	String (25)	Contains the reference number of the Pickup order.
pickupDate	No	date	Contains the date when the order can be picked up from the Supplier. This field is only required if the Purchase Type of the order is Pickup.
appointmentDateTime	No	date-time	This column will hold the date and time of the receiving appointment at the warehouse.
importType	No	String (1)	Valid values are IMporter and EXporter. This is the default importer/exporter assigned to the supplier of the Purchase order.
importId	No	Number (10)	This identifies the importer/exporter assigned to the supplier. This should reference the WH. WH column with foreign key created.
clearingZoneId	No	String (5)	This column will hold the clearing zone id.
routingLocation	No	String (5)	This is the default routing location for the import order. FK on OUTLOC. OUTLOC_ID.
reApproveInd	No	String (1)	This field indicates that the update to the corresponding purchase order needs to be performed, and then it should be approved again. Valid values are Y and N.
earliestShipDate	No	date	The date before which the item cannot be shipped by the supplier.
latestShipDate	No	date	The date after which the item cannot be shipped by the supplier.

Table 5-1285 (Cont.) Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
fileId	No	String (20)	In order to support supplier pooling via the buyer worksheet, the recommended order quantities for the individual suppliers are linked together for logistical purposes while generating orders from the worksheet based on this identifier.
customFlexAttribute	No	Collection of Object	References a collection of custom flexible attributes. This collection is treated as a single group and should include all named flexible attributes for the business entity. If any configured named attribute is missing from the input collection, its value will be set to NULL.

Table 5-1286 CustomFlexAttribute - Object. See list of elements for detail

Element Name	Required	Data Type	Description
name	Yes	String (30)	Holds the attribute name.
value	No	String (250)	Holds the numeric or string value of the attribute.
valueDate	No	date	Holds the date value of the attribute.

Sample Input Message

```
{
  "items": [
    {
      "orderNo": 464690012,
      "supplier": 2400,
      "terms": "02",
      "notBeforeDate": "2001-12-31",
      "notAfterDate": "2001-12-31",
      "otbEowDate": "2001-12-31",
      "dept": null,
      "status": "A",
      "exchangeRate": 2,
      "includeOnOrderInd": "Y",
      "writtenDate": "2001-12-31",
      "origin": "2",
      "ediPoInd": "Y",
      "preMarkInd": "Y",
      "approvedBy": null,
      "commentDesc": "Create Order",
      "dataLoadingDestination": "RMS",
      "masterOrderNo": null,
      "ladingPort": "7",
      "dischargePort": "480",
      "factory": "100",

```

```

    "agent": "007",
    "shipMethod": "32",
    "partnerType1": "AG",
    "partner1": "007",
    "partnerType2": "E",
    "partner2": "4678",
    "partnerType3": null,
    "partner3": null,
    "purchaseType": "FOB",
    "paymentMethod": "LC",
    "titlePassLocationType": "FF",
    "titlePassLocation": "US",
    "purchaseOrderType": "4000",
    "orderType": "N/B",
    "buyer": 1000,
    "location": null,
    "locationType": null,
    "promotion": 41,
    "qualityControlInd": "Y",
    "freightTerms": "03",
    "backhaulType": null,
    "backhaulAllowance": null,
    "shipPayMethod": "PO",
    "transportationResponsibilityType": "FF",
    "transportationResponsibility": "US",
    "vendorOrderNo": "9987001",
    "freightContractNo": "111",
    "pickupLocation": null,
    "pickupNo": null,
    "pickupDate": null,
    "appointmentDateTime": "2001-12-31T23:59:59.123Z",
    "importType": "W",
    "importId": 10001,
    "clearingZoneId": null,
    "routingLocation": null,
    "reApproveInd": null,
    "earliestShipDate": "2001-12-31",
    "latestShipDate": "2001-12-31",
    "fileId": "String",
    "customFlexAttribute": [
      {
        "name": "REVIEW_DATE",
        "value": null,
        "valueDate": "2001-12-31"
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```
{
  "status": "SUCCESS",
  "message": "Service call is successful"
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    "Error message"
  ]
}
```

Replenishment

The following services are included in this functional area:

- [Inventory Request Upload Service](#)
 - [Create Inventory Requests](#)
- [Replenishment Scheduled Update Upload Service](#)

Inventory Request Upload Service

The following services are included in this functional area:

- [Create Inventory Requests](#)

Create Inventory Requests

Functional Area

Replenishment

Business Overview

Merchandising receives requests for inventory using this service, which allows for items to be ordered by the store and fulfilled through Merchandising. Unlike store order replenishment, Merchandising fulfills inventory requests from the store regardless of replenishment review cycles, delivery dates, and any other factors that may restrict a request from being fulfilled.

For item/store combinations that are on the Store Order type of replenishment in Merchandising, orders will be placed using this service and then the replenishment process builds the recommended order quantity (ROQ) based on the store's requests. Requests that will not be reviewed prior to the date requested by the store are fulfilled through a one-off process (through this service) that creates warehouse transfers and/or purchase orders to fulfill the requested quantities.

For item/location combinations that are currently using other methods of replenishment in Merchandising, the store requested quantities will be added on top of the calculated recommended order quantities to increase the overall replenishment. It can also be used for item/store combinations not on replenishment in Merchandising. In these cases, the one-off process described above will be used to create purchase orders or transfers, utilizing attributes defined for the item/location.

Other validation notes:

- Order quantities will be rounded using the store order multiple when an order is created for a warehouse or rounded to the case size if ordering from the supplier.
- Up charges will always be applied to a transfer, if they are defined for the items with locations matching the transfer.
- Merchandising will validate that all items belong to the same department when department level ordering (supplier) or department level transfers (warehouse) are being used.
- The store must be open for ordering.

Service Type

POST

ReST URL

MerchIntegrations/services/inventory/requests/create

Input Payload Details

Table 5-1287 Create - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of inventory request details.

Table 5-1288 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
requestId	Yes	Number (10)	Used to build an error record when necessary.
store	Yes	Number (10)	The store ID which is requesting the item. The store must exist as a valid stockholding store in Merchandising.
requestType	Yes	String (2)	Defines the request type. Valid values are: Store order (SO) or Inventory Request (IR). If the request type is SO or blank, then replenishment method should be store order. If the request type is IR, delivery slot should be provided.
details	Yes	Collection of Object	References a collection of items on the inventory request.

Table 5-1289 Details - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Unique alphanumeric value that identifies the item to be requested. The item must be approved, orderable, and inventoried item; it must also be ranged to the location in the inventory request and must be active at that location.
quantity	Yes	Number (12,4)	This is the quantity being requested in standard UOM.
unitOfPurchase	Yes	String (6)	Unit of purchase must either be eaches (EA), case (CA), or pallet (PA).
needDate	Yes	date	This is the date that the store needs the item by.
deliverySlotId	No	String (15)	Specifies when the store requested quantity is needed at the store. Valid delivery slots are in the DELIVERY_SLOT table.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "requestId": 693933,
      "store": 6000,
      "requestType": "SO",
      "details": [
        {
          "item": "100100076",
          "quantity": 3,

```

```

        "unitOfPurchase": "EA",
        "needDate": "2001-12-31",
        "deliverySlotId": null
      }
    ]
  }
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` will be present when input payload or input parameters do not match the schema definition for this service.

Sample Error Message

```

{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ]
}

```

Replenishment Scheduled Update Upload Service

This section describes the Replenishment Schedule service.

Functional Area

Inventory Movement

Business Overview

The primary role of these services is to create, modify, and delete scheduled replenishments and send them to Merchandising.

Create Scheduled Replenishment Updates

This section describes the Create Replenishment Schedule service.

Business Overview

This service creates scheduled replenishments by calling the SVCPROV_REPL package to load input data to the staging tables and then calling the core replenishment package to validate and insert data to the Merchandising tables.

Service Type

Post

ReST URL

inventory/replenishment/createReplSched

Input Parameters

Table 5-1290 ReplSchedCreModRDO

Parameter Name	Data Type
replAttrId	BigDecimal
schRplDesc	String
scheduledActiveDate	Date
replAction	String
item	String
diff1	String
diff2	String
diff3	String
diff4	String
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
loc	BigDecimal
locType	String
autoRangeInd	String
activateDate	Date
deactivateDate	Date
presStock	BigDecimal
demoStock	BigDecimal
stockCat	String
replOrderCtrl	String
sourcingWh	BigDecimal
supplier	BigDecimal
originCountryId	String
pickupLeadTime	BigDecimal
whLeadTime	BigDecimal
replMethodInd	String

Table 5-1290 (Cont.) ReplSchedCreModRDO

Parameter Name	Data Type
replMethod	String
optimizeInd	String
minStock	BigDecimal
maxStock	BigDecimal
incrPct	BigDecimal
minSupplyDays	BigDecimal
maxSupplyDays	BigDecimal
timeSupplyHorizon	BigDecimal
addLeadTimeInd	String
invSellingDays	BigDecimal
serviceLevelType	String
serviceLevel	BigDecimal
serviceLevelFloatingStd	String
lostSalesFactor	BigDecimal
terminalStockQty	BigDecimal
seasonId	BigDecimal
phaseId	BigDecimal
rejectStoreOrdInd	String
multRunsPerDayInd	String
tsfZeroSohInd	String
nonScalingInd	String
maxScaleValue	BigDecimal
sizeProfileInd	String
reviewCycle	String
updateDaysInd	String
mondayInd	String
tuesdayInd	String
wednesdayInd	String
thursdayInd	String
fridayInd	String
saturdayInd	String
sundayInd	String
primaryPackNo	String
defaultPackInd	String
removePackInd	String
mraUpdate	String
mraRestore	String
includeAdditionalDemand	String
storeUserAccess	String

Table 5-1290 (Cont.) ReplSchedCreModRDO

Parameter Name	Data Type
storeToleranceType	String
storeToleranceLimit	BigDecimal

JSON Structure:

```
[
  {
    "replAttrId": null,
    "schRplDesc": null,
    "scheduledActiveDate": "yyyy-MM-dd",
    "replAction": null,
    "item": null,
    "diff1": null,
    "diff2": null,
    "diff3": null,
    "diff4": null,
    "dept": null,
    "class1": null,
    "subclass": null,
    "loc": null,
    "locType": null,
    "autoRangeInd": null,
    "activateDate": "yyyy-MM-dd",
    "deactivateDate": "yyyy-MM-dd",
    "presStock": null,
    "demoStock": null,
    "stockCat": null,
    "replOrderCtrl": null,
    "sourcingWh": null,
    "supplier": null,
    "originCountryId": null,
    "pickupLeadTime": null,
    "whLeadTime": null,
    "replMethodInd": null,
    "replMethod": null,
    "optimizeInd": null,
    "minStock": null,
    "maxStock": null,
    "incrPct": null,
    "minSupplyDays": null,
    "maxSupplyDays": null,
    "timeSupplyHorizon": null,
    "addLeadTimeInd": null,
    "invSellingDays": null,
    "serviceLevelType": null,
    "serviceLevel": null,
    "serviceLevelFloatingStd": null,
    "lostSalesFactor": null,
    "terminalStockQty": null,
    "seasonId": null,
    "phaseId": null,
  }
]
```

```
"rejectStoreOrdInd": null,  
"multRunsPerDayInd": null,  
"tsfZeroSohInd": null,  
"nonScalingInd": null,  
"maxScaleValue": null,  
"sizeProfileInd": null,  
"reviewCycle": null,  
"updateDaysInd": null,  
"mondayInd": null,  
"tuesdayInd": null,  
"wednesdayInd": null,  
"thursdayInd": null,  
"fridayInd": null,  
"saturdayInd": null,  
"sundayInd": null,  
"primaryPackNo": null,  
"defaultPackInd": null,  
"removePackInd": null,  
"mraUpdate": null,  
"mraRestore": null,  
"includeAdditionalDemand": null,  
"storeUserAccess": null,  
"storeToleranceType": null,  
"storeToleranceLimit": null}}
```

Output

Table 5-1291 ReplStatusRDO

Parameter Name	Data Type
statusMsg	String
failReplTable	List<ReplFailRDO>

Table 5-1292 ReplFailRDO

Parameter Name	Data Type
replAttrId	BigDecimal
item	String
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
loc	BigDecimal
locType	String
effectiveDate	Long
errorMsg	String

The output will contain the status of the request including validation errors, if any.

```
JSON Structure:  
{  
    "statusMsg": null,
```

```

    "failReplTable": [
      {
        "replAttrId": null,
        "item": null,
        "dept": null,
        "class1": null,
        "subclass": null,
        "loc": null,
        "locType": null,
        "effectiveDate": null,
        "errorMsg": null,
        "links": [],
        "hyperMediaContent": {
          "linkRDO": []
        }
      }
    ],
    "links": [],
    "hyperMediaContent": {
      "linkRDO": []
    }
  }
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
REPL_ATTR_UPDATE_HEAD	Yes	Yes	No	No
REPL_ATTR_UPDATE_ITEM	Yes	Yes	No	No
REPL_ATTR_UPDATE_LOC	Yes	Yes	No	No
SVC_PROCESS_TRACKER	Yes	Yes	Yes	No
CORESVC_REPL_ERR	No	Yes	No	No
SVC_REPL_ATTR_UPDATE	Yes	Yes	No	Yes
REPL_ITEM_LOC	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
V_STORE	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Manage Additional Demand in Replenishment Order Quantity (ROQ)

Functional Area

Replenishment

Business Overview

This service allows external systems to send the additional demand for the following replenishment methods: Time Supply, Time Supply - Issues, Time Supply - Seasonal,

Dynamic, Dynamic - Issues, Dynamic - Seasonal. If the item/Location has additional demand enabled for these forecast based replenishment methods, then the forecasted demand will be augmented based on the additional demand at the location and the replenishment order quantity (ROQ) generated will be based on the overall demand.

This management service supports both create and update actions. After all field validations are completed successfully, the additional demand record in the message will be inserted, if the demand record does not exist for the given item/location/date or, if the additional demand exists, the additional demand quantity and unit of measure (UOM) fields will be updated. If the message contains zero in the additional demand quantity field for an existing record, the service will delete this record from the Merchandising table.

Service Type

POST

ReST URL

MerchIntegrations/services/replenishment/additionalDemand/manage

Input Payload Details

Table 5-1293 Manage - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection.
items	Yes	Collection of Object	References a collection of additional demand quantities.

Table 5-1294 Items - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the Item for which additional demand needs to get added in ROQ.
locationType	Yes	String (1)	This indicates the type for the location. Valid values are S - Store and W - Warehouse.
location	Yes	Number (10)	Contains the location corresponding to the forecast information for the item.
demandDate	Yes	date	Contains the date for additional demand.
additionalDemandQuantity	Yes	Number (12,4)	This field contains the additional demand for the given item/location/date combination.
additionalDemandUom	No	String (4)	Indicates the unit of measure in which the additional demand is uploaded by the external system. When storing the additional demand in the database table, the additional demand is converted into item's standard UOM.

Table 5-1295 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Contains the Item for which additional demand needs to get added in ROQ.
locationType	No	String (1)	This indicates the type for the location. Valid values are S - Store and W - Warehouse.
location	No	Number (10)	Contains the location corresponding to the forecast information for the item.
demandDate	No	date	Contains the date for additional demand.
additionalDemandQuantity	No	Number (12,4)	This field contains the additional demand for the given item/location/date combination.
additionalDemandUom	No	String (4)	Indicates the unit of measure in which the additional demand is uploaded by the external system. When storing the additional demand in the database table, the additional demand is converted into item's standard UOM.
errors	No	Array of String	Holds the errors of the record.

Sample Input Message

```
{
  "collectionSize": 1,
  "items": [
    {
      "item": "101450060",
      "locationType": "S",
      "location": 1131,
      "demandDate": "2001-12-31",
      "additionalDemandQuantity": 60,
      "additionalDemandUom": "EA"
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of an error, the following standard error response is returned. The element `validationErrors` is present when input payload or input parameters do not follow the schema definition for this service. The element `businessError` is present if the payload passes schema validation but an exception is caught while processing the business logic.

Table 5-1296 ManageError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	No	String (25)	Contains the Item for which additional demand needs to get added in ROQ.
locationType	No	String (1)	This indicates the type for the location. Valid values are S - Store and W - Warehouse.
location	No	Number (10)	Contains the location corresponding to the forecast information for the item.
demandDate	No	date	Contains the date for additional demand.
additionalDemandQuantity	No	Number (12,4)	This field contains the additional demand for the given item/location/date combination.
additionalDemandUom	No	String (4)	Indicates the unit of measure in which the additional demand is uploaded by the external system. When storing the additional demand in the database table, the additional demand is converted into item's standard UOM.
errors	No	Array of String	Holds the errors of the record.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "item": "101450060",
      "locationType": "S",
      "location": 1131,
      "demandDate": "2001-12-31",
      "additionalDemandQuantity": 60,
      "additionalDemandUom": "EA",
      "errors": [
        "String"
      ]
    }
  ]
}
```

Delete Scheduled Replenishment Updates

This section describes the Delete Replenishment Schedule service.

Business Overview

This service deletes scheduled replenishments by calling the SVCPROV_REPL package to load input to the staging tables and then calling the core replenishment package to validate and delete data from the Merchandising tables.

Service Type

Post

ReST URL

inventory/replenishment/deleteReplSched

Input Parameters

Table 5-1297 ReplSchedDelRDO

Parameter Name	Data Type
replAttrId	BigDecimal
item	String
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
loc	BigDecimal
locType	String

JSON Structure:

```
[
  {
    "replAttrId": null,
    "item": null,
    "dept": null,
    "class1": null,
    "subclass": null,
    "loc": null,
    "locType": null
  }
]
```

Output

Table 5-1298 ReplStatusRDO

Parameter Name	Data Type
statusMsg	String
failReplTable	List<ReplFailRDO>

The output will contain the status of the request including validation errors, if any.

Table 5-1299 ReplFailRDO

Parameter Name	Data Type
replAttrId	BigDecimal
item	String
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
loc	BigDecimal
locType	String
effectiveDate	Long
errorMsg	String

JSON Structure:

```
{
  "statusMsg": null,
  "failReplTable": [
    {
      "replAttrId": null,
      "item": null,
      "dept": null,
      "class1": null,
      "subclass": null,
      "loc": null,
      "locType": null,
      "effectiveDate": null,
      "errorMsg": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
REPL_ATTR_UPDATE_HEAD	Yes	Yes	No	Yes
REPL_ATTR_UPDATE_ITEM	Yes	Yes	No	Yes
REPL_ATTR_UPDATE_LOC	Yes	Yes	No	Yes
SVC_PROCESS_TRACKER	Yes	Yes	Yes	No
CORESVC_REPL_ERR	No	Yes	No	No
SVC_REPL_ATTR_UPDATE	Yes	Yes	No	Yes
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Update Scheduled Replenishment Updates

This section describes the Modify Replenishment service.

Business Overview

This service modifies scheduled replenishments by calling the SVCPROV_REPL package to load input to the staging tables and then calling the core replenishment package to validate and process data to the Merchandising tables.

Service Type

Post

ReST URL

inventory/replenishment/modifyReplSched

Input Parameters

Table 5-1300 ReplSchedCreModRDO

Parameter Name	Data Type
replAttrId	BigDecimal
schRplDesc	String
scheduledActiveDate	Date
replAction	String
item	String
diff1	String
diff2	String
diff3	String
diff4	String
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal

Table 5-1300 (Cont.) ReplSchedCreModRDO

Parameter Name	Data Type
loc	BigDecimal
locType	String
autoRangeInd	String
activateDate	Date
deactivateDate	Date
presStock	BigDecimal
demoStock	BigDecimal
stockCat	String
replOrderCtrl	String
sourcingWh	BigDecimal
supplier	BigDecimal
originCountryId	String
pickupLeadTime	BigDecimal
whLeadTime	BigDecimal
replMethodInd	String
replMethod	String
optimizeInd	String
minStock	BigDecimal
maxStock	BigDecimal
incrPct	BigDecimal
minSupplyDays	BigDecimal
maxSupplyDays	BigDecimal
timeSupplyHorizon	BigDecimal
addLeadTimeInd	String
invSellingDays	BigDecimal
serviceLevelType	String
serviceLevel	BigDecimal
serviceLevelFloatingStd	String
lostSalesFactor	BigDecimal
terminalStockQty	BigDecimal
seasonId	BigDecimal
phaseId	BigDecimal
rejectStoreOrdInd	String
multRunsPerDayInd	String
tsfZeroSohInd	String
nonScalingInd	String
maxScaleValue	BigDecimal
sizeProfileInd	String
reviewCycle	String

Table 5-1300 (Cont.) ReplSchedCreModRDO

Parameter Name	Data Type
updateDaysInd	String
mondayInd	String
tuesdayInd	String
wednesdayInd	String
thursdayInd	String
fridayInd	String
saturdayInd	String
sundayInd	String
primaryPackNo	String
defaultPackInd	String
removePackInd	String
mraUpdate	String
mraRestore	String
includeAdditionalDemand	String
storeUserAccess	String
storeToleranceType	String
storeToleranceLimit	BigDecimal

JSON Structure:

```
[
  {
    "replAttrId": null,
    "schRplDesc": null,
    "scheduledActiveDate": "yyyy-MM-dd",
    "replAction": null,
    "item": null,
    "diff1": null,
    "diff2": null,
    "diff3": null,
    "diff4": null,
    "dept": null,
    "class1": null,
    "subclass": null,
    "loc": null,
    "locType": null,
    "autoRangeInd": null,
    "activateDate": "yyyy-MM-dd",
    "deactivateDate": "yyyy-MM-dd",
    "presStock": null,
    "demoStock": null,
    "stockCat": null,
    "replOrderCtrl": null,
    "sourcingWh": null,
    "supplier": null,
    "originCountryId": null,
```



```
"pickupLeadTime": null,
"whLeadTime": null,
"replMethodInd": null,
"replMethod": null,
"optimizeInd": null,
"minStock": null,
"maxStock": null,
"incrPct": null,
"minSupplyDays": null,
"maxSupplyDays": null,
"timeSupplyHorizon": null,
"addLeadTimeInd": null,
"invSellingDays": null,
"serviceLevelType": null,
"serviceLevel": null,
"serviceLevelFloatingStd": null,
"lostSalesFactor": null,
"terminalStockQty": null,
"seasonId": null,
"phaseId": null,
"rejectStoreOrdInd": null,
"multRunsPerDayInd": null,
"tsfZeroSohInd": null,
"nonScalingInd": null,
"maxScaleValue": null,
"sizeProfileInd": null,
"reviewCycle": null,
"updateDaysInd": null,
"mondayInd": null,
"tuesdayInd": null,
"wednesdayInd": null,
"thursdayInd": null,
"fridayInd": null,
"saturdayInd": null,
"sundayInd": null,
"primaryPackNo": null,
"defaultPackInd": null,
"removePackInd": null,
"mraUpdate": null,
"mraRestore": null
"storeUserAccess": null,
"storeToleranceType": null,
"storeToleranceLimit": null}}]
```

Output

Table 5-1301 ReplStatusRDO

Parameter Name	Data Type
statusMsg	String
failReplTable	List<ReplFailRDO>

Table 5-1302 ReplFailRDO

Parameter Name	Data Type
replAttrId	BigDecimal
item	String
dept	BigDecimal
class1	BigDecimal
subclass	BigDecimal
loc	BigDecimal
locType	String
effectiveDate	Long
errorMsg	String

The output will contain the status of the request including validation errors, if any.

JSON Structure:

```
{
  "statusMsg": null,
  "failReplTable": [
    {
      "replAttrId": null,
      "item": null,
      "dept": null,
      "class1": null,
      "subclass": null,
      "loc": null,
      "locType": null,
      "effectiveDate": null,
      "errorMsg": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
REPL_ATTR_UPDATE_HEAD	Yes	Yes	No	No
REPL_ATTR_UPDATE_ITEM	Yes	Yes	No	No
REPL_ATTR_UPDATE_LOC	Yes	Yes	No	No
SVC_PROCESS_TRACKER	Yes	Yes	Yes	No
CORESVC_REPL_ERR	No	Yes	No	No
SVC_REPL_ATTR_UPDATE	Yes	Yes	No	Yes
REPL_ITEM_LOC	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
V_STORE	Yes	No	No	No
V_WH	Yes	No	No	No
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Sales

The following services are included in this functional area:

- [Sales Download Service](#)
 - [Get Item Location Weekly Sales \(deprecated\)](#)
 - [Get Item Location Weekly Sales](#)
- [Store Day User Upload Service](#)

Sales Download Service

The following services are included in this functional area:

- [Get Item Location Weekly Sales \(deprecated\)](#)
- [Get Item Location Weekly Sales](#)

Get Item Location Weekly Sales (deprecated)

Functional Area

Sales

Business Overview

This service retrieves weekly item sales information for store locations or inventory issues for warehouse locations from the Item Location History table (ITEM_LOC_HIST). Given the high volume of data in this table, it's recommended to use this service for retrieving filtered information using the available query parameters. This service requires at least one of the following query parameter to be included in the request

End of Week Date

Item

Location

The optional query parameter of 'Include Late Posted Transaction' (valid values Y/N) is applicable if End of Week Date query parameter is passed. If this is passed as 'Y', the service will also retrieve late posted transactions from previous weeks that were posted within the requested End of Week period. This option is recommended when requesting information for the currently open week or recently closed week.

Table 5-1303 Input Parameters

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset Key
eowDate	No	String	End of Week Date (YYYY-MM-DD)
location	No	String (10)	Location
locationType	No	String	Location Type
item	No	String	Item
dept	No	String (4)	Dept
class	No	String (4)	Class
subclass	No	String (4)	Subclass
salesType	No	String	Sales Type
year	No	String (4)	Year 454
month	No	String	Month 454
includeLatePosted	No	String	Include Late Posted Transaction
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Service Type

GET

ReST URL

MerchIntegrations/services/sales/itemLocation/weekly

Input Payload Details

Table 5-1304 Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
item	Yes	String (25)	Contains the item number in the distro.
location	Yes	Number (10)	This contains the destination location of the store order.

Table 5-1304 (Cont.) Get - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationType	Yes	String (1)	This contains the destination location type of the store order.

Sample Input Message

```
{
  "item": "100100076",
  "location": 6000,
  "locationType": "S"
}
```

Response Code: 200 (Success)

Table 5-1305 PageResultItemsItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1306 ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
eowDate	No	date	This field contains the end of the week date of the week for the sales total and other calculations. Format YYYY-MM-DD
location	No	Number	This field contains numeric identifier of the location. This field may contain a store, warehouse.

Table 5-1306 (Cont.) ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
locationType	No	String	This field specifies type of location in the location field. Valid values are S - Store, W - Warehouse.
item	No	String	This field specifies unique alphanumeric value that identifies the item
salesType	No	String	This field indicates the type of sale or inventory movement. Valid Values are defined in code detail with code type SIST. Default seeded values are I - Issues From Warehouses to Stores, C - Clearance, P - Promotional, R - Regular
salesIssues	No	Number	This field contains the number of units sold or transferred for item/location/week/sales type combination. If the location type is a store, then this value represents sales. If the location type is a warehouse, then this value represents outbound transfer/allocation transactions to stores.
value	No	Number	This field contains the retail value of the sales for the item/location/week/sales type combination. This field is stored in the local currency. This field will only hold a value for stores.

Table 5-1306 (Cont.) ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
grossProfit	No	Number	This field contains the total gross profit calculated for the item/location/week/sales type. This is updated for each sale that occurs during the week and is calculated as (sales retail - cost of sales). This field will only hold a value for stores.
stock	No	Number	This field contains the stock on hand for the item/location combination at the end of the week.
retail	No	Number	This field contains the unit retail price of the item at the location at the end of the week. This field is stored in the local currency.
averageCost	No	Number	This field contains the average cost of the item at the location at the end of the week. This field is stored in the local currency.
year	No	Number	This field contain the identifier of the year in which the transactions occurred based on the 454 calendar
month	No	Number	This field contain the identifier of the month in which the transactions occurred based on the 454 calendar
week	No	Number	This field contain the identifier of the week in which the transactions occurred based on the 454 calendar
dept	No	Number	This field uniquely identifies the department
class	No	Number	This field identifies the class within the department

Table 5-1306 (Cont.) ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
subclass	No	Number	This field identifies the subclass within the department and class combination
postDate	No	date	Holds the business date the record is inserted or last updated. Format YYYY-MM-DD

Table 5-1307 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "eowDate": "2001-12-31",
      "location": 10051,
      "locationType": "S",
      "item": "110023212",
      "salesType": "R",
      "salesIssues": 123,
      "value": 3823.83,
      "grossProfit": 8.6,
      "stock": 25,
      "retail": 29.95,
      "averageCost": 17.2,
      "year": 2023,
      "month": 1,
      "week": 4,
      "dept": 1031,
      "class": 1,
      "subclass": 1,
      "postDate": "2001-12-31"
    }
  ],
  "hasMore": true,
}
```



```
"limit": 1,
"count": 1,
"links": [
{
"href": "https://<server>:<port>/Application/services/resources?limit=1000",
"rel": "self"
}
]
}
```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service.

Sample Error Message

```
{
"status": "ERROR",
"message": "Error found in validation of input payload",
"validationErrors": [
{
"error": "must be one of Y, N",
"field": "createRecord.arg0.approveInd",
"inputValue": "X"
}
]
}
```

Get Item Location Weekly Sales

Functional Area

Sales

Business Overview

This service retrieves weekly item sales information for store locations or inventory issues for warehouse locations from the Item Location History table (ITEM_LOC_HIST). Given the high volume of data in this table, it's recommended to use this service for retrieving filtered information using the available query parameters. This service requires at least one of the following query parameter to be included in the request

End of Week Date

Item

Location

The optional query parameter of 'Include Late Posted Transaction' (valid values Y/N) is applicable if End of Week Date query parameter is passed. If this is passed as 'Y', the service will also retrieve late posted transactions from previous weeks that were posted within the requested End of Week period. This option is recommended when requesting information for the currently open week or recently closed week.

Service Type

GET

ReST URL

MerchIntegrations/services/sales/itemLocation/weekly

Input Parameters

Parameter Name	Required	Data Type	Description
limit	No	String (6)	Pagination limit. Default value is 1000.
offsetkey	No	String	Offset Key
eowDate	No	String	End of Week Date (YYYY-MM-DD)
location	No	String (10)	Location
locationType	No	String	Location Type
item	No	String	Item
dept	No	String (4)	Dept
class	No	String (4)	Class
subclass	No	String (4)	Subclass
salesType	No	String	Sales Type
year	No	String (4)	Year 454
month	No	String	Month 454
includeLatePosted	No	String	Include Late Posted Transaction
include	No	String	Fields to include - comma separated (Example items.name,items.address.city)
exclude	No	String	Fields to exclude - comma separated (Example items.address.contactFax,items.customFlexAttribute)

Response Code: 200 (Success)

Table 5-1308 PageResultItemsItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
items	No	Collection of Object	Collection of records

Table 5-1308 (Cont.) PageResultItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
hasMore	No	Boolean	Boolean to signify if there are additional pages of data. This is true if limit = count
limit	No	Number	Requested page size. If the page size was not defined in the input request, a default page size is used to limit the number of records to fetch
count	No	Number	Number of record in the response object
links	No	Collection of Object	List of Self and/or Next URL

Table 5-1309 ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
eowDate	No	date	This field contains the end of the week date of the week for the sales total and other calculations. Format YYYY-MM-DD
location	No	Number	This field contains numeric identifier of the location. This field may contain a store, warehouse.
locationType	No	String	This field specifies type of location in the location field. Valid values are S - Store, W - Warehouse.
item	No	String	This field specifies unique alphanumeric value that identifies the item

Table 5-1309 (Cont.) ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
salesType	No	String	This field indicates the type of sale or inventory movement. Valid Values are defined in code detail with code type SIST. Default seeded values are I - Issues From Warehouses to Stores, C - Clearance, P - Promotional, R - Regular
salesIssues	No	Number	This field contains the number of units sold or transferred for item/location/week/sales type combination. If the location type is a store, then this value represents sales. If the location type is a warehouse, then this value represents outbound transfer/allocation transactions to stores.
value	No	Number	This field contains the retail value of the sales for the item/location/week/sales type combination. This field is stored in the local currency. This field will only hold a value for stores.
grossProfit	No	Number	This field contains the total gross profit calculated for the item/location/week/sales type. This is updated for each sale that occurs during the week and is calculated as (sales retail - cost of sales). This field will only hold a value for stores.
stock	No	Number	This field contains the stock on hand for the item/location combination at the end of the week.

Table 5-1309 (Cont.) ItemLocHist - Object. See list of elements for detail

Element Name	Required	Data Type	Description
retail	No	Number	This field contains the unit retail price of the item at the location at the end of the week. This field is stored in the local currency.
averageCost	No	Number	This field contains the average cost of the item at the location at the end of the week. This field is stored in the local currency
year	No	Number	This field contain the identifier of the year in which the transactions occurred based on the 454 calendar
month	No	Number	This field contain the identifier of the month in which the transactions occurred based on the 454 calendar
week	No	Number	This field contain the identifier of the week in which the transactions occurred based on the 454 calendar
dept	No	Number	This field uniquely identifies the department
class	No	Number	This field identifies the class within the department
subclass	No	Number	This field identifies the subclass within the department and class combination
postDate	No	date	Holds the business date the record is inserted or last updated. Format YYYY-MM-DD

Table 5-1310 LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
href	No	String	Service URL

Table 5-1310 (Cont.) LinkItem - Object. See list of elements for detail

Element Name	Required	Data Type	Description
rel	No	String	This can be either 'self' or 'next'. The 'next' link is present if there are more pages to fetches and contains the offsetkey which unique identifies the last record of this response.

Sample Response Message

```
{
  "items": [
    {
      "eowDate": "2001-12-31",
      "location": 10051,
      "locationType": "S",
      "item": "110023212",
      "salesType": "R",
      "salesIssues": 123,
      "value": 3823.83,
      "grossProfit": 8.6,
      "stock": 25,
      "retail": 29.95,
      "averageCost": 17.2,
      "year": 2023,
      "month": 1,
      "week": 4,
      "dept": 1031,
      "class": 1,
      "subclass": 1,
      "postDate": "2001-12-31"
    }
  ],
  "hasMore": true,
  "limit": 1,
  "count": 1,
  "links": [
    {
      "href": "https://<server>:<port>/Application/services/resources?limit=1000",
      "rel": "self"
    }
  ]
}
```

Store Day User Upload Service

This section describes the Store Day User service.

Functional Area

Financials

Business Overview

The primary role of this service is to create or delete store day user and send them to Merchandising.

Create Store Day User

Business Overview

This service creates store day user by calling the SVCPROV_STOREDAYUSER package to load input data to the staging tables and then calling the core store day user package to validate and insert data to the Merchandising tables.

Service Type

Post

ReSTURL

financials/StoreDayUserREST/create StoreDayUser

Input Parameters

SvcprovSdudescRecRDO

Parameter Name	Data Type
store	BigDecimal
businessDate	String
userId	String

JSON Structure

```
[{"store": null,  
  "businessDate": null,  
  "userId": null}]
```

BusinessDate will take input in string with the format as 'DD-MON-YYYY' and later converted to Timestamp format and insert in table.

Output

SvcprovSduStatusRecRDO

Parameter Name	Data Type
statusMsg	String
sduErrTbl	List< SvcprovFailSduRecRDO >

SvcprovFailSduRecRDO

Parameter Name	Data Type
store	BigDecimal
businessDate	String
userId	String
errorMsg	String

The output will contain the status of the request including validation errors, if any.

JSON Structure:

```
{
  "statusMsg": null,
  " sduErrTbl ": [
    {
      " store": null,
      " businessDate": null,
      " userId": null,
      "errorMsg": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ],
  "links": [],
  "hyperMediaContent": {
    "linkRDO": []
  }
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
STORE_DAY_USER	Yes	Yes	No	No
SVC_STORE_DAY_USER	Yes	Yes	Yes	Yes
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Delete Store Day User

Business Overview

This service deletes store day user by calling the SVCPROV_SDU package to load input data to the staging tables and then calling the core store day user package to validate and delete data to the Merchandising tables.

Service Type

Post

ReSTURL

financials/StoreDayUserREST/delete StoreDayUser

Input Parameters

SvcprovSdudescRecRDO

Parameter Name	Data Type
store	BigDecimal
businessDate	String
userId	String

JSON Structure

```
[{"store": null,
  "businessDate": null,
  "userId": null}]
```

BusinessDate will take input in string with the format as 'DD-MON-YYYY' and later converted to Timestamp format and delete from table.

Output

SvcprovSduStatusRecRDO

Parameter Name	Data Type
statusMsg	String
sduErrTbl	List< SvcprovFailSduRecRDO >

SvcprovFailSduRecRDO

Parameter Name	Data Type
store	BigDecimal
businessDate	String
userId	String
errorMsg	String

The output will contain the status of the request including validation errors, if any.

JSON Structure

```
{
  "statusMsg": null,
  "sduErrTbl": [
    {
      "store": null,
      "businessDate": null,
      "userId": null,
      "errorMsg": null,
      "links": [],
      "hyperMediaContent": {
        "linkRDO": []
      }
    }
  ]
}
```

```

    }
  }
],
"links": [],
"hyperMediaContent": {
  "linkRDO": []
}
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
STORE_DAY_USER	Yes	No	No	Yes
SVC_STORE_DAY_USER	Yes	Yes	Yes	Yes
JOB_AUDIT_LOGS	No	Yes	No	No
JOB_AUDIT_PAYLOAD	No	Yes	No	No

Upload and Process Sales

Functional Area

Sales - Sales Processing

Business Overview

This ReST service allows for sales from an external Sales Audit or Point of Sale system to be uploaded directly into Merchandising and provides an alternative solution to existing POSU file upload solution. The validations imposed by this is similar to those enforced by the upload sales process that loads sales data from Sales Audit into Merchandising. This service expects each message to have transaction for a single store location. It supports Item level taxes and promotion or discount details, and determines posting and processing locations using existing configuration.

Service Type

POST

ReST URL

MerchIntegrations/services/sales/uploadAndProcess

Input Payload Details

Table 5-1311 UploadAndProcess - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The unique identifier of the store.
salesTransactions	Yes	Collection of Object	

Table 5-1312 SalesTransactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
transactionSeqNo	Yes	Number (10)	Unique Sequence Number for each sales transaction line.
transactionDate	Yes	date	Date sale/return transaction was processed at the POS
item	Yes	String (25)	The ID number of an ITM or REF
transactionType	Yes	String (1)	Transaction type code to specify whether transaction is a sale or a return
totalSalesQuantity	Yes	Number (12,4)	number of units sold at a particular location
sellingUom	Yes	String (4)	UOM at which this item was sold
totalSalesValue	Yes	Number (20,4)	net sales value of goods sold
actualWeightQuantity	No	Number (12,4)	The actual weight of the item, only populated for catchweight Items
totalTaxValue	Yes	Number (20,4)	This will indicate total of all Tax amount for the item.
noInventoryReturnInd	No	String (1)	This field contains an indicator that identifies a return without inventory. This is generally a non-required column, but in case of Returns, this is required.
returnDisposition	No	String (10)	This field will contain the disposition code published by RWMS as part of the Returns upload to OMS.
returnWarehouse	No	Number (10)	This field contains the physical warehouse ID for the warehouse identifier where the item was returned.

Table 5-1312 (Cont.) SalesTransactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
salesType	Yes	String (1)	This field indicates whether the line item is a Regular Sale, a customer order serviced by OMS (External CO) or a customer order serviced by a store (In Store CO).
customerOrderNo	No	String (48)	This field contains the customer order number ID.
fulfillOrderNo	No	String (48)	This field contains the fulfillment order number ID.
fulfillLocationType	No	String (2)	This field contains the fulfillment location type. and Valid values are found under the FLTP code type, for e.g In Store (ST), Supplier (SU), Warehouse (WH).
fulfillLocation	No	String (10)	This field contains the fulfillment loc ID.
dropShipmentInd	No	String (1)	Indicates whether the transaction is a drop shipment or not. If it is a drop shipment, indicator will be 'Y'. This field is not required, but will be defaulted to 'N' if blank
origStore	No	String (10)	This field contains the original store value for a Return transaction.
postingStore	No	String (10)	This field contains the store at which the item sale/return should be accounted for in case of cross-store sales happening at co-located stores. It is expected that this field will be populated only for items that are checked out at a different store from the one at which they are originally managed.

Table 5-1312 (Cont.) SalesTransactions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
posTranId	No	Number (20)	This field contains the unique identifier for a sale transaction. This is an Optional field.
consignmentUnitCost	No	String (20)	This field contains the consignment unit cost that should be applied while posting the sales/returns to Merchandising.
consignmentRate	No	String (12)	This field contains the consignment rate that should be applied while posting the sales/returns to Merchandising.
inventoryIdentifierType	No	String (6)	This field contains the inventory identifier type passed in Sales/ Return transactions. Valid values are found under the Inventory Identifier Types (IIDT) code type, for e.g Lot (L), Expiry Date (E), Import Document (D).
inventoryId	No	String (120)	This field contains the inventory id value being passed in sales/ return transactions.
taxes	No	Collection of Object	
promotions	No	Collection of Object	

Table 5-1313 Taxes - Object. See list of elements for detail

Element Name	Required	Data Type	Description
taxCode	Yes	String (6)	The Tax Code of the item
taxRate	Yes	Number (20,10)	The tax rate of the item.
taxAmount	Yes	Number (20,4)	The item level tax or prorated transaction level tax of the item .

Table 5-1314 Promotions - Object. See list of elements for detail

Element Name	Required	Data Type	Description
promoTranType	Yes	String (5)	This field contains the Promotion Sales Types and Valid values are found under the PRMT code type, for e.g In Store Discount (1004), Employee Discount (1005), External Promotion (2000), Internal Promotion (9999).
promoId	No	Number (10)	Promotion number from Merchandising
promoComponent	No	Number (10,4)	Links the promotion to additional pricing attributes. This contains the OFFER_ID from Pricing.
salesQuantity	Yes	Number (12,4)	Sales quantity sold for this promotion type
salesValue	Yes	Number (20,4)	Sales value for this promotion type
discountValue	Yes	Number (20,4)	Discount value for this promotion type

Table 5-1315 UploadAndProcessError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	Input store Value.
transactionSeqNo	Yes	Number (10)	Input Unique Sequence Number.
errorMessage	Yes	Array of String	

Sample Input Message

```
{
  "store": 1521,
  "salesTransactions": [
    {
      "transactionSeqNo": 1,
      "transactionDate": "2001-12-31",
      "item": "100100076",
      "transactionType": "S",
      "totalSalesQuantity": 21,
      "sellingUom": "EA",
      "totalSalesValue": 210,
      "actualWeightQuantity": null,
      "totalTaxValue": 13,
      "noInventoryReturnInd": null,

```

```

"returnDisposition": null,
"returnWarehouse": null,
"salesType": "R",
"customerOrderNo": null,
"fulfillOrderNo": null,
"fulfillLocationType": null,
"fulfillLocation": null,
"dropShipmentInd": "N",
"origStore": null,
"postingStore": null,
"posTranId": 625,
"consignmentUnitCost": null,
"consignmentRate": null,
"inventoryIdentifierType": null,
"inventoryId": null,
"taxes": [
{
"taxCode": "S",
"taxRate": 1,
"taxAmount": 1
}
],
"promotions": [
{
"promoTranType": "1004",
"promoId": null,
"promoComponent": null,
"salesQuantity": 1,
"salesValue": 1,
"discountValue": 1
}
]
]
}

```

Response Code: 200 (Success)

Sample Response Message

```

{
  "status": "SUCCESS"
}

```

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element "validationErrors" will be present when input payload or input parameters are not as per the schema definition of this service. The element "businessError" will be present if the payload passes schema validation but exception is caught while processing business logic.

Table 5-1316 UploadAndProcessError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	Input store Value.
transactionSeqNo	Yes	Number (10)	Input Unique Sequence Number.
errorMessage	Yes	Array of String	

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "store": 1521,
      "transactionSeqNo": 1,
      "errorMessage": [
        "u0027Invalid Itemu0027"
      ]
    }
  ]
}
```

Fiscal Document Generation Services

The following services are included in this functional area:

- [Fiscal Pre Document Request Service](#)
- [Fiscal Document Request Service](#)
- [Fiscal Document Status Detail Request Service](#)
- [Next Fiscal Document Request Service](#)
- [Fiscal Document Process Status Request Service](#)
- [Fiscal Document Cancel Request Service](#)
- [Fiscal Document Detail Request Service](#)

Fiscal Pre Document Request Service

Functional Area

Finance

Business Overview

This web service supports fiscal document requests from inventory systems. It has a simplified payload for the fiscal document request. Once the shipment system calls this service, Fiscal Document Generation will return an internal ID for the request. This ID can later be used by the shipment systems to request the document(s) created for the specific shipment.

Service Type

Post

ReST URL

{{baseUrl}}/services/private/fdg/processPreDoc

Input

Parameter Name	Required	Data Type	Description	Valid values
requestSystem	Yes	String	This field contains the identification name of system that is requesting a fiscal document. This name is provided by system that is requesting.	
docType	Yes	String	This field contains the type of INV, WBL. the document. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDTY.	
extReqId	No	String	This field contains the external request ID. This is the internal number from system that is requesting a fiscal document. For example: asn_nbr ID from shipments.	
issueDate	Yes	OffsetDateTime	This field contains the issue date of document in the format YYYY-MM-DDThh:mm:ssTZD (for example, 1997-07-16T19:20:30+01:00) equivalent to ISO 8601.	

Parameter Name	Required	Data Type	Description	Valid values
transactionType	Yes	String	This field contains the code type of the transaction. Valid values are codes from CODE_DETAIL with a CODE_TYPE of GTRT.	ITMTAX, PO TSF, DEAL, RTV, POSSAL, SALES, TAXINF, FO, FR
currencyCode	Yes	String	This field contains the currency code for the fiscal document. It is based on currency code defined in shipment location that is going to ship goods.	
entities	Yes	List<RestFdgEtt>	Collection of entities.	
details	Yes	List<RestFdgDtl>	Collection of item details.	
comments	No	List<RestFdgText>	Collection of comments.	
references	No	List<RestFdgRef>	Collection of references.	

RestFdgEtt

The *fiscalDocument/entities* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
entityType	Yes	String	This field contains the entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDED.	SHIPP, DEST, VENDOR, CR, LOC
entityInternalType	No	String	This field contains the Merchandising entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDET.	WH, ST, SUPP, PTNR, CUST
entityInternalPtnr Type	No	String	This field contains the Merchandising partner type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of PTNR.	AG, AP, BK, BR, CN, CO, E, EC, ES, EV, FA, FF, IA, IM, S1, S2, S3
entityCode	No	String	This field contains the entity's Merchandising internal code. For example, ID of store or warehouse.	

Parameter Name	Required	Data Type	Description	Valid values
name	No	String	This field contains the name responsible for transportation. This field is used only when there is no entity code defined for transportation. It's used by SIM shipments because there is no transportation entity.	
carrierVehicleRn	No	String	This field contains the vehicle registration number responsible for transportation.	
carrierVehicleState	No	String	This field contains the state of the country where vehicle, responsible for transportation, is registered.	
carrierDriverName	No	String	This field contains the driver name responsible for transportation.	
carrierDriverLn	No	String	This field contains the information about driver license ID responsible for transportation.	

RestFdgDtl

The *fiscalDocument/details* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
item	Yes	String	This field contains the unique code ID that identifies the item.	
uom	Yes	String	This field contains the unit of measure used in item in the fiscal document.	
qty	Yes	BigDecimal	This field contains the quantity of item in this fiscal document.	
references	No	List<RestFdgRef>	Collection of item references	

RestFdgRef

The *fiscalDocument/details/references* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
refType	Yes	String	This field contains the type of the reference at item level. It is used when there is any reference that it is necessary to add in this fiscal document, as shipment number, contract number, order, id, and so on. Valid values: CODE_DETAIL – FDRT.	FDOC, ORDER, SHIPN, FULLF, ALLOC, DISTRO, BOLN, TSFNO, RTVNO, CONO , RTVAUT, CTRC
refCode	Yes	String	This field contains the reference code/value of reference type at item level.	

RestFdgText

The *fiscalDocument/comments* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
text	Yes	String	This field contains additional comments at item level. It is used to provide any additional information or legal message important to fiscal document.	
textGroupId	No	String	Text group ID. It is used to group related comments when there is more than one or when the size of a message is bigger than the size of the text field and it is necessary to generate more than one line.	
textGroupSeq	No	String	Sequence of the text for a given group ID. It is used in case comments information has more than one line.	

JSON Input Structure:

```
"{
  "fiscalDocument": {
    "requestSystem": "SIOCS",
    "docType": "WBL",
    "extReqId": "10001",
    "transactionType": "TSF",
    "issueDate": "2020-12-16T08:19:42-05:00",
```

```

"currencyCode": "USD",
"entities": [
  {
    "entityType": "SHIPP",
    "entityInternalType": "WH",
    "entityCode": "252525"
  },
  {
    "entityType": "DEST",
    "entityInternalType": "ST",
    "entityCode": "9625518"
  },
  {
    "entityType": "CR",
    "name": "UPS",
    "carrierVehicleRn": "IF5597",
    "carrierVehicleState": "SP",
    "carrierDriverName": "Driver Name",
    "carrierDriverLn": "asasdd984955456"
  }
],
"details": [
  {
    "item": "100450025",
    "uom": "EA",
    "qty": 5,
    "references": [
      {
        "refType": "TSFNO",
        "refCode": "897889"
      }
    ]
  }
],
"comments": [{
  "text": "Additional comments to the document. If it's too l'",
  "textGroupId": "1",
  "textGroupSeq": "1"
},
{
  "text": "ong, it can end in other line",
  "textGroupId": "1",
  "textGroupSeq": "2"
}
],
"references": [
  {
    "refType": "DOC",
    "refCode": "9848945651"
  }
]
}
}

```

Output

Parameter Name	Data Type	Description
requestId	String	This field contains the unique identifier of the document request.
requestSystem	String	This column contains the unique identifier of the requester.
requestAction	String	This column contains the current action for the request.
docId	BigDecimal	This column contains the unique document ID.
docNo	String	This column contains the unique document number.
docType	String	This column contains the type of the document.
status	String	This column contains the status of the document.
extReqId	String	This field contains the external request ID. This is the internal number from system that is requesting a fiscal document. Example: asn_nbr ID from shipments.
docPrintUrl	String	This column contains the url to access the fiscal document issued by government. It can be a url from third-party repository responsible to communicate with government or other url.
additionalproperties	List<RestFdgExt>	Collection of additional properties.
errors	List<RestFdgError>	Collection of errors.

RestFdgExt

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.

RestFdgError

Parameter Name	Data Type	Description
errorCode	String	This field contains the error code.
errorDesc	String	This field contains the error description.

JSON Structure:

```
"{
  "requestId": 62,
  "requestSystem": "SIOCS",
  "requestAction": "A",
  "docId": 62,
  "docNo": null,
  "docType": "WBL",
  "status": "N",
  "extReqId": "10003",
  "docPrintUrl": null,
  "additionalproperties": [],
  "errors": []
}"
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
FDG_ATTRIB_MAP	Yes	No	No	No
FDG_ATTRIB_MAP_NM	Yes	No	No	No
SVC_FDG_DTL	Yes	Yes	Yes	No
SVC_FDG_DTL_PACK	Yes	Yes	Yes	No
SVC_FDG_ERROR	Yes	Yes	No	No
SVC_FDG_ETT	Yes	Yes	Yes	No
SVC_FDG_EXT	Yes	Yes	Yes	No
SVC_FDG_HDR	Yes	Yes	Yes	No
SVC_FDG_NON_MERCH	Yes	Yes	Yes	No
SVC_FDG_REF	Yes	Yes	No	No
SVC_FDG_TAX	Yes	Yes	No	Yes
SVC_FDG_TEXT	Yes	Yes	No	No
CODE_DETAIL	Yes	No	No	No
V CODE_DETAIL	Yes	No	No	No
STORE	Yes	No	No	No
PARTNER	Yes	No	No	No
WH	Yes	No	No	No
LANG	Yes	No	No	No
FDG_SYSTEM_OPTIONS	Yes	No	No	No
CFA_ATTRIB	Yes	No	No	No
CFA_ATTRIB_MAP	Yes	No	No	No
CFA_ATTRIB_MAP_LABES	Yes	No	No	No
RTV_DETAIL	Yes	No	No	No
ORDCUST	Yes	No	No	No
ORDCUST_DETAIL	Yes	No	No	No
TSFHEAD	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
TSFDETAIL	Yes	No	No	No
SHIPSKU	Yes	No	No	No
TSFDETAIL_CHRG	Yes	No	No	No
ALLOC_CHRG	Yes	No	No	No
NON_MERCH_CODE_COMP	Yes	No	No	No
V_NON_MERCH_CODE_HEAD_TL	Yes	No	No	No
ADDR	Yes	No	No	No
PACKITEM_BREAKOUT	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No
GTS_CALC_INPUT_STG	Yes	Yes	No	No
GTS_CALC_OUTPUT_STG	Yes	No	No	No

Fiscal Document Request Service

Functional Area

Finance

Business Overview

This web service supports fiscal document requests from any system based on a comprehensive payload with a detailed level of input parameters. This web service supports inbound transactions such as PO receiving. Once the requester system calls for this service, Fiscal Document Generation will return an internal ID for the request. This ID can later be used by these systems to request the document(s) created.

Service Type

Post

ReST URL

{{baseUrl}}/services/private/fdg/processDoc

Input

Parameter Name	Required	Data Type	Description	Valid values
requestSystem	Yes	String	This field contains the identification name of system that is requesting a fiscal document. This name is provided by system that is requesting.	
docType	Yes	String	This field contains the type of the document. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDTY.	INV, WBL
extReqId	No	String	This field contains the external request ID. This is the internal number from system that is requesting a fiscal document. For example, asn_nbr ID from shipments	
issueDate	Yes	OffsetDateTime	This field contains the issue date of document in the format YYYY-MM-DDThh:mm:ssTZD (for example, 1997-07-16T19:20:30+01:00) equivalent to ISO 8601.	
transactionType	Yes	String	This field contains the code type of the transaction. Valid values are codes from CODE_DETAIL with a CODE_TYPE of GTRT.	ITMTAX, PO, TSF, DEAL, RTV, POSSAL, SALES, TAXINF, FO, FR
currencyCode	Yes	String	This field contains the currency code for the fiscal document. It is based on currency code defined in shipment location that is going to ship goods.	
deduceTax	Yes	String	This field contains the indicator to control in this request, if taxes is sent from request system and any calculation is needed or request system not send taxes and it is necessary to call tax calculation to fill taxes information.	Y or N
entities	Yes	List<RestFdgEtt>	Collection of entities.	
details	Yes	List<RestFdgDtl>	Collection of item details.	

Parameter Name	Required	Data Type	Description	Valid values
comments	No	List<RestFdgText>	Collection of comments.	
references	No	List<RestFdgRef>	Collection of references.	

RestFdgEtt:

The *fiscalDocument/entities* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
entityType	Yes	String	This field contains the entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDED.	SHIPP, DEST, VENDOR, CR, LOC
entityInternalType	No	String	This field contains the Merchandising entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDET.	WH, ST, SUPP, PTNR, CUST
entityInternalPtnr Type	No	String	This field contains the Merchandising partner type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of PTNR.	AG, AP, BK, BR, CN, CO, E, EC, ES, EV, FA, FF, IA, IM, S1, S2, S3
entityCode	No	String	This field contains the entity's Merchandising internal code. For example, ID of store or warehouse	
name	No	String	This field contains the name responsible for transportation. This field is used only when there is no entity code defined for transportation. It's used by SIM shipments because there is no transportation entity.	
carrierVehicleRn	No	String	This field contains the vehicle registration number responsible for transportation.	
carrierVehicleState	No	String	This field contains the state of the country where vehicle, responsible for transportation, is registered.	
carrierDriverName	No	String	This field contains the driver name responsible for transportation.	

Parameter Name	Required	Data Type	Description	Valid values
carrierDriverLn	No	String	This field contains the information about driver license ID responsible for transportation.	

RestFdgDtl:

The *fiscalDocument/details* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
Item	Yes	String	This field contains the unique code ID that identifies the item.	
totalAmt	Yes	BigDecimal	This field contains the total value of this item in the fiscal document.	
unitAmt	Yes	BigDecimal	This field contains the item unit cost of this item in the document.	
uom	Yes	String	This field contains the unit of measure used in item in the fiscal document.	
qty	Yes	BigDecimal	This field contains the quantity of item in this fiscal document.	
totalTaxAmt	No	BigDecimal	This field contains the total value of item in this fiscal document.	
totalAmtInclTax	No	BigDecimal	This field contains the total value including all taxes value of item in this fiscal document.	
taxes	No	List<RestFdgTax>	Collection of item taxes.	
references	No	List<RestFdgRef>	Collection of item references.	
allowanceCharges	No	List<RestFdgNonMerch>	Collection of item allowance/changes.	

RestFdgTax

The *fiscalDocument/details/taxes* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
taxType	Yes	String	This field contains the tax type at item level.	U (for tax per value) or P (for tax per percentage).
taxCode	Yes	String	This field contains the tax identification code that should be applied at item level.	
taxAmt	Yes	BigDecimal	This field contains the tax value at item level, when tax type is set as per value.	
taxCodeDesc	Yes	String	This field contains the tax code description at item level. This field can be used to indicate extra code or detailed information about tax code.	

RestFdgRef:

The *fiscalDocument/details/references* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
refType	Yes	String	This field contains the type of the reference at item level. It is used when there is any reference that it is necessary to add in this fiscal document, as shipment number, contract number, order, ID, and so on. Valid values: CODE_DETAIL – FDRT.	FDOC, ORDER, SHIPN, FULLF, ALLOC, DISTRO, BOLN, TSFNO, RTVNO, CONO, RTVAUT,CT RC
refCode	Yes	String	This field contains the reference code/value of reference type at item level.	

RestFdgNonMerch:

The *fiscalDocument/details/allowanceCharges* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
chargeInd	Yes	String	This field contains the charge indicator at item allowance charges level.	Y (represents a charge). N (represents an allowance).
nonMerchCode	Yes	String	This field contains the non-merchandise code that identify non-merchandise cost at item allowance changes level. This value should be existing in table NON_MERCH_CODE_HEAD.	
nonMerchAmt	Yes	BigDecimal	This field contains the non-merchandise value at item allowance changes level.	
taxes	No	List<RestFdgTax>	Collection of non-merchandise taxes.	

RestFdgTax:

The *fiscalDocument/details/allowanceCharges/taxes* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
taxType	Yes	String	This field contains the tax type at item allowance charge taxes level.	U (for tax per value) or P (for tax per percentage).
taxCode	Yes	String	This field contains the tax identification code that should be applied at item allowance charges taxes level.	
taxAmt	Yes	BigDecimal	This field contains the tax value at item allowance charges taxes level, when tax is set up as per value.	
taxCodeDesc	Yes	String	This field contains the tax code description at item allowance charges taxes level. This field can be used to indicate extra code or detailed information about tax code.	
comments	No	List<RestFdgText>	Collection of non-merchandise taxes comments.	

RestFdgText:

The *fiscalDocument/details/allowanceCharges/taxes/comments* and *fiscalDocument/comments* node have the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
text	Yes	String	This field contains additional comments at item level. It is used to provide any additional information or legal message important to fiscal document.	
textGroupId	No	String	Text group ID. It is used to group related comments when there is more than one or when the size of a message is bigger than the size of the text field and it is necessary to generate more than one line.	
textGroupSeq	No	String	Sequence of the text for a given group ID. It is used in case comments information has more than one line.	

RestFdgRef

The *fiscalDocument/references* node has the following parameters:

Parameter Name	Required	Data Type	Description	Valid values
refType	Yes	String	This field contains the type of the reference at item level. It is used when there is any reference that it is necessary to add in this fiscal document, as shipment number, contract number, order, ID, and so on. Valid values: CODE_DETAIL – FDRT.	FDOC, ORDER, SHIPN, FULLF, ALLOC, DISTRO, BOLN, TSFNO, RTVNO, CONO, RTVAUT, CTRC
refCode	Yes	String	This field contains the reference code/value of reference type at item level.	

JSON Input Structure:

```
{
  "fiscalDocument": {
    "requestSystem": "SUPPLIER_PORTAL",
    "docType": "WBL",
    "extReqId": "1234",
    "transactionType": "TSF",
    "issueDate": "2020-12-16T08:19:42-05:00",
    "currencyCode": "USD",
    "deduceTax": "Y",
    "entities": [
```

```

    {
      "entityType": "SHIPP",
      "entityInternalType": "WH",
      "entityCode": "7001"
    },
    {
      "entityType": "DEST",
      "entityInternalType": "ST",
      "entityCode": "777701"
    },
    {
      "entityType": "CR",
      "entityInternalType": "PTNR",
      "entityInternalPtnrType": "FF",
      "entityCode": "11",
      "carrierVehicleRn": "IF5597",
      "carrierVehicleState": "SP",
      "carrierDriverName": "Driver Name",
      "carrierDriverLn": "asasdd984955456"
    }
  ],
  "details": [
    {
      "item": "54987897",
      "totalAmt": 10,
      "unitAmt": null,
      "uom": "EA",
      "qty": 5,
      "totalTaxAmt": 7,
      "totalAmtInclTax": 57,
      "taxes": [
        {
          "taxType": "P",
          "taxCode": "GST",
          "taxAmt": 5,
          "taxCodeDesc": null
        },
        {
          "taxType": "P",
          "taxCode": "PST",
          "taxAmt": 2,
          "unitTaxAmt": null,
          "taxCodeDesc": null
        }
      ],
      "references": [
        {
          "refType": "ORDER",
          "refCode": "897889"
        }
      ],
      "allowanceCharges": [
        {
          "chargeInd": "Y",
          "nonMerchCode": "FRHT",
          "nonMerchAmt": 10,
          "taxes": [
            {

```

```

        "taxType": "P",
        "taxCode": "GST",
        "taxAmt": 1.8,
        "taxCodeDesc": null,
        "comments": []
      },
      {
        "taxType": "P",
        "taxCode": "PST",
        "taxAmt": 1,
        "taxCodeDesc": null,
        "comments": []
      }
    ]
  },
  "comments": [{
    "text": "Open box Item"
  }
]
}
],
"comments": [{
  "text": "Additional comments to the document. If it's too l'",
  "textGroupId": "1",
  "textGroupSeq": "1"
},
{
  "text": "ong, it can end in other line",
  "textGroupId": "1",
  "textGroupSeq": "2"
}
],
"references": [{
  "refType": "DOC",
  "refCode": "9848945651"
}]
}
}"

```

Output

Parameter Name	Data Type	Description
requestId	String	This field contains the unique identifier of the document request.
requestSystem	String	This column contains the unique identifier of the requester.
requestAction	String	This column contains the current action for the request.
docId	BigDecimal	This column contains the unique document ID.
docNo	String	This column contains the unique document number.

Parameter Name	Data Type	Description
docType	String	This column contains the type of the document.
status	String	This column contains the status of the document.
extReqId	String	This field contains the external request ID. This is the internal number from system that is requesting a fiscal document. Example: asn_nbr ID from shipments.
docPrintUrl	String	This column contains the url to access the fiscal document issued by government. It can be a url from third-party repository responsible to communicate with government or other url.
additionalproperties	List<RestFdgExt>	Collection of additional properties.
errors	List<RestFdgError>	Collection of errors.

RestFdgExt

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.

RestFdgError

Parameter Name	Data Type	Description
errorCode	String	This field contains the error code.
errorDesc	String	This field contains the error description.

JSON Structure:

```
"{
  "requestId": 62,
  "requestSystem": "SIOCS",
  "requestAction": "A",
  "docId": 62,
  "docNo": null,
  "docType": "WBL",
  "status": "N",
  "extReqId": "10003",
  "docPrintUrl": null,
```

```

    "additionalproperties": [],
    "errors": []
  }"

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
FDG_ATTRIB_MAP	Yes	No	No	No
FDG_ATTRIB_MAP_NM	Yes	No	No	No
SVC_FDG_DTL	Yes	Yes	Yes	No
SVC_FDG_DTL_PACK	Yes	Yes	Yes	No
SVC_FDG_ERROR	Yes	Yes	No	No
SVC_FDG_ETT	Yes	Yes	Yes	No
SVC_FDG_EXT	Yes	Yes	Yes	No
SVC_FDG_HDR	Yes	Yes	Yes	No
SVC_FDG_NON_MERCH	Yes	Yes	No	No
SVC_FDG_REF	Yes	Yes	No	No
SVC_FDG_TAX	Yes	Yes	No	No
SVC_FDG_TEXT	Yes	Yes	No	No
CODE_DETAIL	Yes	No	No	No
V_CODE_DETAIL_TL	Yes	No	No	No
STORE	Yes	No	No	No
PARTNER	Yes	No	No	No
WH	Yes	No	No	No
LANG	Yes	No	No	No
FDG_SYSTEM_OPTIONS	Yes	No	No	No
CFA_ATTRIB	Yes	No	No	No
CFA_ATTRIB_MAP	Yes	No	No	No
CFA_ATTRIB_MAP_LABES	Yes	No	No	No
RTV_DETAIL	Yes	No	No	No
ORDCUST	Yes	No	No	No
ORDCUST_DETAIL	Yes	No	No	No
TSFHEAD	Yes	No	No	No
TSFDETAIL	Yes	No	No	No
SHIPSKU	Yes	No	No	No
ADDR	Yes	No	No	No
PACKITEM_BREAKOUT	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
V_ITEM_MASTER	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No

Fiscal Document Status Detail Request Service

Functional Area

Finance

Business Overview

In both scenarios for shipment and Inbound, it will be the requestor's responsibility to pull the approved documents from Fiscal Document Generation in order to conclude their transactions. This web service will be used to provide the status of fiscal documents requested through the [Fiscal Pre Document Request Service](#) and [Fiscal Document Request Service](#). In case the document is approved, approval-related data will also be provided, such as the formal document number and print-version URL path.

Service Type

Get

ReST URL

```
{{baseUrl}}/services/private/fdg/documentStatusDetail?  
requestId=<string>&requestSystem=<string>&entityType=<string>&entityInternalTy  
pe=<string>&entityCode=<string>&country=<string>
```

Input Parameters

Parameter Name	Required	Data Type	Description	Valid values
requestId	Yes	String	This column contains the unique identifier of the document request.	
requestSystem	No	String	This column contains the unique identifier of the requester.	
entityType	No	String	This field contains the entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDED.	SHIPP, DEST, VENDOR, CR, LOC
entityInternalType	No	String	This field contains the Merchandising entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDET.	WH, ST, SUPP, PTNR, CUST
entityCode	No	String	This column contains the entity's Merchandising internal code.	
country	No	String	This column contains the country where is the location address.	

Output

Parameter Name	Data Type	Description
requestId	String	Internal fiscal document REQUEST ID generated by FDG. This ID can be used to identify the fiscal document(s) being processed.
requestSystem	String	Name of the system that performed the request for a fiscal document.
requestAction	String	Action code to be passed along with the request for a fiscal document process.
docNo	String	Official fiscal document number. This number is given to approved documents.
docType	String	Valid values: INV (Invoice), WBL (eWaybill), NFe (Nota Fiscal Eletronica).
status	String	Internal FDG status of the fiscal document.
extReqId	String	External system's request ID used to link FDG internal ID with the requestor system's transaction ID.
docPrintUrl	String	URL of the fiscal document report.
additionalproperties	List<RestFdgExt>	Collection of additional properties.
errors	List<RestFdgError>	Collection of errors.

RestFdgExt

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.

RestFdgError

Parameter Name	Data Type	Description
errorCode	String	This field contains the error code.
errorDesc	String	This field contains the error description.

JSON Structure:

```
"{"
  "requestId": 50039,
  "requestSystem": "SUPPLIER_PORTAL",
  "requestAction": "A",
  "docId": 50038,
  "docNo": null,
  "docType": "WBL",
  "status": "S",
  "extReqId": "10000600",
  "docPrintUrl": null,
  "additionalproperties": [],
  "errors": []
}"
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_FDG_ERROR	Yes	No	No	No
SVC_FDG_EXT	Yes	No	No	No
SVC_FDG_HDR	Yes	No	No	No
FDG_ERROR	Yes	No	No	No
FDG_EXT	Yes	No	No	No
FDG_HDR	Yes	No	No	No

Next Fiscal Document Request Service

Functional Area

Finance

Business Overview

This is the service to be called by the 3rd-party system in charge of communicating fiscal document request to local Governments in order to fetch documents ready to be approved.

Service Type

Get

ReST URL

```
{{baseUrl}}/services/private/fdg/nextDocumentDetail?
docId=<string>&docType=<string>&entityType=<string>&internalType=<string>&inte
rnalCode=<string>&integrationSystemName=<string>&integrationSystemId=<string>
```

Input Parameters

Parameter Name	Required	Data Type	Description	Valid values
docId	No	String	Internal fiscal document ID generated by FDG. This ID is unique to each fiscal document created.	
docType	No	String	This field contains the type of the document. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDTY.	INV, WBL
entityType	No	String	This field contains the entity type. Valid values are codes from CODE_DETAIL with a CODE_TYPE of FDED.	SHIPP, DEST, VENDOR, CR, LOC
internalType	No	String	Filter criteria to fetch documents based on specific entity types in Merchandising irrespective of their role in the fiscal documents.	WH, ST, SUPP, PTRN, CUST
internalCode	No	String	Filter criteria to fetch documents based on a specific entity code. This field contains the entity's Merchandising internal code and must sent in conjunction with entityInternatType tag. For example, ID of store or warehouse	
integrationSystemName	Yes	String	Name of the integration system that will process this document.	
integrationSystemId	Yes	String	Unique identifier to be used by the integration system.	

Output

Parameter Name	Data Type	Description
requestId	String	Internal fiscal document REQUEST ID generated by FDG. This ID can be used to identify the fiscal document(s) being processed.
requestSystem	String	Name of the system that performed the request for a fiscal document.

Parameter Name	Data Type	Description
requestAction	String	Action code to be passed along with the request for a fiscal document process.
docId	BigDecimal	Internal fiscal document ID generated by FDG. This ID is unique to each fiscal document created.
docNo	String	Official fiscal document number. This number is given to approved documents.
status	String	Internal FDG status of the fiscal document.
docType	String	Valid values: INV (Invoice), WBL (eWaybill), NFe (Nota Fiscal Eletronica).
docTypeCode	String	Extended description of a document type.
extReqId	String	External system's request ID used to link FDG internal ID with the requestor system's transaction ID.
transactionType	String	Identification of the transaction associated with the fiscal document.
issueDate	OffsetDateTime	Issue date of the fiscal document. Date defined at the time the document is submitted for approval.
dueDate	date	Payment date of fiscal document or invoice.
docLang	String	Language of fiscal document. It is based on language defined in source location.
printerId	String	Printer ID that is designated to print fiscal document.
docPrintUrl	String	URL of the fiscal document report.
currencyCode	String	Currency code for the fiscal document. It is based on currency code defined in source location.
totalTaxAmt	BigDecimal	Total TAX amount of the fiscal document. This field is the sum of totalTaxAmt of all items of the document. Rounding applied based on system option.
totalNonMerchTaxAmt	BigDecimal	Total TAX amount of allowances/charges of the fiscal document. Rounding applied based on system option.
totalAmtInclTax	BigDecimal	Total value of fiscal document including taxes. This field is the sum of totalAmtInclTax of all items of the document. Rounding applied based on system option.

Parameter Name	Data Type	Description
totalAmtExclTax	BigDecimal	Total value of fiscal document excluding taxes. This field has the totalAmtInclTax minus totalTaxAmt. Rounding applied based on system option.
totalAllowance	BigDecimal	Total value of all allowances in fiscal document. This field is the sum of totalAllowance of all items of the document. Rounding applied based on system option.
totalCharge	BigDecimal	Total value of all charges in fiscal document. This field is the sum of totalCharge of all items of the document. Rounding applied based on system option.
totalRounding	BigDecimal	The rounding amount (positive or negative) added to produce the line extension totalAmt. This field is the sum of totalRounding of all items of the document.
terms	String	Payment terms used in this transaction
deduceTax	String	Flag that indicates if taxes were calculated by the system or informed at the fiscal document request.
createBy	String	User ID that created the record.
createDatetime	date	Record creation date.
lastUpdateBy	String	User ID that did last update of fiscal document.
lastUpdateDatetime	date	Date of last update of fiscal document.
entities	List<RestFdgEtt>	Collection of entities.
details	List<RestFdgDtl>	Collection of item details

RestFdgEtt

Parameter Name	Data Type	Description
entityType	String	Entity type.
entityInternalPtrnType	String	Merchandising partner type.
entityInternalType	String	Merchandising entity type.
entityCode	String	Internal Merchandising code of the entity. For example, ID of store or warehouse
name	String	Entity name.
legalName	String	Entity's legal name.
carrierVehicleRn	String	Carrier vehicle's registration number.
carrierVehicleState	String	Carrier vehicle's state.

Parameter Name	Data Type	Description
carrierDriverName	String	Carrier driver's name.
carrierDriverLn	String	Carrier driver's license number.
address	RestFdgEttAddr	Collection of address details.
identification	List<RestFdgExt>	Collection of Entity Identification Additional Properties. Fiscal Attributes will be send in this group.

RestFdgEttAddr

Parameter Name	Data Type	Description
street	String	Street name of the entity's address.
additionalStreetName	String	Extension of street name of the entity's address.
complement	String	Additional information of entity's address, such as neighborhood, county, etc.
city	String	City name of entity's address.
jurisdictionCode	String	Jurisdiction code of entity's address. This can be legal code associated to city, county or any other level.
postalCode	String	Zip or postal code of entity's address.
country	String	Country name of entity's address.
telephone	String	Telephone number of the person who is the contact point.
email	String	Email address of the person who is the contact point.
state	String	State name of entity's address.
contactName	String	Contact name.
additionalproperties	List<RestFdgExt>	Collection of address additional properties.

RestFdgExt

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.
description	String	Description of additional property data.

RestFdgExt (identification)

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.
description	String	Description of additional property data.
references	List<RestFdgRef>	Collection of references.
totaltaxes	List<RestFdgTax>	Collection of taxes.
errors	List<RestFdgError>	Collection of errors.
allowanceCharges	List<RestFdgNonMerc h>	Collection of allowance charges.
additionalproperties	List<RestFdgExt>	Collection of additional properties.
comments	List<RestFdgText>	Collection of comments.

RestFdgDtl

Parameter Name	Data Type	Description
lineNo	long	Sequential number of the item in fiscal document. This number is a sequence starts with number one. Any gap is not allowed.
Item	String	Unique code ID that identifies the item.
itemDesc	String	Long description of the item.
itemShortDesc	String	Short description of the item.
totalAmt	BigDecimal	Total amount for the item inclusive of allowances and charges. Calculation formula is ((unitAmt*qty) - totalAllowance + totalCharge). Rounding applied based on system option.
unitAmt	BigDecimal	Item unit amount. No rounding applied.
uom	String	Unit of measure used in item in the fiscal document.
qty	BigDecimal	Quantity of item in the fiscal document.
totalTaxAmt	BigDecimal	Total amount of taxes applied to the item. This field is sum of all taxAmt at tax level for this item. Rounding applied based on system option.

Parameter Name	Data Type	Description
totalAmtInclTax	BigDecimal	Total amount for item inclusive of taxes. Calculation formula is totalAmt + totalTaxAmt. Rounding applied based on system option.
totalRounding	BigDecimal	The rounding amount (positive or negative) added to produce the line extension totalAmt. Calculation formula ((unitAmt*qty) - totalAllowance + totalCharge) - totalAmt.
totalAllowance	BigDecimal	Total amount of all allowances at item level. This field is sum of all allowances at allowanceCharges level for this item. Rounding applied based on system option.
totalCharge	BigDecimal	Total amount of all charges at item level. This field is sum of all charges at allowanceCharges level for this item. Rounding applied based on system option.
packInd	String	Pack indicator. Valid values: Y (this item is a pack) or N (this item is not a pack).
gtin	String	GTIN number (Global Trade Item Number) of item.
gtinType	String	GTIN Type. Valid values: GTIN-8, GTIN-12, GTIN-13, GTIN-14
vpn	String	Vendor Product Number (VPN) associated with the item.
taxes	List<RestFdgTax>	Collection of item taxes.
references	List<RestFdgRef>	Collection of item references
allowanceCharges	List<RestFdgNonMerch>	Collection of item allowance charges.
packdetails	List<RestFdgDtlPack>	Collection of item pack details.
additionalproperties	List<RestFdgExt>	Collection of item additional properties.
comments	List<RestFdgText>	Collection of item comments.

RestFdgTax

Parameter Name	Data Type	Description
taxType	String	Tax type.
taxCode	String	Tax identification code.
taxRate	BigDecimal	Tax rate when tax type is set as percentage.
taxAmt	BigDecimal	Total tax amount for the tax code. Rounding applied based on system option.

Parameter Name	Data Type	Description
unitTaxAmt	BigDecimal	Tax amount per unit for the tax code.
taxBasis	BigDecimal	Tax basis amount used to calculate tax value based on taxRate field. Rounding applied based on system option.
taxCodeDesc	String	Tax code description. This field can be used to indicate extra code or detailed information about tax code.
additionalproperties	List<RestFdgExt>	Collection of taxes addition properties.
comments	List<RestFdgText>	Collection of taxes comments.

RestFdgText

Parameter Name	Data Type	Description
text	String	Additional information and/or comments. It is used to provide any additional information or legal message important to the fiscal document.
textGroupId	String	Text group ID. It is used to group related comments when there is more than one or when the size of a message is bigger than the size of the text field and it is necessary to generate more than one line.
textGroupSeq	String	Sequence of the text for a given group ID. It is used in case comments information has more than one line.

RestFdgRef

Parameter Name	Data Type	Description
refType	String	Type of the reference information. It is used when there is any reference that it is necessary to add in the fiscal document, as shipment number, contract number, order, ID, and so on. Valid values are pre-defined in the Merchandising system.
refCode	String	Reference code/value of reference type.

RestFdgNonMerch

Parameter Name	Data Type	Description
chargeInd	String	Charge indicator.

Parameter Name	Data Type	Description
nonMerchCode	String	Non-merchandise code that identifies a non-merchandise cost used as a charge or allowance.
nonMerchCodeDesc	String	Non-merchandise code description. It is used to provide more detailed about non-merchandise code, for example: allowance details, types of non-merchandise code such as Freight, Insurance, and so on.
nonMerchAmt	BigDecimal	Non-merchandise amount.
nonMerchType	String	Non-merchandise type exclusive for Charges.
taxes	List<RestFdgTax>	Collection of taxes for non-merchandise.
additionalproperties	List<RestFdgExt>	Collection of additional properties for non-merchandise.
comments	List<RestFdgText>	Collection of comments for non-merchandises.

packdetails

Parameter Name	Data Type	Description
item	String	Unique alphanumeric ID of a pack component that is part of item pack.
itemQty	BigDecimal	Quantity of the component item that is part of the pack.
itemDesc	String	Description of the component item component.
uom	String	Unit of measure of the component item.

totaltaxes

Parameter Name	Data Type	Description
taxType	String	Not applicable for totalTaxes.
taxCode	String	Tax identification code.
taxRate	BigDecimal	Not applicable for totalTaxes.
taxAmt	BigDecimal	Sum of taxAmt for all items for the tax code. Rounding applied based on system option
unitTaxAmt	BigDecimal	Not applicable for totalTaxes.
taxBasis	BigDecimal	Sum of taxBasis for all items for the tax code. Rounding applied based on system option.
taxCodeDesc	String	Tax code description. This field can be used to indicate extra code or detailed information about tax code.

JSON Structure:

```
{
  "fiscalDocument": {
    "docId": 17200,
    "requestId": 17207,
    "requestSystem": "OMNI",
    "requestAction": "A",
    "status": "R",
    "docNo": "17200",
    "docType": "WBL",
    "docTypeCode": null,
    "extReqId": "10065007",
    "transactionType": "SALES",
    "issueDate": 1617196782000,
    "dueDate": null,
    "printerId": null,
    "docPrintUrl": null,
    "currencyCode": "USD",
    "totalTaxAmt": null,
    "totalNonMerchTaxAmt": null,
    "totalAmtInclTax": null,
    "totalAmtExclTax": null,
    "totalAllowance": null,
    "totalCharge": null,
    "totalRounding": null,
    "terms": null,
    "docLang": "EN",
    "deduceTax": null,
    "createBy": "RMS_ADMIN",
    "createDatetime": "2021-06-16",
    "lastUpdateBy": "RFM01",
    "lastUpdateDatetime": "2021-06-16",
    "entities": [
      {
        "entityType": "SHIPP",
        "entityInternalType": "ST",
        "entityInternalPtnrType": null,
        "entityCode": "990011",
        "name": "QABR-Store Test Washington",
        "legalName": "QABR-Store Test Washington",
        "carrierVehicleRn": null,
        "carrierVehicleState": null,
        "carrierDriverName": null,
        "carrierDriverLn": null,
        "address": {
          "street": "R da Juta",
          "additionalStreetName": "1512",
          "complement": null,
          "city": "Washijngton",
          "jurisdictionCode": null,
          "postalCode": "03308070",
          "country": "US",
          "telephone": "2360873",
          "email": "washington.dias@oracle.com",
```

```

        "state": "DC",
        "contactName": "Washington Dias",
        "additionalproperties": []
    },
    "identification": []
},
{
    "entityType": "DEST",
    "entityInternalType": "CUST",
    "entityInternalPtnrType": null,
    "entityCode": "75",
    "name": "wdias losnisk",
    "legalName": "JADLOG",
    "carrierVehicleRn": null,
    "carrierVehicleState": null,
    "carrierDriverName": null,
    "carrierDriverLn": null,
    "address": {
        "street": "end10",
        "additionalStreetName": "end20",
        "complement": "end30",
        "city": "Washington",
        "jurisdictionCode": "3100104",
        "postalCode": "08573160",
        "country": "US",
        "telephone": null,
        "email": null,
        "state": "DC",
        "contactName": "wdias losnisk",
        "additionalproperties": []
    },
    "identification": []
},
{
    "entityType": "CR",
    "entityInternalType": "PTNR",
    "entityInternalPtnrType": "FF",
    "entityCode": "11",
    "name": "Carrier",
    "legalName": "Carrier",
    "carrierVehicleRn": "CJU0939",
    "carrierVehicleState": "São Paulo",
    "carrierDriverName": "Djalma da Carreta",
    "carrierDriverLn": "77799573959957",
    "address": {
        "street": "15333 JFK Blvd",
        "additionalStreetName": null,
        "complement": null,
        "city": "Houston",
        "jurisdictionCode": null,
        "postalCode": "77032",
        "country": "US",
        "telephone": null,
        "email": "juliano.costa@oracle.com",
        "state": "TX",
        "contactName": "Juliano Costa",
    }
}

```

```

      "additionalproperties": [
        {
          "key": "NIP",
          "value": "NIP09",
          "type": "VARCHAR2",
          "description": "NIP"
        },
        {
          "key": "POSTALCODE",
          "value": "POST09",
          "type": "VARCHAR2",
          "description": "Postal Code"
        }
      ]
    },
    "identification": [
      {
        "key": "TAX_ID",
        "value": "332424324",
        "type": "VARCHAR2",
        "description": "TAX ID"
      }
    ]
  },
  ],
  "details": [
    {
      "lineNo": 1,
      "item": "100900006",
      "itemDesc": "item regular scenario 3",
      "itemShortDesc": "regular item regular scenario 3",
      "totalAmt": null,
      "unitAmt": null,
      "uom": "EA",
      "qty": 5,
      "totalTaxAmt": null,
      "totalAmtInclTax": null,
      "packInd": "N",
      "gtin": null,
      "gtinType": "ITEM",
      "vpn": null,
      "totalRounding": null,
      "totalAllowance": null,
      "totalCharge": null,
      "taxes": [],
      "references": [],
      "allowanceCharges": [],
      "packdetails": [],
      "additionalproperties": [],
      "comments": []
    },
    {
      "lineNo": 2,
      "item": "100900065",
      "itemDesc": "itemm simple pack scenario 3",
      "itemShortDesc": "itemm simple pack scenario 3",

```


[illegible]

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_FDG_DTL	Yes	No	No	No
SVC_FDG_DTL_PACK	Yes	No	No	No
SVC_FDG_ERROR	Yes	No	No	No
SVC_FDG_ETT	Yes	No	No	No
SVC_FDG_EXT	Yes	No	No	No
SVC_FDG_HDR	Yes	No	Yes	No
SVC_FDG_NON_MERCH	Yes	No	No	No
SVC_FDG_REF	Yes	No	No	No
SVC_FDG_TAX	Yes	No	No	No
SVC_FDG_TEXT	Yes	No	No	No
FDG_DTL	Yes	No	No	No
FDG_DTL_PACK	Yes	No	No	No
FDG_ERROR	Yes	No	No	No
FDG_ETT	Yes	No	No	No
FDG_EXT	Yes	No	No	No
FDG_HDR	Yes	No	No	No
FDG_NON_MERCH	Yes	No	No	No
FDG_REF	Yes	No	No	No
FDG_TAX	Yes	No	No	No
FDG_TEXT	Yes	No	No	No

Fiscal Document Process Status Request Service

Functional Area

Finance

Business Overview

This web service will be used by 3rd-party systems in charge of communicating fiscal document requests to local Governments, to provide to Fiscal Document Generation the approval status and associated data such as the formal document number and print version URL path.

Service Type

Post

ReST URL

{{baseUrl}}/services/private/fdg/processDocStatus

Input

Parameter Name	Required	Data Type	Description	Valid values
integrationSystemName	Yes	String	Name of the integration system that will process this document.	
integrationSystemId	Yes	String	Unique identifier to be used by the integration system.	
requestAction	No	String	Action code returned for a given document in the nextDocumentDetail response.	A (approve) or C (cancel).
docId	Yes	String	Document ID returned for a given document in nextDocumentDetail response.	
docNo	Yes	String	Document number returned for a given document in nextDocumentDetail response.	
operationStatus	Yes	String	Status of the processing executed by a third party for the given document. Received means the document payload was properly consumed by the third party system and it is ready to proceed with its processing. Success means that the document was successfully approved. Fail means that the document approval failed; in this case it is expected that the error or errors will be provided.	R (Received) S (Success) or F (Fail)
extDocNo	No	String	External document number when applicable. This field should be populated with the approved fiscal document number when it is generated by Government.	
docPrintUrl	No	String	URL to access or download the fiscal document printable version (PDF).	
additionalproperties	No	List<RestFdgExt>	Collection of additional properties.	
errors	No	List<RestFdgError>	Collection of errors.	

RestFdgExt:

Parameter Name	Required	Data Type	Description	Valid values
key	No	String	Key value of the additional property data.	
value	No	String	Value of the additional property data.	
type	No	String	Data type of the additional property.	VARCHAR2, NUMBER or DATE
description	No	String	Description of additional property data.	

RestFdgError:

Parameter Name	Required	Data Type	Description	Valid values
errorCode	Yes	String	Error code defined by 3rd party.	
errorDesc	Yes	String	Error description defined by 3rd party. This description must have enough detail so the problem can be identified.	

JSON Input Structure:

```
"{
  "integrationSystemName": "3RDPARNER",
  "integrationSystemId": 777,
  "requestAction": "A",
  "docId": 669854788524448800,
  "docNo": 1235489884,
  "operationStatus": "S",
  "extDocNo": 999999999999,
  "docPrintUrl": "www.doc.com/print?code=123",
  "additionalproperties": [
    {
      "key": "EXTID1",
      "value": "ExtensionValue",
      "type": "VARCHAR",
      "description": "Description"
    }
  ],
  "errors": [
    {
      "errorCode": "ERR87896",
      "errorDesc": "Unknow Error"
    }
  ]
}"
```

Output

Parameter Name	Data Type	Description
status	String	Request process status. Can be either successfully or with error.
errors	List<RestFdgError>	Collection of errors.

RestFdgError

Parameter Name	Data Type	Description
errorCode	String	This field contains the error code.
errorDesc	String	This field contains the error description.

JSON Structure:

```
"{
  "status": "E",
  "errors": [
    {
      "errorCode": "FDG_DOC_NOT_FOUND",
      "errorDesc": "Fiscal document with id 669854788524448800 not found."
    }
  ]
}"
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_FDG_DTL	Yes	No	No	Yes
SVC_FDG_DTL_PACK	Yes	No	No	Yes
SVC_FDG_ERROR	Yes	Yes	No	Yes
SVC_FDG_ETT	Yes	No	No	Yes
SVC_FDG_EXT	Yes	Yes	No	Yes
SVC_FDG_HDR	Yes	No	Yes	Yes
SVC_FDG_NON_MERCH	Yes	No	No	Yes
SVC_FDG_REF	Yes	No	No	Yes
SVC_FDG_TAX	Yes	No	No	Yes
SVC_FDG_TEXT	Yes	No	No	Yes
FDG_DTL	No	Yes	No	No
FDG_DTL_PACK	No	Yes	No	No
FDG_ERROR	No	Yes	No	No
FDG_ETT	No	Yes	No	No
FDG_EXT	No	Yes	No	No
FDG_HDR	Yes	Yes	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
FDG_NON_MERCH	No	Yes	No	No
FDG_REF	No	Yes	No	No
FDG_TAX	No	Yes	No	No
FDG_TEXT	No	Yes	No	No
FDG_SYSTEM_OPTIONS	No	No	No	No

Fiscal Document Cancel Request Service

Functional Area

Finance

Business Overview

In both scenarios for shipment and Inbound, it will be the requestor's option to cancel a fiscal document request. This process runs by calling the Document Cancel Request Service.

Requestor systems can request cancelation for a specific document. This request is consumed by Fiscal Document Generation and, depending on the current status of the document, different actions will take place including simply changing the status of the document to Canceled, or even the integration of the cancelation request to the 3rd-party system that communicates this request to local Governments. In this scenario, Fiscal Document Generation will expect the 3rd party to confirm the cancelation.

Service Type

Post

ReST URL

```
{{baseUrl}}/services/private/fdg/processDocCancel
```

Input

Parameter Name	Required	Data Type	Description	Valid values
docId	Yes	String	This column contains the unique document ID.	
requestAction	Yes	String	This field contains the current action for the request. This action is to third-party know the action to perform with this request.	A (Approve) or C (Cancel).
additionalproperties	No	List<RestFdgExt>	Collection of additional properties.	
comments	No	List<RestFdgText>	Collection of comments.	

RestFdgExt:

Parameter Name	Required	Data Type	Description	Valid values
key	No	String	Key value of the additional property data.	
value	No	String	Value of the additional property data.	
type	No	String	Data type of the additional property.	VARCHAR2, NUMBER or DATE
description	No	String	Description of additional property data.	

RestFdgText:

Parameter Name	Required	Data Type	Description	Valid values
text	Yes	String	This field contains additional comments at item level. It is used to provide any additional information or legal message important to fiscal document.	
textGroupId	No	String	Text group ID. It is used to group related comments when there is more than one or when the size of a message is bigger than the size of the text field and it is necessary to generate more than one line.	
textGroupSeq	No	String	Sequence of the text for a given group ID. It is used in case comments information has more than one line.	

JSON Input Structure:

```
"{
  "docId": "12345",
  "requestAction": "C",
  "additionalproperties": [
    {
      "key": "EXTID1",
      "value": "ExtensionValue",
      "type": "VARCHAR2"
    }
  ],
  "comments":
    [{
      "text": "additional text"
    }]
}"
```

Output

Parameter Name	Data Type	Description
processStatus	String	This column contains the status of the document
errors	List<RestFdgError>	Collection of errors.

RestFdgError

Parameter Name	Data Type	Description
errorCode	String	This field contains the error code.
errorDesc	String	This field contains the error description.

JSON Structure:

```
"{
  "status": "E",
  "errors": [
    {
      "errorCode": "FDG_DOC_NOT_FOUND",
      "errorDesc": "Fiscal document with id 12345 not found."
    }
  ]
}"
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_FDG_HDR	No	No	Yes	No
FDG_HDR	Yes	No	Yes	No
FDG_EXT	No	Yes	No	No
FDG_TEXT	No	Yes	No	No

Fiscal Document Detail Request Service

Functional Area

Finance

Business Overview

This web service is called by 3rd-party systems responsible for communicating fiscal document request to local Governments when they need to fetch data for a particular document. The response to this service will have the full payload of a fiscal document.

Service Type

Get

ReST URL

```
{{baseUrl}}/services/private/fdg/documentDetail?  
docId=<string>&entityType=<string>&entityInternalType=<string>&entityCode=<string>
```

Input Parameters

Parameter Name	Required	Data Type	Description	Valid values
docId	No	String	Internal fiscal document ID generated by FDG. This ID is unique to each fiscal document created.	
entityType	No	String	Filter criteria to fetch documents for specific entity types in a fiscal document.	SHIPP, DEST, CR, VENDOR, INV, LOC
entityInternalType	No	String	Filter criteria to fetch documents based on specific entity types in Merchandising irrespective of their role in the fiscal documents.	WH, ST, SUPP, PTNR, CUST
entityCode	No	String	Filter criteria to fetch documents based on a specific entity code. This field contains the entity's Merchandising internal code and must sent in conjunction with entityInternalType tag. For example, ID of store or warehouse.	

Output

Parameter Name	Data Type	Description
requestId	String	Internal fiscal document REQUEST ID generated by FDG. This ID can be used to identify the fiscal document(s) being processed.
requestSystem	String	Name of the system that performed the request for a fiscal document.

Parameter Name	Data Type	Description
requestAction	String	Action code to be passed along with the request for a fiscal document process.
docId	BigDecimal	Internal fiscal document ID generated by FDG. This ID is unique to each fiscal document created.
docNo	String	Official fiscal document number. This number is given to approved documents.
status	String	Internal FDG status of the fiscal document.
docType	String	Valid values: INV (Invoice), WBL (eWaybill), NFe (Nota Fiscal Eletronica).
docTypeCode	String	Extended description of a document type.
extReqId	String	External system's request ID used to link FDG internal ID with the requestor system's transaction ID.
transactionType	String	Identification of the transaction associated with the fiscal document.
issueDate	OffsetDateTime	Issue date of the fiscal document. Date defined at the time the document is submitted for approval.
dueDate	date	Payment date of fiscal document or invoice.
docLang	String	Language of fiscal document. It is based on language defined in source location.
printerId	String	Printer ID that is designated to print fiscal document.
docPrintUrl	String	URL of the fiscal document report.
currencyCode	String	Currency code for the fiscal document. It is based on currency code defined in source location.
totalTaxAmt	BigDecimal	Total TAX amount of the fiscal document. This field is the sum of totalTaxAmt of all items of the document. Rounding applied based on system option.
totalNonMerchTaxAmt	BigDecimal	Total TAX amount of allowances/charges of the fiscal document. Rounding applied based on system option.
totalAmtInclTax	BigDecimal	Total value of fiscal document including taxes. This field is the sum of totalAmtInclTax of all items of the document. Rounding applied based on system option.

Parameter Name	Data Type	Description
totalAmtExclTax	BigDecimal	Total value of fiscal document excluding taxes. This field has the totalAmtInclTax minus totalTaxAmt. Rounding applied based on system option.
totalAllowance	BigDecimal	Total value of all allowances in fiscal document. This field is the sum of totalAllowance of all items of the document. Rounding applied based on system option.
totalCharge	BigDecimal	Total value of all charges in fiscal document. This field is the sum of totalCharge of all items of the document. Rounding applied based on system option.
totalRounding	BigDecimal	The rounding amount (positive or negative) added to produce the line extension totalAmt. This field is the sum of totalRounding of all items of the document.
terms	String	Payment terms used in this transaction
deduceTax	String	Flag that indicates if taxes were calculated by the system or informed at the fiscal document request.
createBy	String	User ID that created the record.
createDatetime	date	Record creation date.
lastUpdateBy	String	User ID that did last update of fiscal document.
lastUpdateDatetime	date	Date of last update of fiscal document.
entities	List<RestFdgEtt>	Collection of entities.
details	List<RestFdgDtl>	Collection of item details

RestFdgEtt

Parameter Name	Data Type	Description
entityType	String	Entity type.
entityInternalPtnrType	String	Merchandising partner type.
entityInternalType	String	Merchandising entity type.
entityCode	String	Internal Merchandising code of the entity. Ex. ID of store or warehouse
name	String	Entity name.
legalName	String	Entity's legal name.
carrierVehicleRn	String	Carrier vehicle's registration number.
carrierVehicleState	String	Carrier vehicle's state.

Parameter Name	Data Type	Description
carrierDriverName	String	Carrier driver's name.
carrierDriverLn	String	Carrier driver's license number.
address	RestFdgEttAddr	Collection of address details.
identification	List<RestFdgExt>	Collection of Entity Identification Additional Properties. Fiscal Attributes will be send in this group.

RestFdgEttAddr

Parameter Name	Data Type	Description
street	String	Street name of the entity's address.
additionalStreetName	String	Extension of street name of the entity's address.
complement	String	Additional information of entity's address, such as neighborhood, county, etc.
city	String	City name of entity's address.
jurisdictionCode	String	Jurisdiction code of entity's address. This can be legal code associated to city, county or any other level.
postalCode	String	Zip or postal code of entity's address.
country	String	Country name of entity's address.
telephone	String	Telephone number of the person who is the contact point.
email	String	E-mail address of the person who is the contact point.
state	String	State name of entity's address.
contactName	String	Contact name.
additionalproperties	List<RestFdgExt>	Collection of address additional properties.

RestFdgExt

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.
description	String	Description of additional property data.

RestFdgExt (identification)

Parameter Name	Data Type	Description
key	String	This column contains the key value of the entity additional properties at entity address level.
value	String	This column contains the value of the entity additional properties at entity address level.
type	String	This column contains the extension type at entity address level.
description	String	Description of additional property data.
references	List<RestFdgRef>	Collection of references.
totaltaxes	List<RestFdgTax>	Collection of taxes.
errors	List<RestFdgError>	Collection of errors.
allowanceCharges	List<RestFdgNonMerc h>	Collection of allowance charges.
additionalproperties	List<RestFdgExt>	Collection of additional properties.
comments	List<RestFdgText>	Collection of comments.

RestFdgDtl

Parameter Name	Data Type	Description
lineNo	long	Sequential number of the item in fiscal document. This number is a sequence starts with number one. Any gap is not allowed.
Item	String	Unique code ID that identifies the item.
itemDesc	String	Long description of the item.
itemShortDesc	String	Short description of the item.
totalAmt	BigDecimal	Total amount for the item inclusive of allowances and charges. Calculation formula is ((unitAmt*qty) - totalAllowance + totalCharge). Rounding applied based on system option.
unitAmt	BigDecimal	Item unit amount. No rounding applied.
uom	String	Unit of measure used in item in the fiscal document.
qty	BigDecimal	Quantity of item in the fiscal document.
totalTaxAmt	BigDecimal	Total amount of taxes applied to the item. This field is sum of all taxAmt at tax level for this item. Rounding applied based on system option.

Parameter Name	Data Type	Description
totalAmtInclTax	BigDecimal	Total amount for item inclusive of taxes. Calculation formula is totalAmt + totalTaxAmt. Rounding applied based on system option.
totalRounding	BigDecimal	The rounding amount (positive or negative) added to produce the line extension totalAmt. Calculation formula ((unitAmt*qty) - totalAllowance + totalCharge) - totalAmt.
totalAllowance	BigDecimal	Total amount of all allowances at item level. This field is sum of all allowances at allowanceCharges level for this item. Rounding applied based on system option.
totalCharge	BigDecimal	Total amount of all charges at item level. This field is sum of all charges at allowanceCharges level for this item. Rounding applied based on system option.
packInd	String	Pack indicator. Valid values: Y (this item is a pack) or N (this item is not a pack).
gtin	String	GTIN number (Global Trade Item Number) of item.
gtinType	String	GTIN Type. Valid values: GTIN-8, GTIN-12, GTIN-13, GTIN-14
vpn	String	Vendor Product Number (VPN) associated with the item.
taxes	List<RestFdgTax>	Collection of item taxes.
references	List<RestFdgRef>	Collection of item references
allowanceCharges	List<RestFdgNonMerch>	Collection of item allowance charges.
packdetails	List<RestFdgDtlPack>	Collection of item pack details.
additionalproperties	List<RestFdgExt>	Collection of item additional properties.
comments	List<RestFdgText>	Collection of item comments.

RestFdgTax

Parameter Name	Data Type	Description
taxType	String	Tax type.
taxCode	String	Tax identification code.
taxRate	BigDecimal	Tax rate when tax type is set as percentage.
taxAmt	BigDecimal	Total tax amount for the tax code. Rounding applied based on system option.

Parameter Name	Data Type	Description
unitTaxAmt	BigDecimal	Tax amount per unit for the tax code.
taxBasis	BigDecimal	Tax basis amount used to calculate tax value based on taxRate field. Rounding applied based on system option.
taxCodeDesc	String	Tax code description. This field can be used to indicate extra code or detailed information about tax code.
additionalproperties	List<RestFdgExt>	Collection of taxes addition properties.
comments	List<RestFdgText>	Collection of taxes comments.

RestFdgText

Parameter Name	Data Type	Description
text	String	Additional information and/or comments. It is used to provide any additional information or legal message important to the fiscal document.
textGroupId	String	Text group ID. It is used to group related comments when there is more than one or when the size of a message is bigger than the size of the text field and it is necessary to generate more than one line.
textGroupSeq	String	Sequence of the text for a given group ID. It is used in case comments information has more than one line.

RestFdgRef

Parameter Name	Data Type	Description
refType	String	Type of the reference information. It is used when there is any reference that it is necessary to add in the fiscal document, as shipment number, contract number, order, ID, and so on. Valid values are pre-defined in Merchandising System.
refCode	String	Reference code/value of reference type.

RestFdgNonMerch

Parameter Name	Data Type	Description
chargeInd	String	Charge indicator.

Parameter Name	Data Type	Description
nonMerchCode	String	Non-merchandise code that identifies a non-merchandise cost used as a charge or allowance.
nonMerchCodeDesc	String	Non-merchandise code description. It is used to provide more detailed about non-merchandise code, for example: allowance details, types of non-merchandise code such as Freight, Insurance, etc.
nonMerchAmt	BigDecimal	Non-merchandise amount.
nonMerchType	String	Non-merchandise type exclusive for Charges.
taxes	List<RestFdgTax>	Collection of taxes for non-merchandise.
additionalproperties	List<RestFdgExt>	Collection of additional properties for non-merchandise.
comments	List<RestFdgText>	Collection of comments for non-merchandises.

packdetails

Parameter Name	Data Type	Description
item	String	Unique alphanumeric ID of a pack component that is part of item pack.
itemQty	BigDecimal	Quantity of the component item that is part of the pack.
itemDesc	String	Description of the component item component.
uom	String	Unit of measure of the component item.

totaltaxes

Parameter Name	Data Type	Description
taxType	String	Not applicable for totalTaxes.
taxCode	String	Tax identification code.
taxRate	BigDecimal	Not applicable for totalTaxes.
taxAmt	BigDecimal	Sum of taxAmt for all items for the tax code. Rounding applied based on system option
unitTaxAmt	BigDecimal	Not applicable for totalTaxes.
taxBasis	BigDecimal	Sum of taxBasis for all items for the tax code. Rounding applied based on system option.
taxCodeDesc	String	Tax code description. This field can be used to indicate extra code or detailed information about tax code.

JSON Structure:

```

"{
  "docId": 17200,
  "requestId": 17207,
  "requestSystem": "OMNI",
  "requestAction": "A",
  "status": "S",
  "docNo": "17200",
  "docType": "WBL",
  "docTypeCode": null,
  "extReqId": "10065007",
  "transactionType": "SALES",
  "issueDate": "2021-03-31T08:19:42-05:00",
  "dueDate": null,
  "printerId": null,
  "docPrintUrl": null,
  "currencyCode": "USD",
  "totalTaxAmt": null,
  "totalNonMerchTaxAmt": null,
  "totalAmtInclTax": null,
  "totalAmtExclTax": null,
  "totalAllowance": null,
  "totalCharge": null,
  "totalRounding": null,
  "terms": null,
  "docLang": "EN",
  "deduceTax": null,
  "createBy": "RMS_ADMIN",
  "createDatetime": "2021-06-16",
  "lastUpdateBy": "RFM01",
  "lastUpdateDatetime": "2021-06-16",
  "entities": [
    {
      "entityType": "SHIPP",
      "entityInternalType": "ST",
      "entityInternalPtnrType": null,
      "entityCode": "990011",
      "name": "QABR-Store Test Washington",
      "legalName": "QABR-Store Test Washington",
      "carrierVehicleRn": null,
      "carrierVehicleState": null,
      "carrierDriverName": null,
      "carrierDriverLn": null,
      "address": {
        "street": "R da Juta",
        "additionalStreetName": "1512",
        "complement": null,
        "city": "Washijngton",
        "jurisdictionCode": null,
        "postalCode": "03308070",
        "country": "US",
        "telephone": "2360873",
        "email": "washington.dias@oracle.com",
        "state": "DC",

```

```

        "contactName": "Washington Dias",
        "additionalproperties": []
    },
    "identification": []
},
{
    "entityType": "DEST",
    "entityInternalType": "CUST",
    "entityInternalPtnrType": null,
    "entityCode": "75",
    "name": "wdias losnisk",
    "legalName": "JADLOG",
    "carrierVehicleRn": null,
    "carrierVehicleState": null,
    "carrierDriverName": null,
    "carrierDriverLn": null,
    "address": {
        "street": "end10",
        "additionalStreetName": "end20",
        "complement": "end30",
        "city": "Washington",
        "jurisdictionCode": "3100104",
        "postalCode": "08573160",
        "country": "US",
        "telephone": null,
        "email": null,
        "state": "DC",
        "contactName": "wdias losnisk",
        "additionalproperties": []
    },
    "identification": []
},
{
    "entityType": "CR",
    "entityInternalType": "PTNR",
    "entityInternalPtnrType": "FF",
    "entityCode": "11",
    "name": "Carrier",
    "legalName": "Carrier",
    "carrierVehicleRn": "CJU0939",
    "carrierVehicleState": "São Paulo",
    "carrierDriverName": "Djalma da Carreta",
    "carrierDriverLn": "77799573959957",
    "address": {
        "street": "15333 JFK Blvd",
        "additionalStreetName": null,
        "complement": null,
        "city": "Houston",
        "jurisdictionCode": null,
        "postalCode": "77032",
        "country": "US",
        "telephone": null,
        "email": "juliano.costa@oracle.com",
        "state": "TX",
        "contactName": "Juliano Costa",
        "additionalproperties": [

```

```

        {
            "key": "NIP",
            "value": "NIP09",
            "type": "VARCHAR2",
            "description": "NIP"
        },
        {
            "key": "POSTALCODE",
            "value": "POST09",
            "type": "VARCHAR2",
            "description": "Postal Code"
        }
    ]
},
"identification": [
    {
        "key": "TAX_ID",
        "value": "332424324",
        "type": "VARCHAR2",
        "description": "TAX ID"
    }
]
}],
"details": [
    {
        "lineNo": 1,
        "item": "100900006",
        "itemDesc": "item regular scenario 3",
        "itemShortDesc": "regular item regular scenario 3",
        "totalAmt": null,
        "unitAmt": null,
        "uom": "EA",
        "qty": 5,
        "totalTaxAmt": null,
        "totalAmtInclTax": null,
        "packInd": "N",
        "gtin": null,
        "gtinType": "ITEM",
        "vpn": null,
        "totalRounding": null,
        "totalAllowance": null,
        "totalCharge": null,
        "taxes": [],
        "references": [],
        "allowanceCharges": [],
        "packdetails": [],
        "additionalproperties": [],
        "comments": []
    },
    {
        "lineNo": 2,
        "item": "100900065",
        "itemDesc": "itemm simple pack scenario 3",
        "itemShortDesc": "itemm simple pack scenario 3",
        "totalAmt": null,
        "unitAmt": null,

```

[illegible]

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_FDG_DTL	Yes	No	No	No
SVC_FDG_DTL_PACK	Yes	No	No	No
SVC_FDG_ERROR	Yes	No	No	No
SVC_FDG_ETT	Yes	No	No	No
SVC_FDG_EXT	Yes	No	No	No
SVC_FDG_HDR	Yes	No	Yes	No
SVC_FDG_NON_MERCH	Yes	No	No	No
SVC_FDG_REF	Yes	No	No	No
SVC_FDG_TAX	Yes	No	No	No
SVC_FDG_TEXT	Yes	No	No	No
FDG_DTL	Yes	No	No	No
FDG_DTL_PACK	Yes	No	No	No
FDG_ERROR	Yes	No	No	No
FDG_ETT	Yes	No	No	No
FDG_EXT	Yes	No	No	No
FDG_HDR	Yes	No	No	No
FDG_NON_MERCH	Yes	No	No	No
FDG_REF	Yes	No	No	No
FDG_TAX	Yes	No	No	No
FDG_TEXT	Yes	No	No	No

Sales Audit Services

The following services are included in this functional area:

- [Data Privacy](#)
 - [Sales Audit Data Privacy Download Service](#)
 - * [Get Personally Identifiable Information](#)
 - [Sales Audit Data Privacy Upload Service](#)
 - * [Delete Personally Identifiable Information](#)
- [Store Day Summary](#)
 - [Mobile Workflow Store Day Download Service](#)
 - * [Get Aggregated Open Store Days](#)
 - * [Get Open Store Day Counts](#)
 - * [Get Open Store Day Details](#)
 - * [Get Open Store Day Error Counts](#)
 - * [Get Open Store Day Over/Short Amounts](#)
 - * [Get Open Store Day Over/Short Counts](#)

- * [Get Open Store Days](#)
- * [Get Store Errors](#)
- * [Get Stores with Open Days](#)
- [Transactions](#)
 - [Consignment Sales Transaction Update](#)
 - [Transaction Download Service](#)
 - * [Get Rejected Transactions Since Previous Query](#)
 - [Transaction Upload Service](#)
 - * [Load Transactions](#)

Data Privacy

The following services are included in this functional area:

- [Sales Audit Data Privacy Download Service](#)
 - [Get Personally Identifiable Information](#)
- [Sales Audit Data Privacy Upload Service](#)
 - [Delete Personally Identifiable Information](#)

Sales Audit Data Privacy Download Service

The following services are included in this functional area:

- [Get Personally Identifiable Information](#)

Get Personally Identifiable Information

The Get Personally Identifiable Information service is the same service used in the Merchandising solution. For Sales Audit, this service can be used to obtain personally identifiable information for the Customer and Employee entities. For more details on this service, see the [Data Privacy Download Service](#) under [Merchandising Services](#) > [Administration](#) > [Data Privacy](#).

Sales Audit Data Privacy Upload Service

The following services are included in this functional area:

- [Delete Personally Identifiable Information](#)

Delete Personally Identifiable Information

The Delete Personally Identifiable Information service is the same service used in the Merchandising solution. For Sales Audit, this service can be used to delete personally identifiable information for the Customer and Employee entities. For more details on this service, see the [Data Privacy Upload Service](#) under [Merchandising Services](#) > [Administration](#) > [Data Privacy](#).

Store Day Summary

The following services are included in this functional area:

- [Mobile Workflow Store Day Download Service](#)
 - [Get Aggregated Open Store Days](#)
 - [Get Open Store Day Counts](#)
 - [Get Open Store Day Details](#)
 - [Get Open Store Day Error Counts](#)
 - [Get Open Store Day Over/Short Amounts](#)
 - [Get Open Store Day Over/Short Counts](#)
 - [Get Open Store Days](#)
 - [Get Store Errors](#)
 - [Get Stores with Open Days](#)

Mobile Workflow Store Day Download Service

The following services are included in this functional area:

- [Get Aggregated Open Store Days](#)
- [Get Open Store Day Counts](#)
- [Get Open Store Day Details](#)
- [Get Open Store Day Error Counts](#)
- [Get Open Store Day Over/Short Amounts](#)
- [Get Open Store Day Over/Short Counts](#)
- [Get Open Store Days](#)
- [Get Store Errors](#)
- [Get Stores with Open Days](#)

Get Aggregated Open Store Days

Business Overview

Retrieves aggregated store day information for all dates or store days older than vdate -5.

Service Type

Get

ReST URL

/getStoreAggregations?
allOlderInd={allOlderInd}&stores={stores}&sortAttrib={sortAttrib}&sortDirection={sortDirection}&
pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description	Valid values
allOlderInd	Yes	Search string for locations ID or Name	ALL, OLDER

Parameter Name	Required	Description	Valid values
stores	No	Comma-separated values for stores	NA
sortAttrib	No	Sort Attribute	STORENAME, AUDITOR, OSVALUE, ERRORCNT, DATASTATUS, OPENDAYS, OSDAYS and OSSUMS
sortDirection	No	Sort Direction	ASC, DESC
pageSize	No	Maximum number of locations to retrieve per page	NA
pageNumber	No	Result page to retrieve	NA

Output

Store

Store Name

Chain

Chain Name

Auditors

Open Days

Over Days

Short Days

Over Amount

Short Amount

Currency Code

Error Count

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SA_ERROR	Yes	No	No	No
SA_HQ_VALUE	Yes	No	No	No
SA_POS_VALUE	Yes	No	No	No
SA_STORE_DATA	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_SYS_VALUE	Yes	No	No	No
V_CHAIN	Yes	No	No	No
V_STORE	Yes	No	No	No

Get Open Store Day Counts

Business Overview

This service provides, at a glance, the number of open stores for which the sales audit manager is responsible. The stores for which the user is responsible are those associated with the user in Sales Audit's employee maintenance via location traits.

Service Type

Get

ReST URL

/summaryOpenStoreDay

Input Parameters

No input.

Output

Record Type --DATE, OLDER, ALL

- **For record type DATE:** five records of type date are displayed for today minus 1 through today minus 5
- **One record type OLDER:** is for store days older than today minus 5
- **One record type ALL:** for all store days

Record Date --Date of date type rows

Open Store Count

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
LOC_TRAITS_MATRIX	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_USER_LOC_TRAITS	Yes	No	No	No

Get Open Store Day Details

Business Overview

The service displays a list of open stores to which the user is assigned, for a single day, for 'OLDER' days, or for 'ALL' days.

Service Type

Get

ReST URL

/getStoreDays?
store={store}&recordType={recordType}&recordDate={recordDate}&sortAttrib={sortAttrib}&sort
Direction={sortDirection}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description	Valid Values
recordType	Yes	Record Type	ALL, OLDER, DATE
recordDate	No	Record Date, required when recordType is DATE	NA
store	No	Store ID	NA
sortAttrib	No	Sort Attribute	STORENAME, AUDITOR, OSVALUE, ERRORCNT, DATASTATUS, OPENDAYS, OSDAYS, and OSSUMS
sortDirection	No	Sort Direction	ASC, DESC
pageSize	No	Maximum number of locations to retrieve per page	NA
pageNumber	No	Result page to retrieve	NA

Output

Store
Store Day Seq No
Auditors
Business Date
Store Name
Chain
Chain Name
Data Status
Data Status Description
Audit Status
Audit Status Description
Audit Changed Datetime
Fuel Status
Fuel Status Description
Over Short Amount
Currency Code
Error Count
Transaction Count

Loaded File Count

Expected File Count

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
LOC_TRAITS_MATRIX	Yes	No	No	No
SA_ERROR	Yes	No	No	No
SA_HQ_VALUE	Yes	No	No	No
SA_POS_VALUE	Yes	No	No	No
SA_STORE_DATA	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_SYS_VALUE	Yes	No	No	No
SA_SYSTEM_OPTIONS	Yes	No	No	No
SA_TOTAL	Yes	No	No	No
SA_TRAN_HEAD	Yes	No	No	No
SA_USER_LOC_TRAITS	Yes	No	No	No
V_CHAIN	Yes	No	No	No
V_CODE_DETAIL	Yes	No	No	No
V_STORE	Yes	No	No	No

Get Open Store Day Error Counts

Business Overview

This service provides, at a glance, the number outstanding errors on the specified days for stores for which the sales audit manager is responsible. An outstanding error is defined as an error that exists against a store day that has not been overridden.

Service Type

Get

ReST URL

/summaryError

Input Parameters

No input.

Output

Record Type --DATE, OLDER, ALL

- **For record type DATE:** five records of type date are displayed for today minus 1 through today minus 5
- **One record type OLDER:** is for store days older than today minus 5
- **One record type ALL:** for all store days

Record Date --Date of date type rows
Error Count

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
LOC_TRAITS_MATRIX	Yes	No	No	No
SA_ERROR	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_USER_LOC_TRAITS	Yes	No	No	No

Get Open Store Day Over/Short Amounts

Business Overview

This service provides at a glance the sums of all overages and all shortages for all open stores on a given day for which the sales audit manager is responsible. If all locations to which the user is responsible have the same local currency, all monetary values will be displayed in the local currency. Otherwise, all monetary values will be displayed in the retailer's primary currency. If the Over/Short value for the store day is a positive value it is considered an overage, if the Over/Short value for the store day is a negative value it is a shortage.

Service Type

Get

ReST URL

/summaryOverShortAmount

Input Parameters

No input.

Output

Record Type --DATE, OLDER, ALL

- **For record type DATE:** Five records of type date are displayed for today minus 1 through today minus 5
- **One record type OLDER:** is for store days older than today minus 5
- **One record type ALL:** for all store days

Record Date --Date of date type rows
Over Amount
Short Amount
Currency Code

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
LOC_TRAITS_MATRIX	Yes	No	No	No
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No
SA_HQ_VALUE	Yes	No	No	No
SA_POS_VALUE	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_SYS_VALUE	Yes	No	No	No
SA_TOTAL	Yes	No	No	No
SA_USER_LOC_TRAITS	Yes	No	No	No
STORE	Yes	No	No	No

Get Open Store Day Over/Short Counts

Business Overview

This service provides, at a glance, the count of overages and the count of shortages for all open stores on a given day for which the sales audit manager is responsible. If the Over/Short value for the store day is a positive value it is considered an overage, if the Over/Short value for the store day is a negative value it is a shortage.

Service Type

Get

ReST URL

/summaryOverShortCount

Input Parameters

No input.

Output

Record Type --DATE, OLDER, ALL

- **For record type DATE:** five records of type date are displayed for today minus 1 through today minus 5
- **One record type OLDER:** is for store days older than today minus 5
- **One record type ALL:** for all store days

Record Date --Date of date type rows

Over Count

Short Count

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
LOC_TRAITS_MATRIX	Yes	No	No	No
SA_HQ_VALUE	Yes	No	No	No
SA_POS_VALUE	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_SYS_VALUE	Yes	No	No	No
SA_TOTAL	Yes	No	No	No
SA_USER_LOC_TRAITS	Yes	No	No	No
STORE	Yes	No	No	No

Get Open Store Days

Business Overview

This web service allows the user to find which store days have records that needs attention.

Service Type

Get

ReST URL

/getStoreDateInd?store={store}

Input Parameters

Parameter Name	Required	Description	Valid values
store	Yes	Store ID	NA

Output

Record Type --DATE, OLDER, ALL

- **For record type DATE:** five records of type date are displayed for today minus 1 through today minus 5
- **One record type OLDER:** is for store days older than today minus 5
- **One record type ALL:** for all store days

Record Date --Date of date type rows

Store Has Value indicator

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SA_STORE_DAY	Yes	No	No	No
V_STORE	Yes	No	No	No

Get Store Errors

Business Overview

Retrieves summary of store day errors.

Service Type

Get

ReST URL

/getStoreErrors?store={store}&recordType={recordType}&recordDate={recordDate}

Input Parameters

Parameter Name	Required	Description	Valid values
recordType	Yes	Record Type	ALL, OLDER, DATE
recordDate	No	Record Date, required when recordType is DATE	NA
store	No	Store ID	NA

Output

Store

Error Code

Error Description

Error Percentage

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SA_ERROR	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
V_SA_ERROR	Yes	No	No	No
V_STORE	Yes	No	No	No

Get Stores with Open Days

Business Overview

This web service enables store search and returns aggregated store information.

Service Type

Get

ReST URL

/storeSearch?
searchString={searchString}&searchFilter={searchFilter}&sortAttrib={sortAttrib}&sortDirection={
sortDirection}&pageSize={pageSize}&pageNumber={pageNumber}

Input Parameters

Parameter Name	Required	Description	Valid values
searchString	Yes	Search string for locations ID or Name	NA
searchFilter	Yes	Search all stores or assigned stores	ALL, ASSIGN
sortAttrib	No	Sort Attribute	STORENAME, AUDITOR, OSVALUE, ERRORCNT, DATASTATUS, OPENDAYS, OSDAYS, and OSSUMS
sortDirection	No	Sort Direction	ASC, DESC
pageSize	No	Maximum number of locations to retrieve per page	NA
pageNumber	No	Result page to retrieve	NA

Output

Store
Store Name
Chain
Chain Name
Auditors
Open Days
Over Days
Short Days
Over Amount
Short Amount
Currency Code
Error Count

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
LOC_TRAITS_MATRIX	Yes	No	No	No
SA_ERROR	Yes	No	No	No
SA_HQ_VALUE	Yes	No	No	No
SA_POS_VALUE	Yes	No	No	No

TABLE	SELECT	INSERT	UPDATE	DELETE
SA_STORE_DATA	Yes	No	No	No
SA_STORE_DAY	Yes	No	No	No
SA_SYS_VALUE	Yes	No	No	No
SA_TOTAL	Yes	No	No	No
SA_TRAN_HEAD	Yes	No	No	No
SA_USER_LOC_TRAITS	Yes	No	No	No
V_CHAIN	Yes	No	No	No
V_STORE	Yes	No	No	No

Transactions

The following services are included in this functional area:

- [Consignment Sales Transaction Update](#)
- [Transaction Download Service](#)
 - [Get Rejected Transactions Since Previous Query](#)
- [Transaction Upload Service](#)
 - [Load Transactions](#)

Consignment Sales Transaction Update

Functional Area

Financial

Business Overview

The service provides the ability to accept or provide Adjusted values of Consignment Rate and Consignment unit cost for consignment items staged for Rate Review.

Service Type

POST

ReST URL

Resa/consignmentSalesTransactionUpdate

Input Payload Details

Table 5-1317 Update - Object. See list of elements for detail

Element Name	Required	Data Type	Description
collectionSize	Yes	Number (4)	Number of items in the collection. The maximum number of the items allowed in a collection is 9999.
consignmentSales	Yes	Collection of Object	References a collection of Store, Day, Store Day sequence number, Tran sequence number, Item sequence number.

Table 5-1318 ConsignmentSales - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number to which this transaction belongs.
day	Yes	Number (3)	Day number on the transaction.
storeDaySeqNo	Yes	Number (20)	This unique identifier for a store/day combination.
tranSeqNo	Yes	Number (20)	The sales audit system's transaction identifier.
itemSeqNo	Yes	Number (4)	Sequence number of item within a transaction.
itemStatus	No	String (6)	The status of the item. Valid values are 'SALE', 'RETURN', 'ORD', 'LCO'
itemType	No	String (6)	The type of the item. Valid values are ITEM, NMITEM and REF.
item	No	String (25)	Unique identifier for the item.
refItem	No	String (25)	Contains a reference item to the item field.
nonMerchItem	No	String (25)	Contains a non-merchandise item.
reviewedWithNoUpdates	Yes	String (1)	The review indicator allows for accepting the existing rate/cost detail without any updates when set to Y. Valid values are Y and N.
adjustedConsignmentRate	No	Number (12,4)	This column holds the preferred adjusted consignment rate.
adjustedConsignmentUnitCost	No	Number (20,4)	This column holds the preferred adjusted consignment unit cost.

Table 5-1319 UpdateError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
consignmentSalesError	Yes	Collection of Object	

Table 5-1320 ConsignmentSalesError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number to which this transaction belongs.
day	Yes	Number (3)	Day number on the transaction.
storeDaySeqNo	Yes	Number (20)	This unique identifier for a store/day combination.
tranSeqNo	Yes	Number (20)	The sales audit system's transaction identifier.
itemSeqNo	Yes	Number (4)	Sequence number of item within a transaction.
errorMessages	Yes	Array of String	List of errors.

Sample Input Message

```
{
  "collectionSize": 1000,
  "consignmentSales": [
    {
      "store": 1521,
      "day": 15,
      "storeDaySeqNo": 29000201,
      "tranSeqNo": 102308,
      "itemSeqNo": 1000,
      "itemStatus": "S",
      "itemType": "ITEM",
      "item": "102350079",
      "refItem": "1234560000081",
      "nonMerchItem": "445101",
      "reviewedWithNoUpdates": "N",
      "adjustedConsignmentRate": 60,
      "adjustedConsignmentUnitCost": 20.2131
    }
  ]
}
```

Response Code: 200 (Success)

Response Code: 400 (Error)

In case of error, the following standard error response will be returned. The element `validationErrors` is present when input the payload or input parameters are not as per the schema definition of this service. The element `businessError` will be present if the payload passes schema validation but an exception is caught while processing business logic.

Table 5-1321 UpdateError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
consignmentSalesError	Yes	Collection of Object	

Table 5-1322 ConsignmentSalesError - Object. See list of elements for detail

Element Name	Required	Data Type	Description
store	Yes	Number (10)	The store number to which this transaction belongs.
day	Yes	Number (3)	Day number on the transaction.
storeDaySeqNo	Yes	Number (20)	This unique identifier for a store/day combination.
tranSeqNo	Yes	Number (20)	The sales audit system's transaction identifier.
itemSeqNo	Yes	Number (4)	Sequence number of item within a transaction.
errorMessages	Yes	Array of String	List of errors.

Sample Error Message

```
{
  "status": "ERROR",
  "message": "Error found in validation of input payload",
  "validationErrors": [
    {
      "error": "must be one of Y, N",
      "field": "createRecord.arg0.approveInd",
      "inputValue": "X"
    }
  ],
  "businessError": [
    {
      "consignmentSalesError": [
        {
          "store": 109770,
          "day": 100,
          "storeDaySeqNo": 293567,
          "tranSeqNo": 102308,
          "itemSeqNo": 1000,
          "errorMessages": [
            "String"
          ]
        }
      ]
    }
  ]
}
```

Transaction Download Service

The following services are included in this functional area:

- [Get Rejected Transactions Since Previous Query](#)

Get Rejected Transactions Since Previous Query

Functional Area

Financial

Business Overview

The purpose of this service is to return transactions that were rejected during processing to the calling process or application, for use in troubleshooting and error resolution.

Sales Service

This services fetches and returns transactions from the staging table SVC_RTLOG_DATA_REJECT that meet the input criteria. Transactions failing validations in the sacreatert log.ksh process are inserted into this staging table during processing.

Service Type

Put

ReST URL

Resa/getRejectedTransactions

Input Parameters

Table 5-1323 SvcprovRejectedRecRDO

Parameter Name	Data Type
pageSize	Number
pageNumber	Number
readUnreadErrorInd	String
startDate	String (Format :YYYYMMDD)
endDate	String (Format :YYYYMMDD)
svcprovRejStoreRDOs	List<SvcprovRejStoreRecRDO>
svcprovRejErrorcodeRDOs	List<SvcprovRejErrorcodeRecRDO>
svcprovRejTransactionidRDOs	List<SvcprovRejTransactionidRecRDO>

Table 5-1324 SvcprovRejStoreRecRDO

Parameter Name	Data Type
store	String

Table 5-1325 SvcprovRejErrorcodeRecRDO

Parameter Name	Data Type
errorCode	String

Table 5-1326 SvcprovRejTransactionidRecRDO

Parameter Name	Data Type
transactionId	String

JSON Input Structure:

```
[
  {
    "pageSize": null,
    "pageNumber": null,
    "readUnreadErrorInd": null,
    "startDate": null,
    "endDate": null,
    "svcprovRejStoreRDOs": [
      {
        "store": null
      }
    ],
    "svcprovRejErrorcodeRDOs": [
      {
        "errorCode": null
      }
    ],
    "svcprovRejTransactionidRDOs": [
      {
        "transactionId": null
      }
    ]
  }
]
```

Output

totalRecordCount	Number
Parameter Name	Data Type
results	Table of SVCPROV_REJ_THREAD_REC

Table 5-1327 SVCPROV_REJ_THREAD_REC

Parameter Name	Data Type
Store	Number
businessDate	Date

Table 5-1327 (Cont.) SVCPROV_REJ_THEAD_REC

Parameter Name	Data Type
errorCode	String
errorDesc	String
recType	String
register	String
transactionDate	Date
transactionNo	Number
cashier	String
salesperson	String
transactionType	String
subTransactionType	String
origSalesNo	Number
origRegNo	String
reasonCode	String
vendorNo	String
vendorInvNo	String
paymentRefNo	String
prfDlvyNo	String
refNo1	String
refNo2	String
refNo3	String
refNo4	String
value	BigDecimal
bannerId	String
roundedAmt	BigDecimal
roundedOffAmt	BigDecimal
creditPromoId	String
refNo25	String
refNo26	String
refNo27	String
tranProcessSys	String
refNo28	String
refNo29	String
refNo30	String
refNo31	String
salesThattTbl	List<SvcprovSalesAttrRecRDO>
salesTcustTbl	List< SvcprovSalesTcustRecRDO>
salesCattTbl	List< SvcprovSalesAttrRecRDO>
salesTitemTbl	List< SvcprovSalesTitemRecRDO>
salesTtaxTbl	List< SvcprovSalesTtaxRecRDO>

Table 5-1327 (Cont.) SVCPROV_REJ_THEAD_REC

Parameter Name	Data Type
salesTpymtTbl	List< SvcprovSalesTpymtRecRDO>
salesTtendTbl	List<SvcprovSalesTtendRecRDO>
salesTtailTbl	List<SvcprovSalesCounterRecRDO>

Table 5-1328 SvcprovSalesAttrRecRDO

Parameter Name	Data Type
recType	String
attributeType	String
attributeValue	String

Table 5-1329 SvcprovSalesTcustRecRDO

Parameter Name	Data Type
recType	String
customerId	String
customerType	String
customerName	String
address1	String
address2	String
city	String
state	String
zipCode	String
country	String
homePhone	String
workPhone	String
email	String
birthdate	Date

Table 5-1330 SvcprovSalesTitemRecRDO

Parameter Name	Data Type
recType	String
itemStatus	String
itemType	String
itemNoType	String
formatId	String
item	String
refItem	String

Table 5-1330 (Cont.) SvcprovSalesTitemRecRDO

Parameter Name	Data Type
nonMerchItem	String
voucherNo	String
quantity	String
sellingUom	String
unitRetail	BigDecimal
overrideReason	String
originalUnitRetail	BigDecimal
taxableIndicator	String
pump	String
refNo5	String
refNo6	String
refNo7	String
refNo8	String
itemSwipedInd	String
returnReasonCode	String
salesperson	String
expirationDate	Date
dropShipInd	String
uomQty	BigDecimal
catchWeightInd	String
sellingItem	String
custOrdLineNo	Number
mediaId	Number
totalIgtaxAmount	BigDecimal
uniqueId	String
custOrdNo	String
custOrdDate	Date
fulfillmentOrdNo	String
noInventoryReturn	String
salesType	String
returnWh	Number
returnDeposition	String
originalStore	Number
originalTransactionNo	Number
fulfillmentLocType	String
fulfillmentLoc	Number
postingStore	String
salesItattTbl	List<SvcprovSalesAttrRecRDO>
salesIdiscTbl	List<SvcprovSalesIdiscRecRDO>

Table 5-1330 (Cont.) SvcprovSalesTitemRecRDO

Parameter Name	Data Type
salesIgtaxTbl	List<SvcprovSalesIgtaxRecRDO>

Table 5-1331 SvcprovSalesIdiscRecRDO

Parameter Name	Data Type
recType	String
merchPromoNo	String
discountRefNo	String
discountType	String
couponNo	String
couponRefNo	String
quantity	BigDecimal
unitDiscountAmount	BigDecimal
refNo13	String
refNo14	String
refNo15	String
refNo16	String
uomQty	BigDecimal
catchWeightInd	String
promoComponent	Number
salesTdattTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1332 SvcprovSalesIgtaxRecRDO

Parameter Name	Data Type
recType	String
taxAuthority	String
igntaxCode	String
igntaxRate	BigDecimal
igntaxAmount	BigDecimal
refNo21	String
refNo22	String
refNo23	String
refNo24	String
salesIxattTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1333 SvcprovSalesTtaxRecRDO

Parameter Name	Data Type
recType	String
taxCode	String
taxAmount	BigDecimal
refNo17	String
refNo18	String
refNo19	String
refNo20	String
transactionTaxAttributeTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1334 SvcprovSalesTpymtRecRDO

Parameter Name	Data Type
recType	String
paymentAmount	BigDecimal

Table 5-1335 SvcprovSalesTtendRecRDO

Parameter Name	Data Type
recType	String
tendTypeGroup	String
tendTypeId	Number
tenderAmount	BigDecimal
ccNo	String
ccAuthNo	String
ccAuthSource	String
ccCardholderVerification	String
ccExpirationDate	Date
ccEntryMode	String
ccTerminalId	String
ccSpecialCondition	String
ccToken	String
voucherNo	String
couponNbr	String
couponRefNo	String
chequeAccNo	String
chequeNo	Number
identificationMethod	String
identificationId	String
originalCurrency	String

Table 5-1335 (Cont.) SvcprovSalesTtendRecRDO

Parameter Name	Data Type
originalCurrencyAmount	BigDecimal
refNbr9	String
refNbr10	String
refNbr11	String
refNbr12	String
transactionTenderAttributeTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1336 SvcprovSalesCounterRecRDO

Parameter Name	Data Type
recType	String
recCounter	Number

The output will contain the status of the request including validation errors, if any.

JSON Structure:

```
{
  "totalRecordCount": null,
  "results": [
    {
      "store": null,
      "businessDateDisplay": null,
      "businessDate": null,
      "errorCode": null,
      "errorDesc": null,
      "recType": "THEAD",
      "register": null,
      "transactionDateDisplay": null,
      "transactionDate": null,
      "transactionNo": null,
      "cashier": null,
      "salesperson": null,
      "transactionType": null,
      "subTransactionType": null,
      "origSalesNo": null,
      "origRegNo": null,
      "reasonCode": null,
      "vendorNo": null,
      "vendorInvcNo": null,
      "paymentRefNo": null,
      "prfDlvyNo": null,
      "refNo1": null,
      "refNo2": null,
      "refNo3": null,
      "refNo4": null,
      "value": null,
      "bannerId": null,
    }
  ]
}
```

```

"roundedAmt": null,
"roundedOffAmt": null,
"creditPromoId": null,
"refNo25": null,
"refNo26": null,
"refNo27": null,
"tranProcessSys": null,
"refNo28": null,
"refNo29": null,
"refNo30": null,
"refNo31": null,
"salesThattTbl": [
  {
    "recType": "THATT",
    "attributeType": null,
    "attributeValue": null
  }
],
"salesTcustTbl": [
  {
    "recType": "TCUST",
    "customerId": null,
    "customerType": null,
    "customerName": null,
    "address1": null,
    "address2": null,
    "city": null,
    "state": null,
    "zipCode": null,
    "country": null,
    "homePhone": null,
    "workPhone": null,
    "email": null,
    "birthdateDisplay": null,
    "birthdate": null
  },
  {
    "recType": "TCUST",
    "customerId": null,
    "customerType": null,
    "customerName": null,
    "address1": null,
    "address2": null,
    "city": null,
    "state": null,
    "zipCode": null,
    "country": null,
    "homePhone": null,
    "workPhone": null,
    "email": null,
    "birthdateDisplay": null,
    "birthdate": null
  }
],
"salesCattTbl": [

```

```

    {
      "recType": "CATT",
      "attributeType": null,
      "attributeValue": null
    }
  ],
  "salesTitemTbl": [
    {
      "recType": "TITEM",
      "itemStatus": null,
      "itemType": null,
      "itemNoType": null,
      "formatId": null,
      "item": null,
      "refItem": null,
      "nonMerchItem": null,
      "voucherNo": null,
      "quantity": null,
      "sellingUom": null,
      "unitRetail": null,
      "overrideReason": null,
      "originalUnitRetail": null,
      "taxableIndicator": "Y",
      "pump": null,
      "refNo5": null,
      "refNo6": null,
      "refNo7": null,
      "refNo8": null,
      "itemSwipedInd": null,
      "returnReasonCode": null,
      "salesperson": null,
      "expirationDate": null,
      "dropShipInd": null,
      "uomQty": null,
      "catchWeightInd": null,
      "sellingItem": null,
      "custOrdLineNo": null,
      "mediaId": null,
      "totalIgtaxAmount": null,
      "uniqueId": null,
      "custOrdNo": null,
      "custOrdDateDisplay": null,
      "custOrdDate": null,
      "fulfillmentOrdNo": null,
      "noInventoryReturn": null,
      "salesType": null,
      "returnWh": null,
      "returnDeposition": null,
      "originalStore": null,
      "originalTransactionNo": null,
      "fulfillmentLocType": null,
      "fulfillmentLoc": null,
      "postingStore": null,
      "salesItattTbl": [
        {

```

```

        "recType": "ITATT",
        "attributeType": null,
        "attributeValue": null

    },
    {
        "recType": "ITATT",
        "attributeType": null,
        "attributeValue": null

    }
],
"salesIdiscTbl": [
    {
        "recType": "IDISC",
        "merchPromoNo": null,
        "discountRefNo": null,
        "discountType": null,
        "couponNo": null,
        "couponRefNo": null,
        "quantity": null,
        "unitDiscountAmount": null,
        "refNo13": null,
        "refNo14": null,
        "refNo15": null,
        "refNo16": null,
        "uomQty": null,
        "catchWeightInd": null,
        "promoComponent": null,
        "salesTdattTbl":

    }
],
"salesIgtaxTbl": [
    {
        "recType": "IGTAX",
        "taxAuthority": null,
        "igtaxCode": null,
        "igtaxRate": null,
        "igtaxAmount": null,
        "refNo21": null,
        "refNo22": null,
        "refNo23": null,
        "refNo24": null,
        "salesIxattTbl": [
            {
                "recType": "IXATT",
                "attributeType": null,
                "attributeValue": null

            }
        ]
    }
]
},
"salesTtaxTbl": [

```

```

{
  "recType": "TTAX",
  "taxCode": null,
  "taxAmount": null,
  "refNo17": null,
  "refNo18": null,
  "refNo19": null,
  "refNo20": null,
  "salesTxattTbl": [
    {
      "recType": "TXATT",
      "attributeType": null,
      "attributeValue": null
    }
  ]
},
"salesTpymtTbl": [
  {
    "recType": "TPYMT",
    "paymentAmount": null
  }
],
"salesTtendTbl": [
  {
    "recType": "TTEND",
    "tendTypeGroup": null,
    "tendTypeId": null,
    "tenderAmount": null,
    "ccNo": null,
    "ccAuthNo": null,
    "ccAuthSource": null,
    "ccCardholderVerification": null,
    "ccExpirationDateDisplay": null,
    "ccExpirationDate": null,
    "ccEntryMode": null,
    "ccTerminalId": null,
    "ccSpecialCondition": null,
    "ccToken": null,
    "voucherNo": null,
    "couponNbr": null,
    "couponRefNo": null,
    "chequeAccNo": null,
    "chequeNo": null,
    "identificationMethod": null,
    "identificationId": null,
    "originalCurrency": null,
    "originalCurrencyAmount": null,
    "refNbr9": null,
    "refNbr10": null,
    "refNbr11": null,
    "refNbr12": null,
    "salesTtattTbl": [
      {
        "recType": "TTATT",
        "attributeType": null,

```



```

        "attributeValue": null
      }
    ]
  },
  "salesTtailTbl": []
}
]
}

```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_RTLOG_DATA_REJECT	Yes	No	Yes	No

Transaction Upload Service

The following services are included in this functional area:

- [Load Transactions](#)

Load Transactions

Functional Area

Financial

Business Overview

This service accepts transactions from a Point of Sale (POS) or Order Management Solution (OMS) into Sales Audit. This service provides an alternative to transactions processed from an RTLOG file through the Sales Audit sales process documented under the Scheduled Integration section.

This service accepts transactions from a POS or OMS system in a JSON format and loads them into the staging table SVC_RTLOG_DATA_LOAD for further processing. This service performs minimal data validation as the majority of the sales data validation remains in the Sales Audit import process. The transactions staged through this service are extracted into the RTLOG format by the sacreatertlog.ksh process and processed by the saimptlog process.

In addition, this service supports a near-real-time update of inventory impacted by sales transactions by recording sales units in the SA_INPROGRESS_SALES table. The units record in this table for an item/location can be netted out of stock-on-hand for a more accurate, real-time view of available inventory. When these sales are processed completely through auditing, the in-progress units will be negated.

Service Type

Post

ReST URL

Resa/salesService

Input Parameters

Table 5-1337 SvcprovSalesServiceRecRDO

Parameter Name	Data Type
recType	String
fileCreateDate	String (Format :YYYYMMDDHHMMSS)
businessDate	String(Format: YYYYMMDD)
store	Number
referenceNbr	String
rtlogOrgSys	String
transactionHeadTbl	List<SvcprovSalesTheadRecRDO>
salesFtailTbl	List<SvcprovSalesCounterRecRDO>

Table 5-1338 SvcprovSalesTheadRecRDO

Parameter Name	Data Type
recType	String
register	String
transactionDate	String (Format :YYYYMMDDHHMMSS)
transactionNo	Number
cashier	String
salesperson	String
transactionType	String
subTransactionType	String
origSalesNo	Number
origRegNo	String
reasonCode	String
vendorNo	String
vendorInvNo	String
paymentRefNo	String
prfDlvyNo	String
refNo1	String
refNo2	String
refNo3	String
refNo4	String
value	BigDecimal
bannerId	String
roundedAmt	BigDecimal
roundedOffAmt	BigDecimal
creditPromoId	String
refNo25	String

Table 5-1338 (Cont.) SvcprovSalesTheadRecRDO

Parameter Name	Data Type
refNo26	String
refNo27	String
tranProcessSys	String
refNo28	String
refNo29	String
refNo30	String
refNo31	String
transactionHeadAttributeTbl List<SvcprovSalesAttrRecRDO>	
transactionCustomerTbl	List< SvcprovSalesTcustRecRDO>
customerAttributeTbl	List< SvcprovSalesAttrRecRDO>
transactionItemTbl	List< SvcprovSalesTitemRecRDO>
transactionTaxTbl	List< SvcprovSalesTtaxRecRDO>
transactionPaymentTbl	List< SvcprovSalesTpymtRecRDO>
transactionTenderTbl	List<SvcprovSalesTtendRecRDO>
transactionTrailerTbl	List<SvcprovSalesCounterRecRDO>

Table 5-1339 SvcprovSalesAttrRecRDO

Parameter Name	Data Type
recType	String
attributeType	String
attributeValue	String

Table 5-1340 SvcprovSalesTcustRecRDO

Parameter Name	Data Type
recType	String
customerId	String
customerType	String
customerName	String
address1	String
address2	String
city	String
state	String
zipCode	String
country	String
homePhone	String
workPhone	String

Table 5-1340 (Cont.) SvcprovSalesTcustRecRDO

Parameter Name	Data Type
email	String
birthdate	String (Format: YYYYMMDD)

Table 5-1341 SvcprovSalesTitemRecRDO

Parameter Name	Data Type
recType	String
itemStatus	String
itemType	String
itemNoType	String
formatId	String
item	String
refItem	String
nonMerchItem	String
voucherNo	String
quantity	String
sellingUom	String
unitRetail	BigDecimal
overrideReason	String
originalUnitRetail	BigDecimal
taxableIndicator	String
pump	String
refNo5	String
refNo6	String
refNo7	String
refNo8	String
itemSwipedInd	String
returnReasonCode	String
salesperson	String
expirationDate	String(Format: YYYYMMDD)
dropShipInd	String
uomQty	BigDecimal
catchWeightInd	String
sellingItem	String
custOrdLineNo	Number
mediaId	Number
totalIgtaxAmount	BigDecimal
uniqueId	String
custOrdNo	String

Table 5-1341 (Cont.) SvcprovSalesTitemRecRDO

Parameter Name	Data Type
custOrdDate	String(Format: YYYYMMDDHHMMSS)
fulfillmentOrdNo	String
noInventoryReturn	String
salesType	String
returnWh	Number
returnDeposition	String
originalStore	Number
originalTransactionNo	Number
fulfillmentLocType	String
fulfillmentLoc	Number
postingStore	String
transactionItemAttributeTbl	List<SvcprovSalesAttrRecRDO>
transactionItemDiscountTbl	List<SvcprovSalesIdiscRecRDO>
transactionItemTaxTbl	List<SvcprovSalesIgtaxRecRDO>
consignmentRate	BigDecimal
consignmentUnitCost	BigDecimal
inventoryIdentifierType	String
inventoryId	String

Table 5-1342 SvcprovSalesIdiscRecRDO

Parameter Name	Data Type
recType	String
merchPromoNo	String
discountRefNo	String
discountType	String
couponNo	String
couponRefNo	String
quantity	BigDecimal
unitDiscountAmount	BigDecimal
refNo13	String
refNo14	String
refNo15	String
refNo16	String
uomQty	BigDecimal
catchWeightInd	String
promoComponent	Number
transactionItemDiscountAttributeTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1343 SvcprovSalesIgtaxRecRDO

Parameter Name	Data Type
recType	String
taxAuthority	String
igtaxCode	String
igtaxRate	BigDecimal
igtaxAmount	BigDecimal
refNo21	String
refNo22	String
refNo23	String
refNo24	String
transactionItemTaxAttributeTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1344 SvcprovSalesTtaxRecRDO

Parameter Name	Data Type
recType	String
taxCode	String
taxAmount	BigDecimal
refNo17	String
refNo18	String
refNo19	String
refNo20	String
transactionTaxAttributeTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1345 SvcprovSalesTpymtRecRDO

Parameter Name	Data Type
recType	String
paymentAmount	BigDecimal

Table 5-1346 SvcprovSalesTtendRecRDO

Parameter Name	Data Type
recType	String
tendTypeGroup	String
tendTypeId	Number
tenderAmount	BigDecimal
ccNo	String
ccAuthNo	String
ccAuthSource	String

Table 5-1346 (Cont.) SvcprovSalesTtendRecRDO

Parameter Name	Data Type
ccCardholderVerification	String
ccExpirationDate	String(Format: YYYYMMDD)
ccEntryMode	String
ccTerminalId	String
ccSpecialCondition	String
ccToken	String
voucherNo	String
couponNbr	String
couponRefNo	String
chequeAccNo	String
chequeNo	Number
identificationMethod	String
identificationId	String
originalCurrency	String
originalCurrencyAmount	BigDecimal
refNbr9	String
refNbr10	String
refNbr11	String
refNbr12	String
transactionTenderAttributeTbl	List<SvcprovSalesAttrRecRDO>

Table 5-1347 SvcprovSalesCounterRecRDO

Parameter Name	Data Type
recType	String
recCounter	Number

JSON Structure:

```
[
  {
    "recType": null,
    "fileCreateDate": null,
    "businessDate": null,
    "store": null,
    "referenceNbr": null,
    "rtlogOrgSys": null,
    "transactionHeadTbl": [
      {
        "recType": null,
        "register": null,
        "transactionDate": null,
        "transactionNo": null,
```

```
"cashier": null,
"salesperson": null,
"transactionType": null,
"subTransactionType": null,
"origSalesNo": null,
"origRegNo": null,
"reasonCode": null,
"vendorNo": null,
"vendorInvcNo": null,
"paymentRefNo": null,
"prfDlvyNo": null,
"refNo1": null,
"refNo2": null,
"refNo3": null,
"refNo4": null,
"value": null,
"bannerId": null,
"roundedAmt": null,
"roundedOffAmt": null,
"creditPromoId": null,
"refNo25": null,
"refNo26": null,
"refNo27": null,
"tranProcessSys": null,
"refNo28": null,
"refNo29": null,
"refNo3null": null,
"refNo31": null,
"transactionHeadAttributeTbl": [
  {
    "recType": null,
    "attributeType": null,
    "attributeValue": null
  }
],
"transactionCustomerTbl": [
  {
    "recType": null,
    "customerId": null,
    "customerType": null,
    "customerName": null,
    "address1": null,
    "address2": null,
    "city": null,
    "state": null,
    "zipCode": null,
    "country": null,
    "homePhone": null,
    "workPhone": null,
    "email": null,
    "birthdate": null
  }
],
"customerAttributeTbl": [
  {
    "recType": null,
```



```

        "attributeType": null,
        "attributeValue": null
    }
],
"transactionItemTbl": [
    {
        "recType": null,
        "itemStatus": null,
        "itemType": null,
        "itemNoType": null,
        "formatId": null,
        "item": null,
        "refItem": null,
        "nonMerchItem": null,
        "voucherNo": null,
        "quantity": null,
        "sellingUom": null,
        "unitRetail": null,
        "overrideReason": null,
        "originalUnitRetail": null,
        "taxableIndicator": null,
        "pump": null,
        "refNo5": null,
        "refNo6": null,
        "refNo7": null,
        "refNo8": null,
        "itemSwipedInd": null,
        "returnReasonCode": null,
        "salesperson": null,
        "expirationDate": null,
        "dropShipInd": null,
        "uomQty": null,
        "catchWeightInd": null,
        "sellingItem": null,
        "custOrdLineNo": null,
        "mediaId": null,
        "totalIgtaxAmount": null,
        "uniqueId": null,
        "custOrdNo": null,
        "custOrdDate": null,
        "fulfillmentOrdNo": null,
        "noInventoryReturn": null,
        "salesType": null,
        "returnWh": null,
        "returnDeposition": null,
        "originalStore": null,
        "originalTransactionNo": null,
        "fulfillmentLocType": null,
        "fulfillmentLoc": null,
        "postingStore": null,
        "transactionItemAttributeTbl": [
            {
                "recType": null,
                "attributeType": null,
                "attributeValue": null
            }
        ]
    }
]

```

```

    ],
    "transactionItemDiscountTbl": [
      {
        "recType": null,
        "merchPromoNo": null,
        "discountRefNo": null,
        "discountType": null,
        "couponNo": null,
        "couponRefNo": null,
        "quantity": null,
        "unitDiscountAmount": null,
        "refNo13": null,
        "refNo14": null,
        "refNo15": null,
        "refNo16": null,
        "uomQty": null,
        "catchWeightInd": null,
        "promoComponent": null,
        "transactionItemDiscountAttributeTbl": [
          {
            "recType": null,
            "attributeType": null,
            "attributeValue": null
          }
        ]
      }
    ],
    "transactionItemTaxTbl": [
      {
        "recType": null,
        "taxAuthority": null,
        "igtaxCode": null,
        "igtaxRate": null,
        "igtaxAmount": null,
        "refNo21": null,
        "refNo22": null,
        "refNo23": null,
        "refNo24": null,
        "transactionItemTaxAttributeTbl": [
          {
            "recType": null,
            "attributeType": null,
            "attributeValue": null
          }
        ]
      }
    ],
    "consignmentRate": null,
    "consignmentUnitCost": null,
    "inventoryIdentifierType": null,
    "inventoryId": null
  },
  "transactionTaxTbl": [
    {
      "recType": null,

```

```

        "taxCode": null,
        "taxAmount": null,
        "refNo17": null,
        "refNo18": null,
        "refNo19": null,
        "refNo2null": null,
        "transactionTaxAttributeTbl": [
            {
                "recType": null,
                "attributeType": null,
                "attributeValue": null
            }
        ]
    },
    "transactionPaymentTbl": [
        {
            "recType": null,
            "paymentAmount": null
        }
    ],
    "transactionTenderTbl": [
        {
            "recType": null,
            "tendTypeGroup": null,
            "tendTypeId": null,
            "tenderAmount": null,
            "ccNo": null,
            "ccAuthNo": null,
            "ccAuthSource": null,
            "ccCardholderVerification": null,
            "ccExpirationDate": null,
            "ccEntryMode": null,
            "ccTerminalId": null,
            "ccSpecialCondition": null,
            "ccToken": null,
            "voucherNo": null,
            "couponNbr": null,
            "couponRefNo": null,
            "chequeAccNo": null,
            "chequeNo": null,
            "identificationMethod": null,
            "identificationId": null,
            "originalCurrency": null,
            "originalCurrencyAmount": null,
            "refNbr9": null,
            "refNbr1null": null,
            "refNbr11": null,
            "refNbr12": null,
            "transactionTenderAttributeTbl": [
                {
                    "recType": null,
                    "attributeType": null,
                    "attributeValue": null
                }
            ]
        }
    ]
}

```

```
        }
      ],
      "transactionTrailerTbl": [
        {
          "recType": null,
          "recCounter": null
        }
      ]
    }
  ],
  "salesFtailTbl": [
    {
      "recType": null,
      "recCounter": null
    }
  ]
}
]
```

Output

Table 5-1348 SVCPROV_SALES_STATUS_REC

Parameter Name	Data Type
statusMsg	String
salesErrTbl	List< SVCPROV_SALES_FAIL_REC >

Table 5-1349 SVCPROV_SALES_FAIL_REC

Parameter Name	Data Type
store	BigDecimal
transactionNo	BigDecimal
recType	String
businessDate	Date
errorMsg	String

The output will contain the status of the request including validation errors, if any.

JSON Structure:

```
{
  "statusMsg": null,
  "salesErrTbl": [
    {
      "store": null,
      "transactionNo": null,
      "recType": null,
      "businessDateDisplay": null,
      "businessDate": null,
      "errorMsg": null,
      "links": [],

```

```
        "hyperMediaContent": {  
          "linkRDO": []  
        }  
      },  
      "links": [],  
      "hyperMediaContent": {  
        "linkRDO": []  
      }  
    }  
  }  
}
```

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SVC_RTLOG_DATA_LOAD	No	Yes	No	No
ITEM_LOC_SOH	Yes	No	Yes	No
SA_INPROGRESS_SALES	Yes	Yes	Yes	No
SVC_INPROGRESS_SALES	No	Yes	No	No
SVC_PROCESS_TRACKER	No	Yes	No	No

6

Scheduled Integration

This chapter provides a summary of integrations that are scheduled either to be run once per day or periodically throughout the day. There are generally two types of integrations - those that expect or produce files and those that move data between integration tables, also referred to as Bulk Data Integration (BDI).

File-based Integration Overview

Merchandising and Sales Audit require that all inbound and outbound file-based transactions adhere to standard file layouts. There are two types of file layouts: detail-only and master-detail, which are described in the sections below.

Standard File Layouts

The Merchandising interface library supports two standard file layouts: one for master/detail processing, and one for processing detail records only. True sub-details are not supported within the Merchandising base package interface library functions.

A 5-character identification code or record type identifies all records within an I/O file, regardless of file type. The following includes common record type values:

- FHEAD-File Header
- FDETL-File Detail
- FTAIL-File Tail
- THEAD-Transaction Header
- TDETL-Transaction Detail
- TTAIL-Transaction Tail

Each line of the file must begin with the record type code followed by a 10-character record ID.

Detail-Only Files

File layouts have a standard file header record, a detail record for each transaction to be processed, and a file trailer record. Valid record types are FHEAD, FDETL, and FTAIL.

Example 6-1 Detail-Only Files

```
FHEAD0000000001STKU1996010100000019960929
FDETL0000000002SKU100000040000011011
FDETL0000000003SKU100000050003002001
FDETL0000000004SKU100000050003002001
FTAIL00000000050000000003
```

Master and Detail Files

File layouts consists of:

- Standard file header record

- Set of records for each transaction to be processed
- File trailer record.

The transaction set consists of:

- Transaction set header record
- Transaction set detail for detail within the transaction
- Transaction trailer record

Valid record types are FHEAD, THEAD, TDETL, TTAIL, and FTAIL.

Example 6-2 Master and Detail Files

```
FHEAD0000000001RTV 19960908172000
THEAD00000000020000000000000119960909120200000000003R
TDETL000000000300000000000001000001SKU10000012
TTAIL0000000004000001
THEAD000000000500000000000002199609091202001215720131R
TDETL000000000600000000000002000001UPC400100002667
TDETL0000000007000000000000020000021UPC400100002643 0
TTAIL0000000008000002
FTAIL00000000090000000007
```

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type.
	File Line Identifier	Number(10)	Specified by external system	Line number of the current file.
	File Type Definition	Char(4)	N/A	Identifies transaction type.
	File Create Date	Date	Create date	Date file was written by external system.
Transaction Header	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type.
	File Line Identifier	Number(10)	Specified by external system	Line number of the current file.
	Transaction Set Control Number	Char(14)	Specified by external system	Used to force unique transaction check.
	Detail Sequence Number	Char(6)	Specified by external system	Sequential number assigned to detail records within a transaction.
Transaction Detail	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type.
	File Line Identifier	Number(10)	Specified by external system	Line number of the current file.
	Transaction Set Control Number	Char(14)	Specified by external system	Used to force unique transaction check.

Record Name	Field Name	Field Type	Default Value	Description
	Detail Sequence Number	Char(6)	Specified by external system	Sequential number assigned to detail records within a transaction.
Transaction Trailer	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type.
	File Line Identifier	Number(10)	Specified by external system	Line number of the current file.
	Transaction Detail Line Count	Number(6)	Sum of detail lines	Sum of the detail lines within a transaction.
File Trailer	File Record Type Descriptor	Char(5)	FTAIL	Identifies file record type.
	File Line Identifier	Number(10)	Specified by external system	Line number of the current file.
	Total Transaction Line Count	Number(10)	Sum of all transaction lines	All lines in file less the file header and trailer records.

Integration Batch Wrapper FAQ

See also the "Batch Wrapper Overview" in Chapter 1 of the *Merchandising Operations Guide, Volume 1*.

For input files, are there any specific file format names?

Usually, the input file pattern is the same as the batch name. This can be seen in the ParameterValue column of the Job tab in the Batch Schedule spreadsheet. For example, dealupld has this parameter value:

```
dealupld #SysOpt.dbwallet dealupld
```

The input file pattern is the 3rd parameter (dealupld) so the filename should have dealupld in it (for example, dealupld_input, input_dealupld, and so on). There are some batch programs that should start with a specific pattern like saimptlogi - the input file should start with RTLOG*. This can be seen as well in the Batch Schedule spreadsheet:

```
#SysOpt.dbwallet RTLOG
```

How and where will files be moved?

The input file (zipped) should be in uploaded to the Object Storage through the FTS Rest service. The files should have a prefix of incoming/.

Files uploaded to FTS Rest Service are no longer required to have a .complete file.

The batch wrappers will handle downloading the files from Object Storage using FTS Rest Service with "incoming/" prefix and the corresponding file name pattern at each batch runtime. The files will be downloaded to the batch/incoming folder. Each file downloaded from Object Storage will again be uploaded with a "downloaded/" prefix. The files with "incoming/" prefix will then be deleted from Object Storage.

The batch wrappers will move the downloaded files to the input folder (data/in) given that the filename of the zip file corresponds to the expected file name pattern.

When will files be unzipped?

This will be unzipped when it is moved to the input folder (data/in).

What will happen to the files once they are processed?

Successfully processed files will be move to the processed folder (data/processed). Files with errors will be uploaded to Object Storage with a "reject/" prefix and will be removed from the input folder (data/in) and corresponding errors will be logged in the error log.

Will processed file be archived?

Yes. Input files will be archived in the processed folder (data/processed), while any output file will be archived in the archive folder (data/archive)

What will happen if the file is rejected?

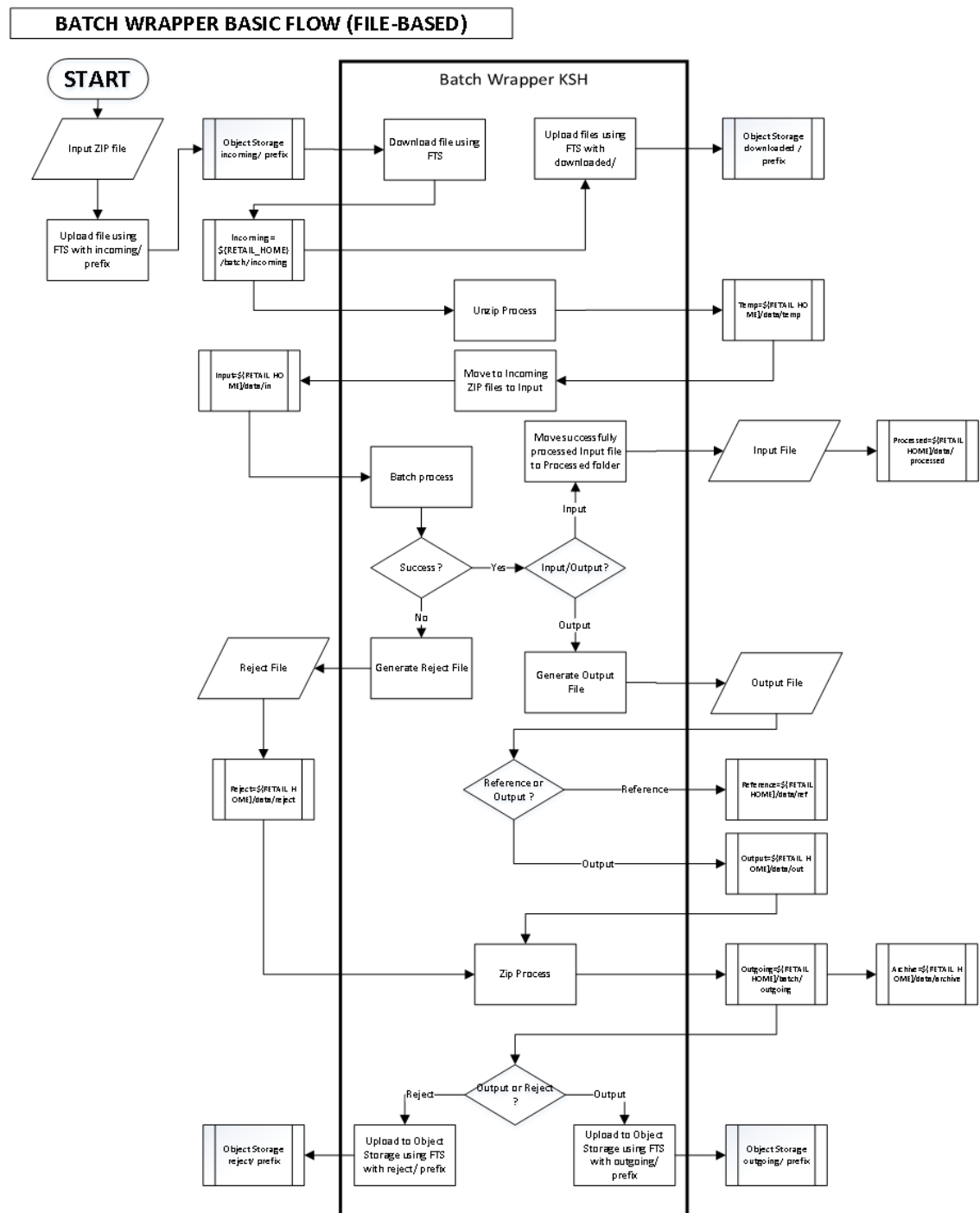
Reject files will be created as applicable and this will be sent to the outgoing folder (batch/outgoing) as well as archived in the archive folder (data/archive). Reject files in the batch/outgoing folder will be uploaded to Object Storage with a "reject/" prefix and deleted from the batch/outgoing folder.

Depending on the batch, if the program completes successfully even with rejected records then the rejected input file will be moved to the processed folder (data/processed). But if the batch aborts, then the rejected input file will be removed from the input folder (data/in) and uploaded to Object Storage with a "reject/" prefix.

How will rejected files be reprocessed?

Reprocessing the files would require manual intervention. Upload a corrected file to Object Storage with an "incoming/" prefix. Ensure the filename is the same as the rejected file.

File Movement Flow



Input/Output File Naming Convention

Batch	Input File Name	Zip File Name
dealupld	*dealupld*	*dealupld*.zip
ediupack	*ediupack*	*ediupack*.zip
ediupavl	*ediupavl*	*ediupavl*.zip
lcup798	*lcup798*	*lcup798*.zip
lcupld	*lcupld*	*lcupld*.zip
cmpupld	*cmpupld*	*cmpupld*.zip
fcosttmplupld.ksh	*fcosttmplupld*	*fcosttmplupld*.zip
htsupld	*htsupld*	*htsupld*.zip
otbupld	*otbupld*	*otbupld*.zip
tranupld	*tranupld*	*tranupld*.zip
fcustomerupload.ksh	*fcustomerupload*	*fcustomerupload*.zip
iiindbatch.ksh	Input file name is provided by user. Must end in *.xml Template name must correspond to S9T_TEMPLATE.TEMPLATE_KEY	*<input file name>*.zip
poindbatch.ksh	Input file name is provided by user. Must end in *.xml Template name must correspond to S9T_TEMPLATE.TEMPLATE_KEY	*<input file name>*.zip
replindbatch.ksh	Input file name is provided by user. Must end in *.xml Template name must correspond to S9T_TEMPLATE.TEMPLATE_KEY	*<input file name>*.zip
saimpadj	*saimpadj*	*saimpadj*.zip
load_item_forecast.ksh	*demand*	*demand*.zip
lcmt700	*lcmt700*	*lcmt700*.zip
lcmt707	*lcmt707*	*lcmt707*.zip
resa2sim	SIMT*	No zip file. Input file is from saexpsim.
resa2dw	RDWT* RDWF* RDWS* RDWC*	No zip file. Input file is from saexpdw.
lcmt730	*lcmt730*	*lcmt730*.zip
lcmt798	*lcmt798*	*lcmt798*.zip
lifstkup	*lifstkup*	*lifstkup*.zip
ordinvupld	ORIN*	ORIN*.zip
sa_rules_total_upload.ksh	sartexp_<table name>*	sartexp*.zip
saimptlogfin	storedayfile	No zip file. Input file is from sagetref.
saimptlogi	RTLOG_<store>_<datetime>.dat	RTLOG*.zip

Batch	Input File Name	Zip File Name
saimptlogtdup_upd	storedayfile storeposfile	No zip file. Input file is from sagetref.
savouch	*savouch*	*savouch*.zip
stlgdnld	*stlgdnld*	*stlgdnld*.zip
stockcountupload.ksh	STK*.<file extension>	STK*.zip
trandataload.ksh	*trandatoload*	*trandatoload*.zip
uploadsales.ksh	POSU_<store>_<tran_date>_ <sysdate>.<thread_val>	POSU*.zip
wfslsupld.ksh	*wfslsupld*	*wfslsupld*.zip
wfordupld.ksh	wford*.dat	wford*.zip
wfretupld.ksh	wfreturn*.dat	wfreturn*.zip

Bulk Data Integration Overview

Oracle Bulk Data Integration (BDI) is a solution that is part of the Oracle Retail Integration Cloud Service that defines the architecture and infrastructure used to move bulk data among Oracle Retail applications.

In a Bulk Data Integration system, message families are represented as interface modules. Each interface module (for example, DiffGrp_Fnd) contains a Merchandising component that takes care of pulling and staging data for publication to the External BDI system. Interface modules are divided by functional entity (for example, Item Master, Stores, Diffs, and so on).

For more information on BDI, see the Oracle Retail Integration Cloud Service documentation on docs.oracle.com/retail/.

Outbound Scheduled Integration

This section provides a summary of integrations that are scheduled either to be run once per day or periodically throughout the day to send data from Merchandising or Sales Audit to another solution. It includes both file-based and BDI-based integrations.

Foundation Data

Merchandising publishes foundation data for many other solution areas, including stores, warehouses, omni-channel, and so on.

The following scheduled outbound integrations are included in this functional area:

- [Brand Publication API \(BDI_Brand_Fnd_PF_From_RMS_EOW_JOB\)](#)
- [Calendar Publication API \(BDI_Calendar_Fnd_PF_From_RMS_EOW_JOB\)](#)
- [Code Detail Publication API \(BDI_CodeDetail_Fnd_PF_From_RMS_JOB\)](#)
- [Code Head Publication API \(BDI_CodeHead_Fnd_PF_From_RMS_JOB\)](#)
- [Company-wide Closings and Company Closed Exceptions \(BDI_CompanyClosed_Fnd_PF_From_RMS_JOB\)](#)
- [Currency Conversion Rates Publication API \(BDI_CurrConvRates_Fnd_PF_From_RMS_EOW_JOB\)](#)

- [Delivery Slot Publication API \(BDI_DeliverySlot_Fnd_PF_From_RMS_JOB\)](#)
- [Diff Group Export \(export_diffgrp.ksh\)](#)
- [Diff Group Publication API \(BDI_DiffGrp_Fnd_PF_From_RMS_JOB\)](#)
- [Diff ID and Type Export \(export_diffs.ksh\)](#)
- [Diff ID Publication API \(BDI_Diff_Fnd_PF_From_RMS_JOB\)](#)
- [Finisher Address Publication API \(BDI_FinisherAddr_Fnd_PF_From_RMS_JOB\)](#)
- [Location Closed Publication API \(BDI_LocClosed_Fnd_PF_From_RMS_JOB\)](#)
- [Merch Hierarchy Publication API \(BDI_MerchHier_Fnd_PF_From_RMS_JOB\)](#)
- [Merchandise Hierarchy Export \(export_merchhier.ksh\)](#)
- [Organization Hierarchy Publication API \(BDI_OrgHier_Fnd_PF_From_RMS_JOB\)](#)
- [Organizational Hierarchy Export \(export_orghier.ksh\)](#)
- [Partner Address Publication API \(BDI_PartnerAddr_Fnd_PF_From_RMS_JOB\)](#)
- [Partner Org Unit Publication API \(BDI_PartOrgUnit_Fnd_PF_From_RMS_JOB\)](#)
- [Partner Publication API \(BDI_Partner_Fnd_PF_From_RMS_JOB\)](#)
- [Store Address Publication API \(BDI_StoreAddr_Fnd_PF_From_RMS_JOB\)](#)
- [Store Hours Publication API \(BDI_StoreHours_Fnd_PF_From_RMS_JOB\)](#)
- [Store Publication API \(BDI_Store_Fnd_PF_From_RMS_JOB\)](#)
- [Stores Export \(export_stores.ksh\)](#)
- [Supplier Address Publication API \(BDI_SupplierAddr_Fnd_PF_From_RMS_JOB\)](#)
- [Sups Publication API \(BDI_Supplier_Fnd_PF_From_RMS_JOB\)](#)
- [Tax Download - Brazil \(taxdnld\)](#)
- [Ticket Download \(tcktdnld\)](#)
- [UDA Publication API \(BDI_Uda_Fnd_PF_From_RMS_JOB\)](#)
- [UDA Values Publication API \(BDI_UdaValues_Fnd_PF_From_RMS_JOB\)](#)
- [UOM Class Publication API \(BDI_UomClass_Fnd_PF_From_RMS_JOB\)](#)
- [UOM Conversion Publication API \(BDI_UomConversion_Fnd_PF_From_RMS_JOB\)](#)
- [VAT Codes, Regions, and Rates Export \(export_vat.ksh\)](#)
- [VAT Publication API \(BDI_Vat_Fnd_PF_From_RMS_JOB\)](#)
- [Warehouse \(BDI_Wh_Fnd_PF_From_RMS_JOB\)](#)
- [Warehouse Address Publication API \(BDI_WhAddr_Fnd_PF_From_RMS_JOB\)](#)

Brand Publication API (BDI_Brand_Fnd_PF_From_RMS_EOW_JOB)

This section describes the Brand Publication BDI.

Functional Area

Foundation Data

Business Overview

This API publishes the creation, update and deletion of brand information to the RIB such that all the downstream applications (including an external system) may subscribe to it and have information in sync with Merchandising. Publishing is provided synchronously in a near-real time manner.

New Brand

Creating a new brand triggers a message sent to notify external systems. Brand name and description are sent as part of the create message.

Table 6-1 Brand – Create and Update

Message Element	Included?	Notes
Brand name	Always	This field contains the brand name.
Brand Description	Always	This field contains the description of the brand.

Updated Brand

Updating a brand triggers a message to be sent to notify external systems. Brand name and description are sent as part of the update message.

Deleted Brand

Deleting a brand triggers a message sent to notify external systems. The brand name is sent as part of the delete message.

Table 6-2 Brand – Delete

Message Element	Included?	Notes
Brand name	Always	This field contains the brand name being deleted.

Error Handling

When the publication encounters a non-fatal error, messages continue to be processed. For the message where the error was encountered, a status of Hospital (H) is sent to the RIB and the status of the message in the queue is set to H. These types of errors occur when no changes in the database have been made and a process to try to re-publish the messages is available. In case the error is a fatal error, a status of Error (E) is sent to the RIB and the next message in the queue is not retrieved until the error is resolved.

Message XSD

Here are the filenames that correspond with each message type. Please consult the RIB documentation for each message type for a detailed picture of the composition of each message.

Message Types	Message Type Description	XML Schema Definition (XSD)
brandcre	Brand Create Message	BrandDesc.xsd

Message Types	Message Type Description	XML Schema Definition (XSD)
brandupd	Brand Modify Message	BrandDesc.xsd
branddel	Brand Delete Message	BrandRef.xsd

Calendar Publication API (BDI_Calendar_Fnd_PF_From_RMS_EOW_JOB)

This section describes the Calendar Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Calendar information (2 prior years, current year, 2 future years) from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdifoundationb.pls.pls

```
BDI_FOUNDATION_SQL.CALENDAR_UP(0_error_message IN OUT  
RTK_ERRORS.RTK_TEXT%TYPE,                                O_control_id   IN OUT  
NUMBER,                                                    I_job_context  IN   VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising V_BDI_DAY_LEVEL_CALENDAR view.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Calendar	Calendar upload to BDI	Calendar_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CALENDAR_OUT	No	Yes	No	No
V_BDI_DAY_LEVEL_CALENDAR	Yes	No	No	No

Code Detail Publication API (BDI_CodeDetail_Fnd_PF_From_RMS_JOB)

This section describes the Code Detail Publication BDI.

Functional Area

Cross Pillar

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Detail information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdicrosspillarb.pls

```
BDI_CROSS_PILLAR_SQL.CODE_DETAIL_UP(
    O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id    IN OUT NUMBER,
    I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function updates the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising CODE_DETAIL table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This updates the internal BDI control tables.

A database commit is issued, and the control ID is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Code Detail	Code Detail upload to BDI	CodeDetail_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_DETAIL_OUT	No	Yes	No	No
CODE_DETAIL	Yes	No	No	No

Code Head Publication API (BDI_CodeHead_Fnd_PF_From_RMS_JOB)

This section describes the Code Head Publication BDI.

Functional Area

Cross Pillar

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdicrosspillarb.pls

```
BDI_CROSS_PILLAR_SQL.CODE_HEAD_UP(O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                                   O_control_id    IN OUT NUMBER,
                                   I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising CODE_HEAD table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Code Head	Code Head upload to BDI	CodeHead_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CODE_HEAD_OUT	No	Yes	No	No
CODE_HEAD	Yes	No	No	No

Company-wide Closings and Company Closed Exceptions (BDI_CompanyClosed_Fnd_PF_From_RMS_JOB)

This section describes the Company-wide Closings and Company Closed Exceptions Publication BDI.

Functional Area

Foundation

Design Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Store information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

The following packages are impacted:

Filename: bdifoundations.pls

```
BDI_FOUNDATION_SQL.COMPANY_CLOSED_UP (
    O_error_message  IN OUT  RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id     IN OUT  NUMBER,
    I_job_context    IN      VARCHAR2)
```

Filename: bdifoundationb.pls

```
BDI_FOUNDATION_SQL.COMPANY_CLOSED_UP (
    O_error_message  IN OUT  RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id     IN OUT  NUMBER,
    I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising company closed and company closed exception table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Company Closed	Company Closed upload to BDI	CompanyClosed_Fnd_BdiInterfaceModule.xml
Company Closed Exceptions	Company Closed Exceptions upload to BDI	CompanyClosedExcep_Fnd_BdiInterfaceModule.xml

Tables

TABLE	SELECT	INSERT	UPDATE	DELETE
COMPANY_CLOSED_OUT	No	Yes	No	No
COMPANY_CLOSED_EXCEP_OUT	No	Yes	No	No
COMPANY_CLOSED_ECXEP	Yes	No	No	No
COMPANY_CLOSED	Yes	No	No	No

Currency Conversion Rates Publication API (BDI_CurrConvRates_Fnd_PF_From_RMS_EOW_JOB)

This section describes the Currency Conversion Rates Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Currency conversion rates information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: `bdifoundationb.pls.pls`

```
BDI_FOUNDATION_SQL.CURR_CONV_RATES_UP(
    O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id    IN OUT NUMBER,
    I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising MV_CURRENCY_CONVERSION_RATES materialized view.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Currency Conversion Rates	Currency Conversion Rates upload to BDI	CurrConvRates_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
CURR_CONV_RATES_OUT	No	Yes	No	No
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
STORE	Yes	No	No	No
WH	Yes	No	No	No

Delivery Slot Publication API (BDI_DeliverySlot_Fnd_PF_From_RMS_JOB)

This section describes the Delivery Slot Publication BDI.

Functional Area

Cross Pillar

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Delivery Slot information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdicrosspillarb.pls

```
BDI_CROSS_PILLAR_SQL.DELIVERY_SLOT_UP (
    O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id    IN OUT NUMBER,
    I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising DELIVERY_SLOT table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Delivery Slot	Delivery Slot upload to BDI	DeliverySlot_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
DELIVERY_SLOT_OUT	No	Yes	No	No
DELIVERY_SLOT	Yes	No	No	No

Diff Group Export (export_diffgrp.ksh)

Module Name	export_diffgrp.ksh
Description	Extraction of differentiator groups data.
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS255
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising diff group information into a flat file. Data to be extracted will be pulled off from the differentiator group tables. The mode (full vs. delta) will be an input parameter for this batch. The mode will allow a full extract (all diff group records in Merchandising) as well as delta processing (all diff group record changes in the time frame passed in the program) of data. For a full extract, records will be retrieved from the differentiator group tables. For a delta extract, the action type and diff group ID will be retrieved from the differentiator group staging export table and the attributes will be retrieved from the differentiator group tables.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	diffgrphdr_date_[full/delta]_[#ofLines].dat diffgrpdtl_date_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000212.html IntCon000213.html

Design Assumptions

N/A

Diff Group Publication API (BDI_DiffGrp_Fnd_PF_From_RMS_JOB)

This section describes the Diff Group Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Diff Groups from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: **bdicrosspillarb.pls**

```
BDI_CROSS_PILLAR_SQL.DIFF_GROUP_UP(0_error_message IN OUT VARCHAR2,
                                     0_control_id    IN OUT NUMBER,
                                     I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound tables that reside in the BDI_RMS_INT_SCHEMA schema. These outbound tables are loaded with records from the Merchandising Diff Group head and detail tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Diff Group	Diff Group upload to BDI	DiffGrp_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
DIFF_GRP_OUT	No	Yes	No	No
DIFF_GRP_DTL_OUT	No	Yes	No	No
DIFF_GROUP_HEAD	Yes	No	No	No
DIFF_TYPE	Yes	No	No	No
DIFF_GROUP_DETAIL	Yes	No	No	No

Diff ID and Type Export (export_diffs.ksh)

Module Name	export_diffs.ksh
Description	Extraction of differentiator's data defined for a differentiator type.
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh
Catalog ID	256
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising differentiator information into a flat file. Data to be extracted will be pulled off from the differentiator extract staging and the differentiator IDs table.

The mode (full vs. delta) will be an input parameter for this batch. The mode will allow a full extract (all differentiator records in Merchandising) as well as delta processing (all differentiator record changes in the time frame passed in the program) of data.

For a full extract, records will be solely retrieved from the differentiator IDs table. For a delta extract, the action type and differentiator ID will be retrieved from the differentiator export staging table and the attributes will be retrieved from the differentiator IDs table.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	diffs_date_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000206.html

Design Assumptions

N/A

Diff ID Publication API (BDI_Diff_Fnd_PF_From_RMS_JOB)

This section describes the Diff ID Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Diff IDs from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: **bdicrosspillarb.pls**

```
BDI_CROSS_PILLAR_SQL.DIFF_UP(0_error_message IN OUT VARCHAR2,
                             0_control_id     IN OUT NUMBER,
                             I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Diff tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow Type	Description	XML Schema Definition (XSD)
Diff Id	Diff Id upload to BDI	Diff_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
DIFF_OUT	No	Yes	No	No
DIFF_IDS	Yes	No	No	No
DIFF_TYPE	Yes	No	No	No

Finisher Address Publication API (BDI_FinisherAddr_Fnd_PF_From_RMS_JOB)

This section describes the Finisher Address Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Finisher Address positions from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package

Package Impact

Filename: `bdifoundations/b.pls`

```
BDI_FOUNDATION_SQL.FINISHER_ADDR_UP(0_error_message IN OUT VARCHAR2,
                                     0_control_id      IN OUT NUMBER,
                                     I_job_context      IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Finisher Address tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Finisher Address	Finisher Address upload to BDI	FinisherAddr_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
FINISHER_ADDR_OUT	No	Yes	No	No
ADD_TYPE_MODULE	Yes	No	No	No
WH	Yes	No	No	No
V_ADD_TYPE_TL	Yes	No	No	No
COUNTRY	Yes	No	No	No
STATE	Yes	No	No	No
ADDR	Yes	No	No	No
PARTNER	Yes	No	No	No

Location Closed Publication API (BDI_LocClosed_Fnd_PF_From_RMS_JOB)

This section describes the Location Closed Publication BDI.

Functional Area

Foundation

Design Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Store information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

The following packages are impacted by this BDI:

Bulk Interface Module

The following build interface module packages are impacted:

Filename: **bdifoundations.pls**

```
FUNCTION LOCATION_CLOSED_UP(0_error_message  IN OUT  RTK_ERRORS.RTK_TEXT%TYPE,
                           0_control_id      IN OUT  NUMBER,
                           I_job_context      IN     VARCHAR2)
```

Filename: **bdifoundationb.pls**

```
FUNCTION LOCATION_CLOSED_UP(0_error_message  IN OUT  RTK_ERRORS.RTK_TEXT%TYPE,
                           0_control_id      IN OUT  NUMBER,
                           I_job_context      IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Location closed table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Location Closed	Location Closed upload to BDI	LocationClosed_Fnd_BdiInterfaceModule.xml

Tables

TABLE	SELECT	INSERT	UPDATE	DELETE
LOCATION_CLOSED_OUT	No	Yes	No	No
LOCATION_CLOSED	Yes	No	No	No

Merch Hierarchy Publication API (BDI_MerchHier_Fnd_PF_From_RMS_JOB)

This section describes the Merch Hierarchy Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Merchandise Hierarchy information from Merchandising to other Oracle Retail Applications.

On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdimerchb.pls

BDI_MERCH_SQL.MERCH_HIER_UP(0_error_message	IN OUT	VARCHAR2,
0_control_id	IN OUT	NUMBER,
I_job_context	IN	VARCHAR2)

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Merchandise Hierarchy tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Merchandise Hierarchy	Merchandise Hierarchy upload to BDI	MerchHier_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
MERCH_HIER_OUT	No	Yes	No	No
DIVISION	Yes	No	No	No
COMPHEAD	Yes	No	No	No
GROUPS	Yes	No	No	No
DEPS	Yes	No	No	No
CLASS	Yes	No	No	No
SUBCLASS	Yes	No	No	No

Merchandise Hierarchy Export (export_merchhier.ksh)

Module Name	export_merchhier.ksh
Description	Extraction of merchandise hierarchy data.
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh
Catalog ID	260
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising merchandise hierarchy information from division to subclass into a flat file. Data to be extracted will be pulled off from

the merchandise hierarchy export staging table and the main merchandise hierarchy tables. The mode (full vs. delta) will be an input parameter for this new batch. The mode will allow a full extract (all merchandise hierarchy records in Merchandising) as well as delta processing (all merchandise hierarchy changes since the last export) of data. For a full extract, records will be solely retrieved from the main merchandise hierarchy tables. For a delta extract, the action type and entity ID will be retrieved from the merchandise hierarchy export staging table and the attributes of the entities will be retrieved from their corresponding main entity tables.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	merchierarchy_[Date]_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000207

Design Assumptions

N/A

Organization Hierarchy Publication API (BDI_OrgHier_Fnd_PF_From_RMS_JOB)

This section describes Organization Hierarchy Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Org Hierarchy information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiorgb.pls

```
BDI_ORG_SQL.ORG_HIER_UP(0_error_message IN OUT VARCHAR2,
                        0_control_id    IN OUT NUMBER,
                        I_job_context   IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Organization Hierarchy tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Org Hierarchy	Org Hierarchy upload to BDI	OrgHier_End_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ORG_HIER_OUT	No	Yes	No	No
WH	Yes	No	No	No
AREA	Yes	No	No	No
CHAIN	Yes	No	No	No
COMPHEAD	Yes	No	No	No
DISTRICT	Yes	No	No	No
REGION	Yes	No	No	No
STORE	Yes	No	No	No
WH	Yes	No	No	No

Organizational Hierarchy Export (export_orghier.ksh)

Module Name	export_orghier.ksh
Description	Extraction of organizational hierarchy data.
Functional Area	Foundation
Module Type	Integration
Module Technology	Ksh
Catalog ID	RMS261
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising organizational hierarchy information from company to stores and warehouses into a flat file. Data to be extracted will be pulled off from the organizational hierarchy export staging table and the main organizational

hierarchy tables. The mode (full vs. delta) will be an input parameter for this batch. The mode will allow a full extract (all organizational hierarchy records in Merchandising) as well as delta processing (all organizational hierarchy changes since the last export) of data. For a full extract, records will be solely retrieved from the main organizational hierarchy tables. For a delta extract, the action type and entity ID will be retrieved from the organizational hierarchy export staging table and the attributes of the entities will be retrieved from their corresponding main entity tables.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	orghierarchy_[Date]_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000203

Design Assumptions

N/A

Partner Address Publication API (BDI_PartnerAddr_Fnd_PF_From_RMS_JOB)

This section describes the Partner Address Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdifoundationb.pls.pls

```
BDI_FOUNDATION_SQL.PARTNER_ADDR_UP(
                                O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                                O_control_id     IN OUT NUMBER,
                                I_job_context     IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Partner Address table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Partner Address	Partner Address upload to BDI	PartnerAddr_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PARTNER_ADDR_OUT	No	Yes	No	No
ADDR	Yes	No	No	No
V_ADD_TYPE_TL	Yes	No	No	No
ADD_TYPE_MODULE	Yes	No	No	No
STATE	Yes	No	No	No
COUNTRY	Yes	No	No	No
PARTNER	Yes	No	No	No

Partner Org Unit Publication API (BDI_PartOrgUnit_Fnd_PF_From_RMS_JOB)

This section describes the Partner Org Unit Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: **bdifoundationb.pls.pls**

```
BDI_FOUNDATION_SQL.PARTNER_ORG_UNIT_UP(
    O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id    IN OUT NUMBER,
    I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Partner Org Unit table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Partner Org Unit	Partner Org Unit upload to BDI	PartnerOrgUnit_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PARTNER_ORG_UNIT_OUT	No	Yes	No	No
PARTNER_ORG_UNIT	Yes	No	No	No

Partner Publication API (BDI_Partner_Fnd_PF_From_RMS_JOB)

This section describes the Partner Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: **bdifoundationb.pls.pls**

```
BDI_FOUNDATION_SQL.PARTNER_UP(O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                                O_control_id    IN OUT NUMBER,
                                I_job_context   IN   VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API. A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Partner table. After the insert, another call to a BDI function is performed to

signify the successful loading of records. This will update the internal BDI control tables. A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Partner	Partner upload to BDI	Partner_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PARTNER_OUT	No	Yes	No	No
PARTNER	Yes	No	No	No

Store Address Publication API (BDI_StoreAddr_Fnd_PF_From_RMS_JOB)

This section describes the Store Address Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Store Address information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiorgb.pls

```
BDI_ORG_SQL.STORE_ADDR_UP(0_error_message IN OUT VARCHAR2,
                           0_control_id    IN OUT NUMBER,
                           I_job_context   IN   VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Store Address table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Store Addr	Store Address upload to BDI	StoreAddr_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
STORE_ADDR_OUT	No	Yes	No	No
V_ADD_TYPE_TL	Yes	No	No	No
ADDR	Yes	No	No	No
STORE	Yes	No	No	No
STATE	Yes	No	No	No
COUNTRY	Yes	No	No	No
ADD_TYPE_MODULE	Yes	No	No	No

Store Hours Publication API (BDI_StoreHours_Fnd_PF_From_RMS_JOB)

This section describe the Store Hours Publication BDI.

Function Area

Foundation

Design Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Store information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

The following packages are impacted by the Store Hours Publication BDI:

Bulk Interface Module

In the Build Interface Module:

Filename: bdiorgb.pls

```
BDI_ORG_SQL.STORE_HOURS_UP(O_error_message IN OUT VARCHAR2,
                           O_control_id     IN OUT NUMBER,
                           I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Store table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Store	Store upload to BDI	StoreHours_Fnd_BdiInterfaceModule.xml

Tables

TABLE	SELECT	INSERT	UPDATE	DELETE
STORE_HOURS_OUT	No	Yes	No	No
STORE_HOURS	Yes	No	No	No

Store Publication API (BDI_Store_Fnd_PF_From_RMS_JOB)

This section describes the Store Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Store information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiorgb.pls

```
BDI_ORG_SQL.STORE_UP(0_error_message IN OUT VARCHAR2,
                    0_control_id IN OUT NUMBER,
                    I_job_context IN VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Item Location table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Store	Store upload to BDI	Store_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
STORE_OUT	No	Yes	No	No
CODE_DETAIL	Yes	No	No	No
CHANNELS	Yes	No	No	No
STORE_FORMAT	Yes	No	No	No
LANG	Yes	No	No	No
VAT_REGION	Yes	No	No	No
TSFZONE	Yes	No	No	No

Stores Export (export_stores.ksh)

Module Name	export_stores.ksh
Description	Extraction of store data.
Functional Area	Foundation
Module Type	Integration
Module Technology	Ksh
Catalog ID	RMS263
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising store information into two flat files - one for store and one for store addresses. Data to be extracted will be pulled from the store export staging, store and address tables. The mode (full vs. delta) will be an input

parameter for this batch. The mode will allow a full extract (all store records in Merchandising) as well as delta processing (all store changes since the last export) of data. For a full extract, records will be solely retrieved from the store table for store information and address table for store addresses. For a delta extract, the action type, store ID and address will be retrieved from the store export staging table and the details of the store will be retrieved from both the store and address tables.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	store_[Date]_[full/delta]_[#ofLines].datstoreaddr_[Date]_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000204 IntCon000205

Design Assumptions

N/A

Supplier Address Publication API (BDI_SupplierAddr_Fnd_PF_From_RMS_JOB)

This section describes the Supplier Address Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Supplier Address positions from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdifoundations/b.pls

BDI_FOUNDATION_SQL.SUPPLIER_ADDR_UP(0_error_message	IN OUT	VARCHAR2,
0_control_id	IN OUT	NUMBER,
I_job_context	IN	VARCHAR2)

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Supplier Address tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Supplier Address	Supplier Address upload to BDI	SupplierAddr_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SUPPLIER_ADDR_OUT	No	Yes	No	No
ADDR	Yes	No	No	No
V_ADD_TYPE_TL	Yes	No	No	No
STATE	Yes	No	No	No
COUNTRY	Yes	No	No	No
SUPS	Yes	No	No	No
ADD_TYPE_MODULE	Yes	No	No	No

Sups Publication API (BDI_Supplier_Fnd_PF_From_RMS_JOB)

This section describes the Sups Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: `bdifoundationb.pls.pls`

```
BDI_FOUNDATION_SQL.SUPS_UP(O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                           O_control_id     IN OUT NUMBER,
                           I_job_context    IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Sups table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Supplier	Supplier upload to BDI	Supplier_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
SUPS_OUT	No	Yes	No	No
SUPS	Yes	No	No	No

Tax Download - Brazil (taxdnld)

Module Name	taxdnld
Description	Tax Download to 3rd Party POS in Global Tax [GTAX] Implementations
Functional Area	Integration - 3rd Party POS
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS124
Wrapper Script	N/A

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program downloads the tax information to 3rd Party POS systems when the Merchandising default tax type is GTAX. This program only needs to be run if the client uses Merchandising Global Tax functionality.

Restart/Recovery

The logical unit of work for this module is defined by item, ref_item and store combination. This batch program uses table-based restart/recovery. The commit happens in the database when the commit max counter is reached.

Integration Contract

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000020

Output File Layout

Table 6-3 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file
	File Type Definition	Char(4)	TAXD	Identifies file as 'Tax Details'
	File Create Date	Char(14)	create date	Vdate in 'YYMMDDHHMMSS'format
FDETL	FDETL	Char(5)	FDETL	FDETL
	File Line Sequence	Number(10)	N/A	Line number of the current file
	STORE	Char(10)	N/A	Store number
	ITEM	Char(25)	N/A	Item
	item_number_t ype	Char(6)	S - Store W - Warehouse	Item number type
	format_id	Char(1)	N/A	Format id
	prefix	Char(2)	N/A	Prefix
	ref_item	Char(25)	N/A	Reference Item
	ref_item_num ber_type	Char(6)	N/A	Reference item number type
	ref_format_id	Char(1)	N/A	Ref format id
	ref_prefix	Char(2)	N/A	Ref no. prefix
	taxable indicator	Char(1)	N/A	Taxable indicator
	class_vat_ind	Char(1)	N/A	Class vat indicator
FTAXD	FTAXD	Char(5)	FTAXD	FTAXD
	File Line Sequence	Number(10)	N/A	Line number of the current file
	tax_code	Char(10)	N/A	Tax code
	tax_rate	Char(20)	N/A	Tax rate
	calculation_bas is	Char(1)	N/A	Calculation basis

Table 6-3 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	tax_amount	Char(20)	N/A	Tax amount
	effective_from	Char(8)	N/A	Effective from
	time	Char(6)	N/A	Time
	status	Char(1)	N/A	Status
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file
	rec_counter	Number(10)	N/A	Record counter

Design Assumptions

N/A

Ticket Download (tcktdnld)

Module Name	tcktdnld.pc
Description	Download of Data to be Printed on Tickets
Functional Area	Foundation Data
Module Type	Integration
Module Technology	PROC
Catalog ID	RMS59
Wrapper Script	rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program creates an output file containing the information to be printed on a ticket or label for a particular item/location. This program is driven by the requests for tickets generated from Merchandising and Pricing. The details of what should be printed on each ticket are defined in Merchandising on the ticket type details table.

Restart/Recovery

N/A

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameters

Integration Contract IntCon000107

Output File Layout

Table 6-4 tcktdnld.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file
	File Type Definition	Char(4)	TCKT	Identifies file as 'Print Ticket Requests'
	File Create Date	Char(14)	N/A	The date on which the file was created in 'YYYYMMDDHHMISS' format
THEAD	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file
	ITEM	Char(25)	N/A	ID number of the transaction level item for which the ticket applies.
	Ticket Type	Char(4)	N/A	ID which indicates the ticket type to be printed
	Location Type	Char(1)	N/A	Identifies the type of location for which tickets will be printed. Valid values are store (S) and warehouse (W).
	Location	Char(10)	N/A	The ID of the store or warehouse for which tickets will be printed
	Quantity	Number(12,4)	N/A	The quantity of tickets to be printed; which includes 4 implied decimal places
TCOMP	File Type Record Descriptor	Char(5)	TCOMP	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file
	ITEM	Char(25)	N/A	ID number of the item which is only populated if the item in THEAD is a pack item
	Quantity	Number(12,4)	N/A	Quantity of the component item as a part of the pack; includes 4 implied decimal places
TDETL	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type

Table 6-4 (Cont.) tcktdnld.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	File Line Sequence	Number(10)	N/A	Line number of the current file
	Detail Sequence Number	Number(10)	N/A	Sequential number assigned to the detail records
	Ticket Item	Char(4)	N/A	ID indicating the detail to be printed on the ticket. If the attribute is a UDA, then this will contain the ID of the UDA. Otherwise, it is the code associated with the attribute in Merchandising (such as, CLSS = class)
	Attribute Description	Char(120)	N/A	Description of the attribute – either the UDA description or the Merchandising description for the attribute
	Value	Char(250)	N/A	Detail to be printed on the ticket (for example: Item number, Department Number, Item description)
TTAIL	Supplement	Char(120)	N/A	Supplemental description to the Value (for example: Department Name)
	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file
FTAIL	Transaction Detail Line Count	Number(6)	sum of detail lines	Sum of the detail lines within a transaction
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file

Design Assumptions

N/A

UDA Publication API (BDI_Uda_Fnd_PF_From_RMS_JOB)

This section describes the UDA Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdifoundationb.pls

```
BDI_FOUNDATION_SQL.UDA_UP(0_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,  
                           0_control_id    IN OUT NUMBER,  
                           I_job_context   IN    VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising UDA table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
UDA	UDA upload to BDI	Uda_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UDA_OUT	No	Yes	No	No
UDA	Yes	No	No	No

UDA Values Publication API (BDI_UdaValues_Fnd_PF_From_RMS_JOB)

This section describes the UDA Values Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: `bdifoundationb.pls.pls`

```
BDI_FOUNDATION_SQL.UDA_VALUES_UP(0_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                                0_control_id      IN OUT NUMBER,
                                I_job_context      IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising UDA Values table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
UDA Values	UDA Values upload to BDI	UdaValues_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UDA_VALUES_OUT	No	Yes	No	No
UDA_VALUES	Yes	No	No	No

UOM Class Publication API (BDI_UomClass_Fnd_PF_From_RMS_JOB)

This section describes the UOM Class Publication BDI.

Functional Area

Cross Pillar

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Uom Class information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdicrosspillarb.pls

```
BDI_CROSS_PILLAR_SQL.UOM_CLASS_UP (
    O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id    IN OUT NUMBER,
    I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising UOM_CLASS table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Uom Class	Uom Class upload to BDI	UomClass_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UOM_CLASS_OUT	No	Yes	No	No
UOM_CLASS	Yes	No	No	No

UOM Conversion Publication API (BDI_UomConversion_Fnd_PF_From_RMS_JOB)

This section describes the UOM Conversion BDI.

Functional Area

Cross Pillar

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Uom Conversion information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: `bdicrosspillarb.pls`

```
BDI_CROSS_PILLAR_SQL.UOM_CONVERSION_UP (
    O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id    IN OUT NUMBER,
    I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the `BDI_RMS_INT_SCHEMA` schema. This outbound table is loaded with records from the Merchandising `UOM_CONVERSION` table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Uom Conversion	Uom Conversion upload to BDI	UomConversion_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UOM_CONVERSION_OUT	No	Yes	No	No
UOM_CONVERSION	Yes	No	No	No

VAT Codes, Regions, and Rates Export (export_vat.ksh)

Module Name	export_vat.ksh
Description	Extraction of vat data
Functional Area	Foundation
Module Type	Integration
Module Technology	Ksh

Catalog ID	RMS264
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising VAT information into a flat file. Data to be extracted will be pulled off from the VAT export staging, VAT region, VAT codes, and VAT code rates tables.

The mode (full vs. delta) will be an input parameter for this batch. The mode will allow a full extract (all vat region/vat code/vat code rate combination records in RMS) as well as delta processing (all VAT record changes in the time frame passed in the program) of data.

In either of the mode exempt vat region won't get fetched in case of SVAT tax type.

For a full extract, records will be retrieved from the VAT region, VAT code, and VAT code rates tables. For a delta extract, the action type, vat region, vat code and active date will be retrieved from the VAT export staging table and the attributes will be retrieved from the main table.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	vat_date_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000215

Design Assumptions

N/A

VAT Publication API (BDI_Vat_Fnd_PF_From_RMS_JOB)

This section describes the VAT Publication BDI.

Functional Area

Foundation

Design Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Vat information from RMS to other Oracle Retail Applications. On this particular integration stream, the data flow is from RMS to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling an RMS owned API that will pull data from RMS and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Bulk Interface Module

Filename: `bdifoundationb.pls`

```

BDI_FOUNDATION_SQL.FUNCTION VAT_UP
    (O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
    O_control_id     IN OUT  NUMBER,
    I_job_context    IN      VARCHAR2)

```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the VAT_CODES, VAT_CODE_RATES and VAT_REGION tables from RMS.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Vat	Vat upload to BDI	Vat_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
VAT_OUT	No	Yes	No	No
VAT_CODES	Yes	No	No	No
VAT_CODE_RATES	Yes	No	No	No
VAT_REGION	Yes	No	No	No

Warehouse (BDI_Wh_Fnd_PF_From_RMS_JOB)

This section describes Warehouse Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Warehouse information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API

that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiorgb.pls

```
BDI_ORG_SQL.WH_UP(O_error_message IN OUT VARCHAR2,
                  O_control_id     IN OUT NUMBER,
                  I_job_context    IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Warehouse tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Warehouse	Warehouse upload to BDI	Wh_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
WH_OUT	No	Yes	No	No
WH	Yes	No	No	No

Warehouse Address Publication API (BDI_WhAddr_Fnd_PF_From_RMS_JOB)

This section describes Warehouse Address Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Warehouse Address information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API

that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: **bdiorgb.pls**

```
BDI_ORG_SQL.WH_ADDR_UP(O_error_message IN OUT VARCHAR2,
                        O_control_id    IN OUT NUMBER,
                        I_job_context   IN    VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Warehouse Address tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Warehouse Address	Warehouse Address upload to BDI	WhAddr_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
WH_ADDR_OUT	No	Yes	No	No
ADDR	Yes	No	No	No

Items

Merchandising publishes item data for many other solution areas, including stores, warehouses, omni-channel, and so on.

- [Item Image Publication API \(BDI_ItemImage_Fnd_PF_From_RMS_JOB\)](#)
- [Item Location Export \(export_itemloc.ksh\)](#)
- [Item Location Publication API \(BDI_ItemLoc_Fnd_PF_From_RMS_JOB\)](#)
- [Item Master Export \(export_itemmaster.ksh\)](#)
- [Item Master Publication API \(BDI_ItemHdr_Fnd_PF_From_RMS_JOB\)](#)

- [Item Supplier Country Dimensions Publication API \(BDI_ItSupCtryDim_Fnd_PF_From_RMS_JOB\)](#)
- [Item Supplier Country Publication API \(BDI_ItSupCtry_Fnd_PF_From_RMS_JOB\)](#)
- [Item Supplier Manufacturing Country Publication API \(BDI_ItSupManCtry_Fnd_PF_From_RMS_JOB\)](#)
- [Item Supplier Publication API \(BDI_ItemSupp_Fnd_PF_From_RMS_JOB\)](#)
- [Item Supplier UOM Publication API \(BDI_ItemSuppUom_Fnd_PF_From_RMS_JOB\)](#)
- [Item VAT Rates Export \(export_itemvat.ksh\)](#)
- [Pack Item Publication API \(BDI_PckitemBrkout_Fnd_PF_From_RMS_JOB\)](#)
- [POS Configuration Data to 3rd Party POS \(poscdnld\)](#)
- [Price History Publication API \(BDI_PriceHist_Fnd_PF_From_RMS_JOB\)](#)
- [Related Items Export \(export_relitem.ksh\)](#)
- [Related Item Publication API \(BDI_RelatedItem_Fnd_PF_From_RMS_JOB\)](#)
- [UDA Item Date Publication API \(BDI_UdaltemDate_Fnd_PF_From_RMS_JOB\)](#)
- [UDA Item FF Publication API \(BDI_UdaltemFF_Fnd_PF_From_RMS_JOB\)](#)
- [UDA Item LOV Publication API \(BDI_UdaltemLov_Fnd_From_RMS_JOB\)](#)
- [Capture Item Inventory Records \(maintain_ilseod_hist.ksh\)](#)
- [Maintain Item Cost Records \(maintain_isc_hist.ksh\)](#)

Item Image Publication API (BDI_ItemImage_Fnd_PF_From_RMS_JOB)

This section describes the Item Image Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item Image information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

BDI_ITEM_SQL.ITEM_IMAGE_UP	(0_error_message	IN OUT	VARCHAR2,
	0_control_id	IN OUT	NUMBER,
	I_job_context	IN	VARCHAR2)

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising ITEM_IMAGE table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Item Image	Item Image upload to BDI	ItemImage_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_IMAGE_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_IMAGE	Yes	No	No	No

Item Location Export (export_itemloc.ksh)

Module Name	export_itemloc.ksh
Description	Extraction of item location data.
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS257
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job extracts new, updated and deleted Merchandising item-location information into a flat file.

- This batch supports both a full and delta export of item-location data.
- A threading indicator parameter should be passed. Passing 'Y' means a thread number (1-20) will be passed in. Passing 'N' means no thread number will be passed in and the program will use a default thread number.
- An optional location parameter may be passed in for either modes. If this value is passed in, the batch will create a flat file for the location passed in. If it is not passed in, the batch will create flat files for all locations.

- This creates separate files per location (Store, Warehouse or External Finisher).
- This exports delta item header information for each applicable store location.
- This will export data only for approved, sellable items.
- This will export attributes from item/location and item/location traits.
- This should also include the item parent as its own record in the extract.
- The flat files that will be created will now be pipe delimited.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	itemloc_[#date]_[#loc_type]_[#location]_[full/delta]_[#ofLines].dat itemhdr_[#date]_[#store_id]_delta_[#ofLines].dat
Integration Contract	IntCon000209 IntCon000208

Design Assumptions

N/A

Item Location Publication API (BDI_ItemLoc_Fnd_PF_From_RMS_JOB)

This section describes the Item Location Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item Location information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_LOC_UP(0_error_message  IN OUT  VARCHAR2,
                        0_control_id       IN OUT  NUMBER,
                        I_job_context       IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Item Location table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Item Location	Item Location upload to BDI	ItemLoc_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_LOC_OUT	No	Yes	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_TRAITS	Yes	No	No	No
STORE	Yes	No	No	No
WH	Yes	No	No	No
PARTNER	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No

Item Master Export (export_itemmaster.ksh)

Module Name	export_itemmaster.ksh
Description	Extraction of item data
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS258
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising item master information into a flat file.

Data to be extracted will be pulled off from the item export information, item export staging and the main item tables.

- The mode (full vs. delta) will be an input parameter for this new batch. The mode will allow a full extract (all approved, sellable items in Merchandising) as well as delta processing (all approved, sellable item changes in the main item table since the last export) of data.
- A threading indicator parameter should be passed. Passing 'Y' means a thread number (1-20) will be passed in. Passing 'N' means no thread number will be passed in and the program will use a default thread number.
- In full mode, normal operation will produce both a corporate level file and files for all stores. An optional input parameter will also allow the program to produce a location level file for a specified store.
- In delta mode, the only option is to produce corporate level files. Item header files at the store level will be created in the export_itemloc.ksh for delta mode.
- The store specific file will also include UPC items. To determine which UPC Items to include, the store where the UPC's parent and/or grandparent item is ranged should be taken into consideration.
- The flat files that will be created will now be pipe delimited.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	itemhdr_[#date]_corp_[full/delta]_[#ofLines].dat itemhdr_[#date]_[location]_full_[#ofLines].dat
Integration Contract	IntCon000208

Design Assumptions

N/A

Item Master Publication API (BDI_ItemHdr_Fnd_PF_From_RMS_JOB)

This section describes the Item Master Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item Master information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI calls a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API is in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_MASTER_UP(O_error_message IN OUT VARCHAR2,
                             O_control_id    IN OUT NUMBER,
                             I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Item Master table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

Data Flow	Description	XML Schema Definition (XSD)
Item Master	Item Master upload to BDI	ItemHdr_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_HDR_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
CLASS	Yes	No	No	No
SUBCLASS	Yes	No	No	No
DIFF_GROUP_HEAD	Yes	No	No	No
DIFF_IDS	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No

Item Supplier Country Dimensions Publication API (BDI_ItSupCtryDim_Fnd_PF_From_RMS_JOB)

This section describes the Item Supplier Country Dim Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item Supplier Country Dim information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_SUP_CTY_DIM_UP (O_error_message IN OUT VARCHAR2,
                                O_control_id      IN OUT NUMBER,
                                I_job_context     IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising ITEM_SUPP_COUNTRY_DIM table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Item Supplier Country Dim	Item supplier country Dim upload to BDI	ItSupCtryDim_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUP_CTY_DIM_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY_DIM	Yes	No	No	No

Item Supplier Country Publication API (BDI_ItSupCtry_Fnd_PF_From_RMS_JOB)

This section describes the Item Supplier Country Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item supplier country information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_SUPP_COUNTRY_UP (O_error_message IN OUT VARCHAR2,
                                     O_control_id    IN OUT NUMBER,
                                     I_job_context    IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising ITEM_SUPP_COUNTRY table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Item Supplier Country	Item supplier country upload to BDI	ItSupCtry_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPP_COUNTRY_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No

Item Supplier Manufacturing Country Publication API (BDI_ItSupManCtry_Fnd_PF_From_RMS_JOB)

This section describes the Item Supplier Manufacturing Country Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item Supplier Manufacturing Country information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_SUP_MAN_CTY_UP (O_error_message IN OUT VARCHAR2,
                                   O_control_id      IN OUT NUMBER,
                                   I_job_context      IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising ITEM_SUPP_MANU_COUNTRY table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Item Supplier Manufacturing Country	Item supplier Manufacturing Country upload to BDI	ItSupManCtry_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUP_MAN_CTY_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_MANU_COUNTRY	Yes	No	No	No

Item Supplier Publication API (BDI_ItemSupp_Fnd_PF_From_RMS_JOB)

This section describes the Item Supplier Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item supplier information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_SUPPLIER_UP(O_error_message IN OUT VARCHAR2,
                                O_control_id    IN OUT NUMBER,
                                I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising ITEM_SUPPLIER table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Item supplier	Item Supplier upload to BDI	ItemSupplier_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPPLIER_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No

Item Supplier UOM Publication API (BDI_ItemSuppUom_Fnd_PF_From_RMS_JOB)

This section describes the Item Supplier UOM Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Item supplier UOM information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.ITEM_SUPP_UOM_UP(O_error_message IN OUT VARCHAR2,
                                O_control_id    IN OUT NUMBER,
                                I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising ITEM_SUPP_UOM table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Item supplier UOM	Item Supplier UOM upload to BDI	ItemSuppUom_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ITEM_SUPP_UOM_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_SUPP_UOM	Yes	No	No	No

Item VAT Rates Export (export_itemvat.ksh)

Module Name	export_itemvat.ksh
Description	Extraction of vat item data.
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh

Catalog ID	RMS259
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising item VAT information into a flat file.

- This batch supports both a full and delta export of item VAT data.
- A threading indicator parameter should be passed. Passing 'Y' means a thread number (1-20) will be passed in. Passing 'N' means no thread number will be passed in and the program will use a default thread number.
- In full mode, normal operation will produce both a corporate level file and files for all stores. An optional input parameter will also allow the program to produce a location level file for a specified store.
- In full mode for store specific file if store belong to such a vat region, which is exempt (In case of tax type SVAT), then files for that store won't get generated.
- In delta mode, this will produce both corporate level files and files for all stores the modified items are ranged to and the vat region the store is associated with.
- In delta mode for store specific file if store belong to such a vat region, which is exempt, then files for that store won't get generated.
- This will export data only for approved, sellable items.
- This will export item VAT information from the item export staging and item tables.
- This should also include the item parent as its own record in the extract.
- The flat files that will be created will now be pipe delimited.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	vatitem_[#date]_corp_[full/delta]_[#ofLines].dat vatitem_[#date]_[location]_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000214

Design Assumptions

N/A

Pack Item Publication API (BDI_PckitemBrkout_Fnd_PF_From_RMS_JOB)

This section describes the Pack Item Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Pack Item information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.PACK_ITEM_UP(O_error_message IN OUT VARCHAR2,
                           O_control_id    IN OUT NUMBER,
                           I_job_context   IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising PACKITEM table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Pack Item	Pack Item upload to BDI	PackItem_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PACK_ITEM_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
PACKITEM	Yes	No	No	No

Populate ITEM_LOC_SOH_EOD table for RDE Extract (maintain_ilseod_hist)

Module Name	maintain_ilseod_hist.ksh (ilseodhistsqls.pls/ ilseodhistsqlb.pls)

Description	Populate ITEM_LOC_S0H_E0D_HIST table for RDE Extract
Functional Area	Stock Ledger
Module Type	Business Processing
Module Technology	PLSQL
Catalog ID	
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program copies current data from ITEM_LOC_S0H_E0D into ITEM_LOC_S0H_E0D_HIST and drops old partitions based on defined retention days (SYSTEM_OPTIONS.ILS_E0D_HIST_RETENTION_DAYS). It inserts a row into the RMS_RDE_BATCH_STATUS table stating that the INVENTORY_SNAPSHOT staging is complete for the business date.

Restart/Recovery

In the case of a restart from failure, it rebuilds records on ITEM_LOC_S0H_E0D_HIST for the current business date.

Design Assumptions

N/A

Populate ITEM_SUPP_COUNTRY_HIST and ITEM_SUPP_COUNTRY_LOC_HIST table (maintain_isc_hist)

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program copies data from ITEM_SUPP_COUNTRY into ITEM_SUPP_COUNTRY_HIST and drops old partitions based on defined retention days (SYSTEM_OPTIONS.ISC_HIST_RETENTION_DAYS). It also populates ITEM_SUPP_COUNTRY_LOC_HIST based on ITEM_SUPP_COUNTRY_LOC. The data is as of the SCN stored in the FLASHBACK_SNAPSHOT_INFO table.

This accepts an optional parameter of I_business_date which will then be used to extract the SCN data from FLASHBACK_SNAPSHOT_INFO that will be the basis for the extract from ITEM_SUPP_COUNTRY and ITEM_SUPP_COUNTRY_LOC. The SCN stored at this table should still be accessible. If the I_business_date input parameter is not provided, it uses the current PERIOD.VDATE.

This job should not be skipped. If it is skipped as part of the nightly batch, it should run immediately after the nightly has completed with the I_business_date equal to the latest

business_date stored in the FLASHBACK_SNAPSHOT_INFO table. The data that is available to extract is based on the availability of the redo data for the SCN query.

Restart/Recovery

In the case of a restart from failure, it rebuilds records on ITEM_SUPP_COUNTRY_HIST and ITEM_SUPP_COUNTRY_LOC_HIST for the business date (current vdate is no input parameter passed in).

Design Assumptions

N/A

POS Configuration Data to 3rd Party POS (poscdnld)

Module Name	poscdnld.pc
Description	Download of POS Configuration Data to 3rd Party POS
Functional Area	Integration - 3rd Party POS
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS69
Wrapper Script	rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program downloads POS configuration information from Merchandising to a flat file. This file can be used to load POS and back-office systems. This program (and its related prepost function) should only be run if Merchandising is used to master:

- Coupon definitions and relationships to items
- Restrictions on product sales, including but not limited to minimum age of purchaser, time/days when product cannot be sold, tenders that cannot be used to purchase the product, and so on.

Restart/Recovery

The logic unit of work is pos configuration type and pos configuration ID. The commit_max_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter.
Integration Contract	IntCon000063

Output File Layout

Table 6-5 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record Type	Char(5)	'FHEAD'	Record Identifier
	Line id	Number(10)	0000000001	Sequential Line Identifier
	File Name	Char(4)	'POSC'	File Identifier
	File Date	Char(14)	N/A	Date the file was created in 'YYYYMMDD HHMMSS' format
TCOUP	Record Type	Char(5)	TCOUP	Record Identifier
	Line id	Number(10)	N/A	Sequential Line Identifier
	Coupon id	Number(6)	N/A	N/A
	Coupon Desc	Char(250)	N/A	N/A
	Currency Code	Char2(3)	N/A	N/A
	Max Discount Amount	Number(20,4)	N/A	N/A
	Amount	Number(20,4)	N/A	N/A
	Percent Ind	Char(1)	'N' - Amount 'Y' - Percentage	N/A
	Profit Center	Char(6)	N/A	N/A
	Tax Class	Char(6)	N/A	N/A
	Export Code	Char(6)	N/A	N/A
	Effective Date	Char(14)	N/A	Indicates the first day the coupon can be used in 'YYYYMMDD HHMMSS' format
	Expiration Date	Char(14)	N/A	Indicates the day the coupon becomes invalid in 'YYYYMMDD HHMMSS' format
	Prompted Ind	Char(1)	'Y', 'N'	This indicator identifies if the cashier should be prompted to ask for a Coupon.
	Display Ind	Char(1)	'Y', 'N'	This indicator specifies whether the coupon is displayed in the list of valid coupons on the register.

Table 6-5 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TPRES	Status	Char(1)	'A','C','D'	Indicates if the coupon configuration is new, has been changed, or being deleted.
	Vendor	Number(10)	N/A	N/A
	Vendor Type	Char(6)	'AG' - Agent 'AP' - Applicant 'BK' - Bank 'BR' - Broker 'CN' - Coonsignee 'CO' - Consolidator 'FA' - Factory 'FF' - Freight Forwarder 'IM' - Importer 'SU' - Supplier	N/A
	Promotion	Number(10)	N/A	N/A
	Coupon Barcode	Char(20)	N/A	N/A
	Coupon Max Qty	Number(6)	N/A	N/A
	Record Type	Char(5)	TPRES	Record Identifier
	Line id	Number(10)	N/A	Sequential Line Identifier
	POS Product Restriction id	Number(6)	N/A	N/A
	POS Product Restriction Desc	Char(120)	N/A	N/A

Table 6-5 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	POS Product Restriction Type	Char(6)	'PPRT' include: 'STMP' - Food Stamp 'MNAG' - Minimum Age 'CNDP' - Container Deposit 'CNVL' - Container Redemption Value 'DTDR' - Day/Time/Date Restriction 'TENT' - Tender Type 'NDSC' - Non-Discountable 'RTRN' - Returnable 'QLMT' - Quantity Limit	N/A
	Effective Date	Char(14)	N/A	Date the product restriction is first effective in 'YYYYMMDD HHMMSS' format
	Currency Code	Char(3)	N/A	N/A
	Product Restriction Amount	Number(20,4)	N/A	N/A
	Age Minimum	Number(2)	N/A	N/A
	Date Restriction	Char(14)	N/A	Date on which a specified product restriction is applied in 'YYYYMMDD HHMMSS' format
	Before Time Restriction	Char(6)	N/A	N/A
	After Time Restriction	Char(6)	N/A	N/A
	Day Restriction	Char(6)	N/A	N/A
	Max Qty Amount	Number(12,4)	N/A	N/A

Table 6-5 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Tender Type Group	Char(6)	'CASH' - Cash,	N/A
			'CHECK' - Check,	
			'CCARD' - Credit,	
			'COUPON' - Coupon,	
			'LOTTRY' - Lottery,	
			'FSTAMP' - Food Stamp,	
			'DCARD' - Debit Card,	
			'MORDER' - Money Order	
			'VOUCH' - Voucher	
			'ERR' - Error,	
	Status	Char(1)	'SOCASS' - Social Assistance,	Indicates if the product restriction configuration is new, has been changed, or being deleted.
			'TERM' - Termination Record,	
			'DRIVEO' - Drive Off,	
			'EBS' - Electronic Benefits (Food Stamps)	
			'A','C','D'	
TSTOR	Record Type	Char(5)	'TSTOR'	N/A
	Line id	Number(10)	N/A	N/A
	Store	Number(10)	N/A	N/A
	Status	Char(1)	'A' - Add 'D' - Delete 'C' - Change	N/A
TITEM	Record Type	Char(5)	TITEM	Record Identifier
	Line id	Number(10)	N/A	Sequential Line Identifier
	Item	Char(25)	N/A	Left-Justified Item Identifier

Table 6-5 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	Status	Char(1)	'A' - Add 'D' - Delete 'C' - Change	Indicates the item's status at the POS. Overlays of items as a result of a change to the merch criteria will have a 'C' status.
	Record Type	Char(5)	FTAIL	Marks end of file
	Line id	Number(10)	N/A	Total number of lines in file
	Number of transactions	Number(10)	N/A	Number of transactions in file

Design Assumptions

N/A

Price History Publication API (BDI_PriceHist_Fnd_PF_From_RMS_JOB)

This section describes the Price History Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Price History positions from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: **bdifoundations/b.pls**

```
BDI_FOUNDATION_SQL.PRICE_HIST_UP(0_error_message IN OUT VARCHAR2,
                                0_control_id      IN OUT NUMBER,
                                I_job_context      IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Price History tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Price History	Price History upload to BDI	PriceHist_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
PRICE_HIST_OUT	No	Yes	No	No
PRICE_HIST	Yes	No	No	No

Related Items Export (export_relitem.ksh)

Module Name	export_relitem.ksh
Description	Extraction of related item data
Functional Area	Foundation
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS262
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract new, updated and deleted Merchandising related items information into a flat file.

- This batch will support both a full and delta export of related item data.
- A threading indicator parameter should be passed. Passing 'Y' means a thread number (1-20) will be passed in. Passing 'N' means no thread number will be passed in and the program will use a default thread number.
- In full mode, normal operation will produce both a corporate level files and files for all stores. An optional input parameter will also allow the program to produce location level files for a specified store.
- In delta mode, this will produce both corporate level files and files for all stores the modified data are ranged to.
- This will export data only for approved, sellable items.

- This will export item related item information from the related item export staging and related item tables.
- Two types of flat files will be created for this extract - one for the related item header information and one for the related item detail information.
- When creating the location level files, ensure that both items (the main item and related item) are ranged in the location.
- The flat files that will be created will now be pipe delimited.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	relitemhead_date_corp_[full/delta]_[#ofLines].dat relitemhead_date_[Location]_[full/delta]_[#ofLines].dat relitemdet_date_corp_[full/delta]_[#ofLines].dat relitemdet_date_[Location]_[full/delta]_[#ofLines].dat
Integration Contract	IntCon000210 IntCon000211

Design Assumptions

N/A

Related Item Publication API (BDI_RelatedItem_Fnd_PF_From_RMS_JOB)

This section describes the Related Item Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Related Items from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

BDI_ITEM_SQL.REL_ITEM_UP(0_error_message	IN	OUT	VARCHAR2,
0_control_id	IN	OUT	NUMBER,
I_job_context	IN		VARCHAR2)

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound tables that reside in the BDI_RMS_INT_SCHEMA schema. These outbound tables are loaded with records from the Merchandising Related Item head and detail tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Related Item	Related Item upload to BDI	RelatedItem_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
RELATED_ITEM_OUT	No	Yes	No	No
RELATED_ITEM_DTL_OUT	No	Yes	No	No
RELATED_ITEM_HEAD	Yes	No	No	No
RELATED_ITEM_DETAIL	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No

UDA Item Date Publication API (BDI_UdaltemDate_Fnd_PF_From_RMS_JOB)

This section describes the UDA Item Date Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.UDA_ITEM_DATE_UP(0_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                                0_control_id    IN OUT NUMBER,
                                I_job_context    IN    VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising UDA Item Date table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
UDA ITEM DATE	UDA Item Date upload to BDI	UdaltemDate_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UDA_ITEM_DATE_OUT	No	Yes	No	No
UDA_ITEM_DATE	Yes	No	No	No

UDA Item FF Publication API (BDI_UdaltemFF_Fnd_PF_From_RMS_JOB)

This section describes the UDA Item FF Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.UDA_ITEM_FF_UP(0_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                             0_control_id   IN OUT NUMBER,
                             I_job_context   IN   VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising UDA Item FF table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
UDA ITEM FF	UDA Item FF upload to BDI	UdaltemFF_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UDA_ITEM_FF_OUT	No	Yes	No	No
UDA_ITEM_FF	Yes	No	No	No

UDA Item LOV Publication API (BDI_UdaltemLov_Fnd_From_RMS_JOB)

This section describes the UDA Item LOV Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Code Head information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdiitemb.pls

```
BDI_ITEM_SQL.UDA_ITEM_LOV_UP(0_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                             0_control_id    IN OUT NUMBER,
                             I_job_context   IN    VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising UDA Item LOV table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables. A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
UDA ITEM LOV	UDA Item LOV upload to BDI	UdaItemLov_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
UDA_ITEM_LOV_OUT	No	Yes	No	No
UDA_ITEM_LOV	Yes	No	No	No

Financials

Merchandising stages General Ledger (GL) data for subsequent upload into a financial system. A set of batch processes gather and organize the data before using it to populate the staging table, STG_FIF_GL_DATA.

For more information about how data moves from these staging tables to the General Ledger of a financial application and other integration between Merchandising and financial applications, see *Oracle Retail Financial Integration for Oracle Retail Merchandise Operations Management and Oracle E-Business Suite Financials Implementation Guide*

The following scheduled outbound integrations are included in this functional area:

- [Daily or Weekly Download of Stock Ledger Data \(stlgdnl\)](#)
- [Finance General Ledger to RFI \(BDI_RFI_FinGenLdgr_Tx_PF_From_RMS_JOB\)](#)
- [Financial Data Publish Extract \(fif_gl_publish_extract\)](#)
- [Invoice Matching Accounts Payable Detail Outbound Publish Extract \(fif_ap_detail_publish_extract\)](#)
- [Invoice Matching Accounts Payable Header Publish Extract \(fif_ap_head_publish_extract\)](#)
- [Fixed Deal Income \(dealfinc\)](#)
- [Franchise Billing Extract \(wfbillex.ksh\)](#)
- [General Ledger Cross Reference Dynamic Mapping \(fif_gl_dynamic_map\)](#)
- [Item/Location Daily Stock Ledger Transactions \(fifglnd1\)](#)
- [Monthly Stock Ledger Transactions \(fifglnd3\)](#)
- [Open to Buy Download Stock Ledger \(otbdlsal\)](#)
- [RMS Financial Data Publish \(fif_gl_publish\)](#)
- [Rolled Up Daily Stock Ledger Transactions \(fifglnd2\)](#)
- [Stage G/L Extracts \(gl_extract.ksh\)](#)
- [Tran Data Publication \(BDI_TranData_Tx_PF_From_RMS_EOW_JOB\)](#)
- [Capture Stock Ledger Cut-off Time for Business Date \(stockledger_switchtime.ksh\)](#)
- [Maintain Transaction Data Records \(maintain_iftd_hist.ksh\)](#)

Daily or Weekly Download of Stock Ledger Data (stlgdnld)

Module Name	stlgdnld.pc
Description	Weekly or Historical Download of Stock Ledger Data
Functional Area	Stock Ledger
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS17
Wrapper Script	batch_stlgdnld.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program extracts stock ledger data at the item level. The program can extract data for a historic period or for the most current complete week. The program accepts an input file that determines whether the extract is a historic extract or a weekly extract.

This program is often used in integration with RPAS applications.

Restart/Recovery

The logical unit of work for this program is set at item, location type, location and date. Threading is done by dept using the v_restart_dept view to thread properly.

The changes will be posted when the commit_max_ctr value is reached. The commit_max_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The value of the counter is subject to change based on implementation.

I/O Specification

Integration Type	Download from Merchandising
File Name	The input filename is a runtime parameter. The output filename is hardcoded to stldgr%d.dat where %d is substituted with the domain id. Each run of the program can produce multiple output files, one for each department. Additional input parameters are defined in the input file
Integratin Contract	IntCon000034 (output file)

Input File Layout

Table 6-6 Input File Layout

Field Name	Field Type	Default Value	Description
Task Indicator	Char(1)	N/A	Task Indicator. Valid values are 'H' - historical, 'W' - weekly

Table 6-6 (Cont.) Input File Layout

Field Name	Field Type	Default Value	Description
From Date	Char(8)	N/A	From Date in 'YYYYMMDD' format
To Date	Char(8)	N/A	To Date in 'YYYYMMDD' format

Output File Layout

Table 6-7 Output File Layout

Field Name	Field Type	Default Value	Description
Item	Char(25)	N/A	Item number
Location Type	Char(1)	N/A	Location Type Valid values are 'S','W'
Location	Number(20)	N/A	Location Number
Eow_date	Char(8)	N/A	End of Week date in 'YYYYMMDD' format
Update_Ind	Char(1)	N/A	Update Indicator Valid values are 'I' and 'U'
Regular_sales_retail	Number(25,4)	N/A	Regular sales value (retail)
Regular_sales_cost	Number(25,4)	N/A	Regular sales value (cost)
Regular_sales_units	Number(17,4)	N/A	Regular sales value (units)
Promo_sales_retail	Number(25,4)	N/A	Promo sales value (retail)
Promo_sales_cost	Number(25,4)	N/A	Promo sales value (cost)
Promo_sales_units	Number(17,4)	N/A	Promo sales value (units)
Clear_sales_retail	Number(25,4)	N/A	Clearance sales value (retail)
Clear_sales_cost	Number(25,4)	N/A	Clearance sales value (cost)
Clear_sales_units	Number(17,4)	N/A	Clearance sales value (units)
Sales_retail_excluding_vat	Number(25,4)	N/A	Sales value excluding vat (retail)
Custom_returns_retail	Number(25,4)	N/A	Custom returns value (retail)
Custom_returns_cost	Number(25,4)	N/A	Custom returns value (cost)
Custom_returns_units	Number(17,4)	N/A	Custom returns value (units)

Table 6-7 (Cont.) Output File Layout

Field Name	Field Type	Default Value	Description
Rtv_retail	Number(25,4)	N/A	Return to Vendor value (retail)
Rtv_cost	Number(25,4)	N/A	Return to Vendor value (cost)
Rtv_units	Number(17,4)	N/A	Return to Vendor value (units)
Reclass_in_retail	Number(25,4)	N/A	Reclass In value (retail)
Reclass_in_cost	Number(25,4)	N/A	Reclass In value (cost)
Reclass_in_units	Number(17,4)	N/A	Reclass In value (units)
Reclass_out_retail	Number(25,4)	N/A	Reclass Out value (retail)
Reclass_out_cost	Number(25,4)	N/A	Reclass Out value (cost)
Reclass_out_units	Number(17,4)	N/A	Reclass Out value (units)
Perm_markdown_value	Number(25,4)	N/A	Permanent markdown value (retail)
Prom_markdown_value	Number(25,4)	N/A	Promotion markdown value (retail)
Clear_markdown_value	Number(25,4)	N/A	Clearance markdown value (retail)
Markdown_cancel_value	Number(25,4)	N/A	Markdown cancel value
Markup_value	Number(25,4)	N/A	Markup value
Markup_cancel_value	Number(25,4)	N/A	Markup cancel value
Stock_adj_retail	Number(25,4)	N/A	Stock adjustment value (retail)
Stock_adj_cost	Number(25,4)	N/A	Stock adjustment value (cost)
Stock_adj_units	Number(17,4)	N/A	Stock adjustment value (units)
Received_retail	Number(25,4)	N/A	Received value (retail)
Received_cost	Number(25,4)	N/A	Received value (cost)
Received_units	Number(17,4)	N/A	Received value (units)
Tsf_in_retail	Number(25,4)	N/A	Transfer In value (retail)
Tsf_in_cost	Number(25,4)	N/A	Transfer In value (cost)
Tsf_in_units	Number(17,4)	N/A	Transfer In value (units)
Tsf_out_retail	Number(25,4)	N/A	Transfer Out value (retail)

Table 6-7 (Cont.) Output File Layout

Field Name	Field Type	Default Value	Description
Tsf_out_cost	Number(25,4)	N/A	Transfer Out value (cost)
Tsf_out_units	Number(17,4)	N/A	Transfer Out value (units)
Freight_cost	Number(25,4)	N/A	Freight cost
Employee_disc_retail	Number(25,4)	N/A	Employee disc (retail)
Cost_variance	Number(25,4)	N/A	Cost variance
Wkroom_other_cost_sales	Number(25,4)	N/A	Wkroom other sales (cost)
Cash_disc_retail	Number(25,4)	N/A	Cash disc (retail)
Freight_claim_retail	Number(25,4)	N/A	Freight Claim (retail)
Freight_claim_cost	Number(25,4)	N/A	Freight Claim (cost)
Freight_claim_units	Number(25,4)	N/A	Freight Claim (Units)
Stock_adj_cogs_retail	Number(25,4)	N/A	Stock Adjust COGS (retail)
Stock_adj_cogs_cost	Number(25,4)	N/A	Stock Adjust COGS (cost)
Stock_adj_cogs_units	Number(25,4)	N/A	Stock Adjust COGS (Units)
Intercompany_in_retail	Number(25,4)	N/A	Intercompany In value (retail)
Intercompany_in_cost	Number(25,4)	N/A	Intercompany In value (cost)
Intercompany_in_units	Number(25,4)	N/A	Intercompany In value (units)
Intercompany_out_retail	Number(25,4)	N/A	Intercompany Out value (retail)
Intercompany_out_cost	Number(25,4)	N/A	Intercompany Out value (cost)
Intercompany_out_units	Number(25,4)	N/A	Intercompany Out value (units)
Intercompany_markup	Number(25,4)	N/A	Intercompany Markup
Intercompany_markup_units	Number(25,4)	N/A	Intercompany Markup (units)
Intercompany_markdown	Number(25,4)	N/A	Intercompany Markdown
Intercompany_markdown_units	Number(25,4)	N/A	Intercompany Markdown (units)
Wo_activity_upd_inv	Number(25,4)	N/A	Work Order Activity - Update Inventory (cost)
Wo_activity_upd_inv_units	Number(25,4)	N/A	Work Order Activity - Update Inventory (units)

Table 6-7 (Cont.) Output File Layout

Field Name	Field Type	Default Value	Description
Wo_activity_post_fin	Number(25,4)	N/A	Work Order Activity - Post to Financials (retail)
Wo_activity_post_fin_units	Number(25,4)	N/A	Work Order Activity - Post to Financials (units)

Design Assumptions

N/A

Finance General Ledger to RFI (BDI_RFI_FinGenLdgr_Tx_PF_From_RMS_JOB)

Module Name	BDI_RFI_FinGenLdgr_Tx_PF_From_RMS_JOB
Description	Extracts financial general ledger to RFI
Functional Area	Finance
Module Type	Integration
Module Technology	BDI Job
Catalog ID	N/A
Runtime Parameters	FinGenLdgr_Tx_ProcessFlow_From_RMS FinGenLdgr_Tx_Extractor

Design Overview

This API extracts staged data from Merchandising and Sales Audit and transfers it to the General Ledger inbound staging tables. To accomplish this data transfer, BDI will call a Merchandising owned API that will pull data from RMS and deliver these to the BDI integration layer.

This integration is applicable when using Oracle Retail Financial Integration (RFI) to supported financial solutions. For more information on RFI, see the ORFI Implementation Guide, as part of the documentation for Oracle Retail Integration Cloud Service.

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Daily
Scheduling Consideration	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XML)
Finance	General Ledger upload to BDI	FinGenLdgr_Tx_BdiInterfaceModule.xml

Financial Data Publish Extract (fif_gl_publish_extract)

Module Name	fif_gl_publish_extract.ksh
Description	Financial Data Publish Extract
Functional Area	General Ledger
Module Type	Business Processing
Module Technology	ksh
Catalog ID	
Wrapper Script	batch_fif_gl_publish_extract.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will have the following runtime parameters:

Parameter	Description
application_name	Source of the GL record. Example: RMS or IM or RFM
file_prefix	Prefix for the generated file.
num_records_per_file	Number of records extracted per file. If -1, only 1 file will be generated.
upload_to_object_store	Default value N. Indicates if the generated file will only be uploaded to object store.
republish_ind	Default to N for nightly batch, Y for adhoc run to republish records.
republish_date	Required when republish_ind is Y. Extract date of records to republish [YYYYMMDD]

During nightly run, this job will extract records from FIF_GL_PUBLISH for the application name (either RMS or IM or RFM) where journal_entry_create_date = current date, write to file, and upload to object storage

Example: /outgoing/RMS_ACCOUNT_ENTRY_20250131024118.zip

The zip can contain the files like: RMS_ACCOUNT_ENTRY_20250131_20250131_00.csv

The num_records_per_file parameter should be big enough because the program can only generate up to 100 files that is packaged in the zip.

During adhoc run, records from FIF_GL_PUBLISH where journal_entry_create_date = republish date will be extracted.

If upload_to_object_store is Y, this job will extract records from FIF_GL_PUBLISH, write to file, and upload to object storage.

If upload_to_object_store is N, this job will invoke the merch background engine which will extract data from FIF_GL_PUBLISH table into a csv file, zip the file and publish to Fusion Financials using theCFIN by invoking a ReST end point.

Restart/Recovery

N/A

Key Tables Affected

Table	Select	Insert	Update	Delete
FIF_GL_PUBLISH	Yes	No	No	No
FIF_PUBLISH_STATUS	Yes	Yes	Yes	Yes
PERIOD	Yes	No	No	No

Fixed Deal Income (dealfinc)

Module Name	dealfinc.pc
Description	Calculation & Interface of Fixed Deal Income for General Ledger
Functional Area	Integration - General Ledger
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS65
Wrapper Script	rmswrap_multi.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This module writes to the STG_FIF_GL_DATA financial staging table to perform stock ledger processing for fixed deals. It splits deal income over all dept/class/subclass locations on the deal. This prorated income is written to the general ledger under a suitable cost center mapping.

Restart/Recovery

The logical unit of work for this program is a DEAL_ID. The database commit takes place when number of deal records processed is equal to the commit max counter in the restart control table.

I/O Specification

Integration Type	Download from Merchandising
File Name	N/A
Integration Contract	IntCon000019 STG_FIF_GL_DATA table

Design Assumptions

N/A

Franchise Billing Extract (wfbillex.ksh)

Module Name	wfbillex.ksh
Description	Franchise Billing Extract
Functional Area	Franchise Management
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS155
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this shell script module is to fetch all billing information for Franchise sale and return transactions and write these to an output file for integration with an external financial application that manages billing. A file is generated for each customer location (store)/day.

The format of the generated file is based on a run time parameter:

- If no parameter is passed or if the value 1 is used, the previously existing format (refer Output File Layout Format 1) will be generated.
- If the value 2 (or greater than 2) is passed, the new file format (refer Output File Layout Format 2) will be generated.

Restart/Recovery

The logical unit of work for this module is defined as the customer location (store). Only one commit will be done for a customer location that has been completely processed. The WFBX formatted output file will be created with a temporary name and renamed just before a customer location commit. In case of failure, all work done will be rolled back.

I/O Specification

Integration Type	Download from Merchandising
File Name	WFBX_<store>_<SYSDATE>
Integration Contract	IntCon000110

Output File Layout

Table 6-8 Output File Layout Format 1

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number

Table 6-8 (Cont.) Output File Layout Format 1

Record Name	Field Name	Field Type	Default Value	Description
THEAD	File type definition	Char(4)	WFBX	Identifies the file type
	File Create Date	Char(14)		File Create Date in YYYYMMDDHH MMSS format
	Record descriptor	Char(5)	THEAD	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Customer Location	Number(10)		Franchise store number
	Customer Order Reference Number	Char(20)		Reference number provided by the franchise customer
	Franchise Order Number	Number(10)		Franchise Order Number
	Transaction Type	Char(6)		SALES or RETURN
	RMA Number	Number(10)		Return Merchandise Authorization Number for the return
	Order Return Date	Number(8)		Order return date for Return transaction type or Order date for Sale transaction type in YYYYMMDD format
TDETL	Shipment Date	Number(8)		Date on which the item was shipped to the franchise location or returned to the retailer
	Record descriptor	Char(5)	TDETL	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Item	Char(25)		Item sequence number
	Department	Number(4)		Department number of the item

Table 6-8 (Cont.) Output File Layout Format 1

Record Name	Field Name	Field Type	Default Value	Description
TTAIL	Class	Number(4)		Class number of the item
	Subclass	Char(4)		Subclass number of the item
	Order Return Quantity	Number(12)		Return quantity with 4 implied decimal places
	Order Return Quantity UOM	Char(4)		Return quantity unit of measure
	Order Return Cost	Number(20)		Return cost for Return transaction type or Customer cost for Sale transaction type. For both it is the per-unit cost
	Freight Cost	Number(20)		Freight associated to the franchise order
	Return Restocking Fee	Number(20)		Unit restocking fee charged for received items
	VAT Code	Char(6)		VAT code for the item
	VAT Rate	Number(20)		VAT rate associated to the VAT code for the item
	Other Order Charges	Number(20)		Other charges for the item
TTAIL	Record descriptor	Char(5)	TTAIL	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Tran Record Counter	Number(6)		Number of TDETL records in this transaction set
FTAIL	Record descriptor	Char(5)	FTAIL	Identifies the file record type
	File Line Id	Number(10)		Sequential file line number

Table 6-8 (Cont.) Output File Layout Format 1

Record Name	Field Name	Field Type	Default Value	Description
	File Record counter	Number(10)		Number of records/ transactions processed in current file (only records between head & tail)

Table 6-9 Output File Layout Format 2

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	File type definition	Char(4)	WFBX	Identifies the file type
	File Create Date	Char(14)		File Create Date in YYYYMMDDHH MMSS format
	Version No.	Char(2)	02	Identifies the file format version
THEAD	Record descriptor	Char(5)	THEAD	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Customer Location	Number(10)		Franchise store number
	Customer Order Reference Number	Char(20)		Reference number provided by the franchise customer
	Franchise Order Number	Number(10)		Franchise Order Number
	Transaction Type	Char(6)		SALES or RETURN
	RMA Number	Number(10)		Return Merchandise Authorization Number for the return

Table 6-9 (Cont.) Output File Layout Format 2

Record Name	Field Name	Field Type	Default Value	Description
TDETL	Order Return Date	Number(8)		Order return date for Return transaction type or Order date for Sale transaction type in YYYYMMDD format
	Shipment Date	Number(8)		Date on which the item was shipped to the franchise location or returned to the retailer
	Record descriptor	Char(5)	TDETL	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Item	Char(25)		Item sequence number
	Department	Number(4)		Department number of the item
	Class	Number(4)		Class number of the item
	Subclass	Char(4)		Subclass number of the item
	Order Return Quantity	Number(12)		Return quantity with 4 implied decimal places
	Order Return Quantity UOM	Char(4)		Return quantity unit of measure
	Order Return Cost	Number(20)		Return cost for Return transaction type or Customer cost for Sale transaction type. For both it is the per-unit cost
	Freight Cost	Number(20)		Freight associated to the franchise order
	Return Restocking Fee	Number(20)		Unit restocking fee charged for received items
	VAT Code	Char(6)		VAT code for the item

Table 6-9 (Cont.) Output File Layout Format 2

Record Name	Field Name	Field Type	Default Value	Description
	VAT Rate	Number(20)		VAT rate associated to the VAT code for the item
	Other Order Charges	Number(20)		Other charges for the item
	Uom Type	Char(4)		Uom Type fetched from GTS tax engine to be considered when tax is in value
	Uom Value	Number(20)		UOM value fetched from GTS tax engine to be considered when tax is in value. Having 10 places of decimal digits.
	Uom Tax Value Per Unit	Number(20)		UOM tax value per unit fetched from GTS tax engine to be considered when tax is in value. Having 10 places of decimal digits.
TTAIL	Record descriptor	Char(5)	TTAIL	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Tran Record Counter	Number(6)		Number of TDETL records in this transaction set
FTAIL	Record descriptor	Char(5)	FTAIL	Identifies the file record type
	File Line Id	Number(10)		Sequential file line number
	File Record counter	Number(10)		Number of records/ transactions processed in current file (only records between head & tail)

Design Assumptions

N/A

General Ledger Cross Reference Dynamic Mapping (fif_gl_dynamic_map)

Module Name	fif_gl_dynamic_map.ksh
Description	General Ledger Cross Reference Dynamic Mapping
Functional Area	General Ledger
Module Type	Business Processing
Module Technology	ksh
Catalog ID	
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

If the system is set up to use dynamic segments (FINANCIAL_UNIT_OPTIONS.USE_DYNAMIC_SEGMENTS = Y) and there are changes to either GL cross reference or dynamic segment records, this program calls the DYNAMIC_MAPPING_SQL.PROCESS function to perform the following:

- Synchronize fully formed GL cross reference records between the FIF_GL_CROSS_REF and FIF_GL_CROSS_REF_DYNAMIC tables.
- Build fully formed combinations in FIF_GL_CROSS_REF_DYNAMIC using dynamic segment mappings and GL cross reference records based on the following order:
 - GL cross reference rows missing minimal segments to rows with maximal segments.
 - Fully formed GL cross reference rows with base indicator set.
 - Fully formed GL cross reference rows.
- When a GL cross reference record is deleted, all child records in FIF_GL_CROSS_REF_DYNAMIC will also be deleted. The program will attempt to rebuild the combination using existing GL Cross Reference records.
- When a dynamic segment mapping is deleted, all records in FIF_GL_CROSS_REF_DYNAMIC using this dynamic segment will be deleted.
- When a GL cross reference record is updated, all child records in FIF_GL_CROSS_REF_DYNAMIC will also be updated. The account combination will need to be re-validated.
- When a dynamic segment mapping is updated, all records in FIF_GL_CROSS_REF_DYNAMIC using this dynamic segment will be updated. The account combination will need to be re-validated.
- After building the combinations, validate the account against the FIF_GL_ACCT table. If the account does not exist, it will be deleted from FIF_GL_CROSS_REF_DYNAMIC.

Records from FIF_GL_CROSS_REF_DYNAMIC are used by fifglnd1, fifglnd2, fifglnd3, and dealfinc batch programs when getting General Ledger cross-reference mappings for the transaction data.

Restart/Recovery

N/A

Design Assumptions

N/A

Invoice Matching Accounts Payable Detail Outbound Publish Extract
(fif_ap_detail_publish_extract)

Module Name	fif_ap_detail_publish_extract.ksh
Description	Invoice Matching Accounts Payable Detail Publish Extract
Functional Area	General Ledger
Module Type	Business Processing
Module Technology	ksh
Catalog ID	
Wrapper Script	batch_fif_ap_publish_extract.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will have the following runtime parameters:

Parameter	Description
application_name	Source of the GL record. Valid values: IM or RFM
file_prefix	Prefix for the generated file.
num_records_per_file	Number of records extracted per file. If -1, only 1 file will be generated.
upload_to_object_store	Default value N. Indicates if the generated file will only be uploaded to object store.
republish_ind	Default to N for nightly batch, Y for adhoc run to republish records.
republish_date	Required when republish_ind is Y. Extract date of records to republish [YYYYMMDD]
republish_date_run_id	Run Id from the extract date of records to republish. Valid values between 1 and 9999. Optional. Defaulted to 1 if Null.

During nightly run, this job will extract records from FIF_AP_DETAIL_PUBLISH for the application name (either IM or RFM) where im_create_date = current date, write to file, and upload to object storage.

Example: /outgoing/INVOICE_DETAIL_20250131024118.zip

The zip can contain the files like: INVOICE_DETAIL_20250131_20250131_00.csv

The num_records_per_file parameter should be big enough because the program can only generate up to 100 files that is packaged in the zip.

Invoice publish from Invoice Matching to AP can happen multiple times in a day if the job is scheduled to run multiple times in a day. Each run is assigned a one up number, starting from 1 for that day.

During republish to AP for exception scenarios, the run ID for the batch run (republish_date_run_id) can be provided to selectively republish a particular job run. In this case, if a run ID is not provided, it is defaulted to 1. That is, during republish, records from FIF_AP_DETAIL_PUBLISH where im_create_date = republish_date and extract_date_run_id = republish_date_run_id will be extracted. If republish_date_run_id is not provided, then records with extract_date_run_id = 1 will be extracted.

Restart/Recovery

N/A

Key Tables Affected

Table	Select	Insert	Update	Delete
FIF_AP_DETAIL_PUBLISH	Yes	No	Yes	No
FIF_PUBLISH_STATUS	Yes	Yes	Yes	Yes
PERIOD	Yes	No	No	No

Invoice Matching Accounts Payable Header Publish Extract (fif_ap_head_publish_extract)

Module Name	fif_ap_head_publish_extract.ksh
Description	Invoice Matching Accounts Payable Header Publish Extract
Functional Area	General Ledger
Module Type	Business Processing
Module Technology	ksh
Catalog ID	
Wrapper Script	batch_fif_ap_publish_extract.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will have the following runtime parameters:

Parameter	Description
application_name	Source of the GL record. Valid values: IM or RFM
file_prefix	Prefix for the generated file.
num_records_per_file	Number of records extracted per file. If -1, only 1 file will be generated.
upload_to_object_store	Default value N. Indicates if the generated file will only be uploaded to object store.
republish_ind	Default to N for nightly batch, Y for adhoc run to republish records.

Parameter	Description
republish_date	Required when republish_ind is Y. Extract date of records to republish [YYYYMMDD]
republish_date_run_id	Run Id from the extract date of records to republish. Valid values between 1 and 9999. Optional. Defaulted to 1 if Null.

During nightly run, this job will extract records from FIF_AP_HEAD_PUBLISH for the application name (either IM or RFM) where im_create_date = current date, write to file, and upload to object storage.

Example: /outgoing/INVOICE_HEADER_20250131024118.zip

The zip can contain the files like: INVOICE_HEADER_20250131_20250131_00.csv

The num_records_per_file parameter should be big enough because the program can only generate up to 100 files that is packaged in the zip.

Invoice publish from Invoice Matching to AP can happen multiple times in a day if the job is scheduled to run multiple times in a day. Each run is assigned a one-up number, starting from 1 for that day.

During republish to AP for exception scenarios, the run ID for the batch run (republish_date_run_id) can be provided to selectively republish a particular job run. In this case, if a run ID is not provided, it is defaulted to 1. That is, during republish, records from FIF_AP_HEAD_PUBLISH where im_create_date = republish date and extract_date_run_id = republish_date_run_id are extracted. If republish_date_run_id is not provided, then records with extract_date_run_id = 1 will be extracted.

If upload_to_object_store is Y, this job will extract records from FIF_AP_HEAD_PUBLISH, write to file, and upload to object storage.

If upload_to_object_store is N, this job will extract data from FIF_AP_HEAD_PUBLISH and FIF_AP_DETAIL_PUBLISH table into a csv file, zip the file and publish to Fusion Financials using the ReST end point.

Restart/Recovery

N/A

Key Tables Affected

Table	Select	Insert	Update	Delete
FIF_AP_HEAD_PUBLISH	Yes	No	Yes	No
FIF_AP_DETAIL_PUBLISH	Yes	No	Yes	No
FIF_PUBLISH_STATUS	Yes	Yes	Yes	Yes
PERIOD	Yes	No	No	No

Item/Location Daily Stock Ledger Transactions (fifglnd1)

Module Name	fifglnd1sqls.pls/fifglnd1sqlb.pls
Description	Interface to General Ledger of Item/Loc Level Transactions
Functional Area	General Ledger

Module Type	Integration
Module Technology	PLSQL
Catalog ID	RMS66
Wrapper Script	rmswrap_plsql.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program extracts the detailed stock ledger information for certain transaction types on a daily basis in order to bridge the information to an interfaced financial application. The program reads from the IF_TRAN_DATA table for each transaction type/amount type and posts it to the Oracle Retail General Ledger staging table at the SKU detail level.

If transactions exist for which GL cross mappings do not exist, these will be logged in TRAN_DATA_ERRORS and a notification will be sent to the Finance Analyst user indicating that unmapped transactions exist. The TRAN_DATA_ERRORS table is available through the Data Access Schema, enabling users to view the errors and create the missing mappings. The fifgldn1 module will attempt to reprocess records from TRAN_DATA_ERRORS if mappings have been created. During the month-end processing run, if unmapped transactions still exist they will be posted into the clearing accounts as outlined below:

1. The cost value of the transaction will be posted into the Credit Clearance account [Credit Clearance Segment 1-10] associated with the location's set of books.
2. The cost value of the transaction will be posted into the Debit Clearance account [Debit Clearance Segment 1-10] associated with the location's set of books.
3. The retail value of the transaction will be posted into the Credit Clearance account [Credit Clearance Segment 1-10] associated with the location's set of books.
4. The retail value of the transaction will be posted into the Debit Clearance account [Debit Clearance Segment 1-10] associated with the location's set of books.

The month-end processing is typically executed on the End of Month (EOM) date plus Close Month After Days to allow for late-posted transactions to be attributed to the correct month. If on this date, clearing accounts are not defined and unresolved mappings continue to exist, then the month will remain open for an additional number of days specified in **Force Close Month After Additional Days**. After this duration, the fifgldn1 module will move the records-pending posting from TRAN_DATA_ERRORS to STG_FIF_GL_EOM_ERRORS and proceed to close the month without failure. The STG_FIF_GL_EOM_ERRORS table is available through DAS/RDS to enable extraction and processing of unposted transaction records.

Restart/Recovery

The logical unit of work is department/class/subclass/location/item. The batch is multithreaded using the restart department view.

I/O Specification

Integration Type	Download from Merchandising
File Name	N/A

Integration Contract IntCon000019
STG_FIF_GL_DATA table

Design Assumptions

N/A

Monthly Stock Ledger Transactions (fifgldn3)

Module Name	Fifgldn3sqls.pls/fifgldn3sqlb.pls
Description	General Ledger Interface 3
Functional Area	Interface to General Ledger of Month Level Information
Module Type	Integration
Module Technology	PLSQL
Catalog ID	RMS68
Wrapper Script	rmswrap_plsql.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program summarizes stock ledger data from the monthly stock ledger table based on the level of information required and writes it to the financial general ledger staging table. The transactions extracted are determined by the CODE_TYPE GLRT (general ledger rolled transactions). Written information is then sent to the financial application. Stock ledger information may be rolled-up at department, class or subclass level. The level at which information is rolled-up to is determined by the system parameter GL_ROLLUP.

The fifgldn3 module is scheduled to run monthly and operates off both the TRAN_DATA and MONTH_DATA tables. If the module encounters an unmapped transaction, depending on whether the source of the unmapped transaction is TRAN_DATA or MONTH_DATA, the transaction will be logged into TRAN_DATA_ERRORS or MONTH_DATA_ERRORS tables, respectively. If clearing accounts have been defined, these unmapped transactions will be posted into the clearing accounts as outlined below:

1. The cost value of the transaction will be posted into the Credit Clearance account [Credit Clearance Segment 1-10] associated with the location's set of books.
2. The cost value of the transaction will be posted into the Debit Clearance account [Debit Clearance Segment 1-10] associated with the location's set of books.
3. The retail value of the transaction will be posted into the Credit Clearance account [Credit Clearance Segment 1-10] associated with the location's set of books.
4. The retail value of the transaction will be posted into the Debit Clearance account [Debit Clearance Segment 1-10] associated with the location's set of books.

A notification will be sent to the Finance Analyst user indicating that unmapped transactions exist and whether the postings to the clearance accounts have occurred.

The month-end processing is typically executed on the End of Month (EOM) date plus Close Month After Days to allow for late-posted transactions to be attributed to the correct month. If

on this date, clearing accounts are not defined and unresolved mappings continue to exist, the fifgldn3 module will move the records-pending posting from TRAN_DATA_ERRORS and MONTH_DATA_ERRORS to STG_FIF_GL_EOM_ERRORS and proceed to close the month without failure. The STG_FIF_GL_EOM_ERRORS table is available through DAS/RDS to enable extraction and processing of unposted transaction records.

Restart/Recovery

The logical unit of work is dependent on the level of rollup defined in the system options table. It can be department (department rollup), department/class (class rollup) or department/class/subclass (subclass rollup). The batch is multithreaded using the restart all locations view.

I/O Specification

Integration Type	Download from Merchandising
File Name	N/A
Integration Contract	IntCon000019 STG_FIF_GL_DATA table

Design Assumptions

N/A

Open to Buy Download Stock Ledger (otbdlsal)

Module Name	otbdlsal.pc
Description	Open To Buy Download Stock Ledger
Functional Area	OTB - Stock Ledger to Planning System Interface
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS16
Wrapper Script	Rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This module will sum stock ledger data from the DAILY_DATA table and opening stock information from the WEEK_DATA table across the current week, grouping by department, class, subclass, location and date, and export the data to a flat file for use by an outside planning system.

Restart/Recovery

The logical unit of work for the OTBDLSAL module is department, class, subclass and location. The commit_max_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records. Each time the record counter equals the maximum recommended commit number, an

application image array record will be written to the restart_start_array for restart/recovery if a fatal error occurs.

Locking Strategy

N/A

Security Considerations

N/A

Performance Considerations

N/A

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	OTB - Stock Ledger to Planning System Interface IntCon00030

Output File Format

Table 6-10 File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Sequence Number	Number(10)	0000000001	Keeps track of the record's position in the file by line number
	File Type Definition	Char(4)	STKE	Identifies file as Stock Ledger Export
	File Create Date	Char(14)	vdate	Date file was written by batch program in YYYYMMDD format. Remaining six characters are blank.
FDETL	File Type Record Descriptor	Char(5)	FDETL	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	Keeps track of the record's position in the file by line number

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Transaction Set Control Number	Number(14)	sequence number	Used to force unique file check
	Department	Number(4)	N/A	The ID number of a department
	Class	Number(4)	N/A	The ID number of a class within the department given
	Subclass	Number(4)	N/A	The ID number of a subclass within the class given
	Loc_type	Char(1)	N/A	The type of the location from which stock ledger data was collected
	Location	Number(10)	N/A	The location from which stock ledger data was collected
	Half No.	Number(5)	N/A	The half number for this stock ledger data
	Month No.	Number(2)	N/A	The month number in the half for this stock ledger data
	Week No.	Number(2)	N/A	The week number in the month for this stock ledger data
	Open Stock Retail	Number(20,4)	N/A	The retail opening stock from the week_data table *10000 (implied 4 decimal places) for this stock ledger period
	Open Stock Cost	Number(20,4)	N/A	The cost opening stock from the week_data table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Stock Adjustments Retail	Number(20,4)	N/A	The retail stock adjustments summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Stock Adjustments Cost	Number(20,4)	N/A	The cost stock adjustments summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Purchases Retail	Number(20,4)	N/A	The retail purchases summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Purchases Cost	Number(20,4)	N/A	The cost purchases summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	RTV Retail	Number(20,4)	N/A	The retail return to vendor amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	RTV Cost	Number(20,4)	N/A	The cost return to vendor amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Freight Cost	Number(20,4)	N/A	The freight cost summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Net Sales Retail	Number(20,4)	N/A	The retail net sales summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Net Sales Cost	Number(20,4)	N/A	The cost net sales summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Returns Retail	Number(20,4)	N/A	The retail returns amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Returns Cost	Number(20,4)	N/A	The cost returns amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Promotional Markdowns Retail	Number(20,4)	N/A	The retail promotional markdowns summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Markdown Cancellations Retail	Number(20,4)	N/A	The retail markdown cancellations summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Employee Discount Retail	Number(20,4)	N/A	The retail employee discounts amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Workroom Amount	Number(20,4)	N/A	The workroom amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Cash Discount Amount	Number(20,4)	N/A	The cash discounts amount summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Sales Units	Number(12,4)	N/A	The sales units summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Markups Retail	Number(20,4)	N/A	The retail markups summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Markup Cancellations Retail	Number(20,4)	N/A	The retail markup cancellations summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Clearance Markdowns Retail	Number(20,4)	N/A	The retail clearance markdowns summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Permanent Markdowns Retail	Number(20,4)	N/A	The retail permanent markdowns summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Freight Claim Retail	Number(20,4)	N/A	The retail freight claim summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Freight Claim Cost	Number(20,4)	N/A	The cost freight claim summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Stock Adjust Cost of Goods Sold (COGS) Retail	Number(20,4)	N/A	The retail stock adjust COGS summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Stock Adjust Cost of Goods Sold (COGS) Cost	Number(20,4)	N/A	The cost stock adjust COGS summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Inter-company In Retail	Number(20,4)	N/A	The Inter-company In retail summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Inter-company In Cost	Number(20,4)	N/A	The Inter-company In cost summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Inter-company Out Retail	Number(20,4)	N/A	The Inter-company Out Retail summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Inter-company Out Cost	Number(20,4)	N/A	The Inter-company Out Cost summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Inter-company Markup	Number(20,4)	N/A	The Inter-company Markup summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Inter-company Markdown	Number(20,4)	N/A	The Inter-company Markdown summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	Work Order Activity Update Inventory	Number(20,4)	N/A	The Work Order Activity Update Inventory summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period

Table 6-10 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	Work Order Activity Post Finishing	Number(20,4)	N/A	The Work Order Activity Post Finishing summed from the DAILY_DATA table *10000 (implied 4 decimal places) for this stock ledger period
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence Number	Number(10)	N/A	Keeps track of the record's position in the file by line number
	Control Number File Line Count	Number(10)	N/A	Total number of all transaction lines, not including file header and trailer

Design Assumptions

N/A

Populate FLASHBACK_SNAPSHOT_INFO table (stockledger_switchtime)

Module Name	stockledger_switchtime.ksh (flashbacksnapshots.sql/flashbacksnapshotb.sql)
Description	Populate FLASHBACK_SNAPSHOT_INFO table with database SCN and timestamp values
Functional Area	Stock Ledger
Module Type	Business Processing
Module Technology	PLSQL
Catalog ID	
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch capture the current VDATE, timestamp, and the corresponding SCN into a table named FLASHBACK_SNAPSHOT_INFO. This SCN value will be used to extract the point in time data from various tables as part of other batch processing such as maintain_isc_history.

It will also delete the previous day record. If corresponding records from the previous day do not exist, this batch will fail and will require troubleshooting to ensure that no data is lost for the RDE extract and other dependent modules.

Restart/Recovery

This is restartable.

Design Assumptions

N/A

Populate IF_TRAN_DATA_HIST and IF_FUTURE_TRAN_DATA_HIST tables for RDE Extract (maintain_iftd_hist)

Module Name	maintain_iftd_hist.ksh (iftdhistsqls.pls/ iftdhistsqlb.pls)
Description	Populate IF_TRAN_DATA_HIST and IF_FUTURE_TRAN_DATA_HIST tables for RDE Extract
Functional Area	Stock Ledger
Module Type	Business Processing
Module Technology	PLSQL
Catalog ID	
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program copies current data from IF_TRAN_DATA into IF_TRAN_DATA_HIST and drops old partitions based on defined retention days (SYSTEM_OPTIONS.IFTD_HIST_RETENTION_DAYS). It will also extract data into IF_FUTURE_TRAN_DATA_HIST based on IF_FUTURE_TRAN_DATA. It inserts a row in RMS_RDE_BATCH_STATUS table stating that the TRAN_DATA staging is complete for the business date.

Restart/Recovery

In the case of a restart from failure, it rebuilds records on IF_TRAN_DATA_HIST for the current business date.

Design Assumptions

N/A

RMS Financial Data Publish (fif_gl_publish)

Module Name	fif_gl_publish.ksh
Description	RMS Financial Data Publish
Functional Area	General Ledger

Module Type	Business Processing
Module Technology	ksh
Catalog ID	N/A
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job calls the program FIF_GL_PUBLISH_SQL.PROCESS that loads records from STG_FIF_GL_DATA for the current date to the FIF_GL_PUBLISH table. The reference and attribute columns in this table will be populated with the value from the STG_FIF_GL_DATA column based on the mappings defined in the FIF_OUT_ATTRIBUTE_MAPPING table. If no mapping is defined for any of the fields in this table, a direct copy is performed.

If the set of books from STG_FIF_GL_DATA does not have an external reference ID defined in FIF_GL_SETUP, this program creates an error.

Restart/Recovery

N/A

Key Tables Affected

Table	Select	Insert	Update	Delete
FIF_OUT_ATTRIBUTE_MAPPING	Yes	No	No	No
RMS_PLSQL_BATCH_CONFIG	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
STG_FIF_GL_DATA	Yes	No	No	No
FIF_GL_PUBLISH	Yes	Yes	No	Yes
FIF_GL_SETUP	Yes	No	No	No

Rolled Up Daily Stock Ledger Transactions (fifglndn2)

Module Name	Fifglndn2sqls.pls/fifglndn2sqlb.pls
Description	Interface to General Ledger of Rolled Up Transactions
Functional Area	Integration - General Ledger
Module Type	Integration
Module Technology	PLSQL
Catalog ID	RMS67
Wrapper Script	rmswrap_plsql.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program summarizes stock ledger data from the transaction staging table (IF_TRAN_DATA) based on the level of information required and writes it to the financial general ledger staging table. The transactions extracted are determined by the CODE_TYPE 'GLRT' (General Ledger Rolled Transactions). The written information can then be extracted by the financial applications. Stock ledger information may be rolled-up at department, class or subclass level. The level at which information is rolled-up to is determined by the system parameter GL_ROLLUP.

If transactions exist for which GL cross mappings do not exist, these will be logged in TRAN_DATA_ERRORS and a notification will be sent to the Finance Analyst user indicating that unmapped transactions exist. The TRAN_DATA_ERRORS table is available through the Data Access Schema, enabling users to view the errors and create the missing mappings. The fifgldn2 module will attempt to reprocess records from TRAN_DATA_ERRORS if mappings have been created. During the month-end processing run, if unmapped transactions still exist they will be posted into the clearing accounts as outlined below:

1. The cost value of the transaction will be posted into the Credit Clearance account [Credit Clearance Segment 1-10] associated with the location's set of books.
2. The cost value of the transaction will be posted into the Debit Clearance account [Debit Clearance Segment 1-10] associated with the location's set of books.
3. The retail value of the transaction will be posted into the Credit Clearance account [Credit Clearance Segment 1-10] associated with the location's set of books.
4. The retail value of the transaction will be posted into the Debit Clearance account [Debit Clearance Segment 1-10] associated with the location's set of books.

The month-end processing is typically executed on the End of Month (EOM) date plus Close Month After Days to allow for late-posted transactions to be attributed to the correct month. If on this date, clearing accounts are not defined and unresolved mappings continue to exist, then the month will remain open for an additional number of days specified in **Force Close Month After Additional Days**. After this duration, the fifgldn2 module will move the records-pending posting from TRAN_DATA_ERRORS to STG_FIF_GL_EOM_ERRORS and proceed to close the month without failure. The STG_FIF_GL_EOM_ERRORS table is available through DAS/RDS to enable extraction and processing of unposted transaction records.

Restart/Recovery

The logical unit of work is dependent on the level of rollup defined in the system options table. It can be department (department rollup), department/class (class rollup) or department/class/subclass (subclass rollup). The batch is multithreaded using the restart department view.

I/O Specification

Integration Type	Download from Merchandising
File Name	N/A
Integration Contract	IntCon000019 STG_FIF_GL_DATA table

Design Assumptions

N/A

Stage G/L Extracts (gl_extract.ksh)

Module Name	gl_extract.ksh
Description	Extraction of General Ledger transaction data from Merchandising and Sales Audit to be interfaced to third party GL/Financial system
Functional Area	Integration to General Ledger
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS495
Wrapper Script	rmswrap_shell_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch job will extract general ledger transaction data from Sales Audit and Merchandising into a file. Data to be extracted will be pulled off from the STG_FIF_GL_DATA table. Once the data is extracted into the file batch will purge the data from the table.

Restart/Recovery

N/A

I/O Specification

Integration Type	Extract from Merchandising
File Name	GL_EXTRACT_[#date].dat
Integration Contract	Na

Output File Layout

The output file is comma delimited with the following fields:

Record Name	Field Name
All records have the same structure	SET_OF_BOOKS_ID
	ACCOUNTING_DATE
	CURRENCY_CODE
	STATUS
	DATE_CREATED
	CREATED_BY
	ACTUAL_FLAG
	USER_JE_CATEGORY_NAME
	USER_JE_SOURCE_NAME
	CURRENCY_CONVERSION_DATE

Record Name	Field Name
	CURRENCY_CONVERSION_TYPE
	ACCT_SEGMENT1
	ACCT_SEGMENT2
	ACCT_SEGMENT3
	ACCT_SEGMENT4
	ACCT_SEGMENT5
	ACCT_SEGMENT6
	ACCT_SEGMENT7
	ACCT_SEGMENT8
	ACCT_SEGMENT9
	ACCT_SEGMENT10
	ENTERED_DR_AMOUNT
	ENTERED_CR_AMOUNT
	TRANSACTION_DATE
	REFERENCE1
	REFERENCE2
	REFERENCE3
	REFERENCE4
	REFERENCE5
	ATTRIBUTE1
	ATTRIBUTE2
	ATTRIBUTE3
	ATTRIBUTE4
	ATTRIBUTE5
	ATTRIBUTE6
	PERIOD_NAME
	CODE_COMBINATION_ID
	PGM_NAME
	ACCT_SEGMENT11
	ACCT_SEGMENT12
	ACCT_SEGMENT13
	ACCT_SEGMENT14
	ACCT_SEGMENT15
	ACCT_SEGMENT16
	ACCT_SEGMENT17
	ACCT_SEGMENT18
	ACCT_SEGMENT19
	ACCT_SEGMENT20
	REFERENCE_TRACE_ID
	PRIM_CURRENCY_CODE

Record Name	Field Name
	PRIM_ENTERED_DR_AMOUNT
	PRIM_ENTERED_CR_AMOUNT
	FIN_GL_SEQ_ID
	PROCESSED_FLAG

Design Assumptions

N/A

Tran Data Publication (BDI_TranData_Tx_PF_From_RMS_EOW_JOB)

This section describes the Tran Data Publication BDI.

Functional Area

Transactional Data

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of transactional data from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdimfpb.pls

BDI_MFP_SQL.TRAN_DATA_UP(0_error_message IN OUT
RTK_ERRORS.RTK_TEXT%TYPE, 0_control_id IN OUT
NUMBER, I_job_context IN VARCHAR2)

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising transaction tables/views.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Tran Data	Tran Data upload to BDI	TranData_Tx_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
TRAN_DATA_OUT	No	Yes	No	No
V_BDI_MFP_TRAN_DATA	Yes	No	No	No

Ordering and Inventory

Merchandising publishes purchase order and inventory-related for many other solution areas, including purchase orders, import details, invoices, available inventory, and other inventory related data. This section has been broken down into subsections for:

- [Purchasing](#)
- [Import Management](#)
- [Invoices](#)
- [Inventory](#)

Purchasing

Merchandising has scheduled integration for the following purchasing related data:

- [Download Contracts to Suppliers \(edidlcon\)](#)
- [Download Current & Future OTB by Subclass \(otbdnld\)](#)
- [Download Purchase Orders to Suppliers \(edidlord\)](#)
- [Download Summary of Outstanding Orders on OTB by Subclass \(otbdlord\)](#)
- [On Order Publication API \(BDI_OnOrder_Tx_PF_From_RMS_EOW_JOB\)](#)
- [Replenishment Item Location Publication API \(BDI_ReplItemLoc_Fnd_PF_From_RMS_JOB\)](#)

Download Contracts to Suppliers (edidlcon)

Module Name	edidlcon.pc
Description	Download Contracts to Suppliers
Functional Area	Contracts
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS45
Wrapper Script	rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

Contracts are defined in an Merchandising UI that writes to series of contracts database tables. This program is used to send this contract information to vendors. Only approved contracts

that are flagged as EDI contracts are processed by this batch program. The output file of this program contains all records for the supplier contract data which are in approved status.

Restart/Recovery

The logical unit of work for this program is set at the contract number. This program processes one contract number at a time.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000011

Output File Layout

Table 6-11 edidlcon.pc- File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File head descriptor	Char(5)	FHEAD	Describes file line type
	Line Number	Number(10)	0000000001	Sequential file line number
	Gentran ID	Char(4)	'DNCN'	Identifies which translation Gentran uses
	Current date	Char(14)	N/A	Indicates the date that the file was created in YYYYMMDDHH 24MISS format
THEAD	File head descriptor	Char(5)	THEAD	Describes file line type
	Line Number	Number(10)	N/A	Sequential file line number
	Transaction Number	Number(10)	N/A	Sequential transaction number
	Supplier	Number(10)	N/A	Indicates the supplier associated with the contract
	Contract Number	Number(6)	N/A	Indicates the Merchandising contract number
	Contract type	Char(1)	N/A	Type of contract. Valid types are A, B, C or D

Table 6-11 (Cont.) edidlcon.pc- File Layout

Record Name	Field Name	Field Type	Default Value	Description
TDETL	Department	Number(4)	N/A	Indicates the Merchandising department ID for which the contract applies
	Currency code	Char(3)	N/A	Indicates the currency code for the contract
	Total contract cost	Number(20)	N/A	Contains the total cost of the contract; includes 4 implied decimal places
	File record descriptor	Char(5)	TDETL	Describes file line type
	Line Number	Number(10)	N/A	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number
	Item Number Type	Char(6)	N/A	Indicates the type of item number is represented in the file. This corresponds to the item number type defined for items on ITEM_MASTER
	Item Number	Char(25)	N/A	Contains the unique ID for the item on the contract
	Ref Item Number Type	Char(6)	N/A	Indicates the item number type for the reference number corresponding to the item number
TDETL	Ref Item Number	Char(25)	N/A	Contains the unique ID for the reference number for the item
	Diff1	Char(120)	N/A	Contains the description of Diff1 for the item

Table 6-11 (Cont.) edidlcon.pc- File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Diff2	Char(120)	N/A	Contains the description of Diff2 for the item
	Diff3	Char(120)	N/A	Contains the description of Diff3 for the item
	Diff4	Char(120)	N/A	Contains the description of Diff4 for the item
	VPN	Char(30)	N/A	Vendor Product Number for the item
	Unit cost	Number(20)		Contains the cost of the item on the contract with 4 implied decimal places
	Ready Date	Char(14)		Date on which the items are to be provided by supplier. This field contains only values for contract types of 'A' or 'B'
	Ready Quantity	Number(20)		Quantity contracted with supplier with 4 implied decimal points. This field contains only values for contract types of 'A' or 'B'
	Location Type	Char(2)		Indicates the type of location on the contract - either 'ST' (store) or 'WH' (warehouse). This field contains only values for contract types of 'A' or 'B'

Table 6-11 (Cont.) edidlcon.pc- File Layout

Record Name	Field Name	Field Type	Default Value	Description
TTAIL	Location number	Number(10)		Contains a location on the contract. This field contains only values for contract types of 'A' or 'B'
	File Record descriptor	Char(5)	TTAIL	Describes file line type
	Line Number	Number(10)	N/A	Sequential file line number
FTAIL	Transaction number	Number(10)	N/A	Sequential transaction number
	File record descriptor	Char(5)	FTAIL	Marks the end of file
	Line number	Number(10)	N/A	Sequential file line number
	Number of lines	Number(10)	N/A	Number of lines in file not counting FHEAD and FTAIL

Design Assumptions

- This module should only be run if contracting is turned on in the system.

Download Current & Future OTB by Subclass (otbdnld)

Module Name	otbdnld.pc
Description	Download Current & Future OTB by Subclass
Functional Area	Open To Buy
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS130
Wrapper Script	rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program will extract current and future Open to Buy data from the OTB table in Merchandising and export it to a flat file for use by an external planning system. All records with an end of week date greater than or equal to today will be sent.

Restart/Recovery

The logical unit of work for the OTBDNLD module is department, class, subclass, and end-of-week date, with a recommended commit counter setting of 10,000. Each time the record counter equals the maximum recommended commit number, an application image array record will be written to the restart_start_array for restart/recovery if a fatal error occurs.

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000031

Output File Layout

Table 6-12 otbdnld.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File Type Record Descriptor	Char (5)	FHEAD	Identifies file record type
	File Line Sequence Number	Number (10)	N/A	Keeps track of the record's position in the file by line number
	File Type Definition	Char (4)	N/A	Identifies file as 'OTB Export'
	File Create Date	Char(14)	N/A	Date the file was created in YYYYMMDD format. Remaining 6 characters are blank
FDETL	File record descriptor	Char(5)	FDETL	Identifies file record type
	File Line Sequence Number	Number (10)		Keeps track of the record's position in the file by line number
	Transaction Set Control Number	Number(14)		Used to force unique file check
	Department	Number(4)		The ID number of a department
	Class	Number(4)		The ID number of a class within the department given

Table 6-12 (Cont.) otbdnld.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
	Subclass	Number(4)		The ID number of a subclass within the class given
	EOW Date	Date		The end of week date for the budgeted period. Format is 'YYYYMMDDH HMMSS'
	Week number	Number(2)		The week number in the month for the budgeted period
	Month number	Number(2)		The month number in the half for the budgeted period
	Half number	Number(5)		The half number for the budgeted period
	Cancel Amount	Number(20)		The total amount cancelled from orders of all order type for the budgeted period; value includes 4 implied decimal places
	N Approved Amount	Number(20)		The amount of approved non-basic (order type N/B) orders for the budgeted period; value includes 4 implied decimal places
	N Receipts Amount	Number(20)		The amount of non-basic (order type N/B) orders due in the budgeted period that have been received; value includes 4 implied decimal places

Table 6-12 (Cont.) otbdnld.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
	B Approved Amount	Number(20)		The amount of approved buyer-replenished basic (order type BRB) orders for the budgeted period; value includes 4 implied decimal places
	B Receipts Amount	Number(20)		The amount of buyer-replenished basic (order type BRB) orders due in the budgeted period that have been received; value includes 4 implied decimal places
	A Approved Amount	Number(20)		The amount of approved auto-replenished basic (order type ARB) orders for the budgeted period; value includes 4 implied decimal places
	A Receipts Amount	Number(20)		The amount of auto-replenished basic (order type ARB) orders due in the budgeted period that have been received; value includes 4 implied decimal places
FTAIL	File record descriptor	Char (5)	FTAIL	Identifies file record type
	File Line Sequence Number	Number (10)		Keeps track of the record's position in the file by line number

Table 6-12 (Cont.) otbdnld.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
	Number of lines	Number (10)		Total number of all transaction lines, not including file header and trailer

Design Assumptions

N/A

Download Purchase Orders to Suppliers (edidlord)

Module Name	edidlord.pc
Description	Download of Purchase Order from Merchandising to Suppliers
Functional Area	Purchase Order
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS46
Wrapper Script	rmswrap_multi_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

Orders created within the Oracle Retail system are written to a flat file if they are approved and marked as EDI orders. This module is used to write new and changed purchase order data to a flat file in the Oracle Retail standard format. The translation to EDI format is expected to take place via a 3rd party translation utility. The order revision tables and allocation revision tables are also used to ensure that the latest changes are being sent and to allow both original and modified values to be sent. These revision tables are populated during the online ordering process and the batch replenishment process whenever an order has been approved, and constitutes a history of all revisions to the order.

The program sums up all quantities to the physical warehouse level from the virtual warehouse level for an order, before writing it into the output file.

If shipments are to be pre-marked by the supplier for cross docking, then along with the order information: allocation, location and quantities are also sent.

If the backhaul type is specified as "Calculated", then the backhaul allowances will be calculated.

If the order contains pack items; hierarchical pack information is sent (this may include outer packs, inner packs, and fashion styles with associated pack templates as well as component item information).

If the order is a Drop Ship Customer Order (location is a non-stockholding store), the customer billing and delivery information will be written to the flat file.

Restart/Recovery

The logical unit of work for this program is set at the supplier level. Threading is performed by the supplier using the v_restart_supplier view.

Restart ability is implied because the program updates ordhead.edi_sent_ind as records are written out. The commit_max_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records.

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000012

Output File Layout Summary Across Versions

Version Number	Record Name	Length
1	File Header - FHEAD	34
	Transaction Order Header Information – TORDR	5827
	Transaction Order Item Information – TITEM	900
	Transaction Pack Component Information – TPACK	978
	Transaction Shipment Information – TSHIP	318
	Transaction Customer Order Information – TCUST	3777
	Transaction Trailer - TTAIL	35
	File Trailer - FTAIL	25
2	File Header - FHEAD	34
	Transaction Order Header Information – TORDR	5847
	Transaction Order Item Information – TITEM	900
	Transaction Pack Component Information – TPACK	978
	Transaction Shipment Information – TSHIP	318
	Transaction Customer Order Information – TCUST	3777
	Transaction Trailer - TTAIL	35
	File Trailer - FTAIL	25
3	File Header - FHEAD	34
	Transaction Order Header Information – TORDR	5848
	Transaction Order Item Information – TITEM	900
	Transaction Pack Component Information – TPACK	978
	Transaction Shipment Information – TSHIP	318
	Transaction Customer Order Information – TCUST	3777
	Transaction Trailer - TTAIL	35
	File Trailer - FTAIL	25

Output File Layout

Record Name	Field Name	Field Type	Default Value	Description	Added in version
FHEAD	Record descriptor	Char(5)	FHEAD	File head marker	
	Line id	Number(10)	0000000001	Unique line id	
	Translator id	Char(5)	DLORD	Identifies transaction type	
	File create date	Char(14)		Vdate in YYYYMMDDHH24MISS format	
TORDR	Record descriptor	Char(5)	TORDR	Order header information	
	Line id	Number(10)		Unique file line id	
	Transaction id	Number(10)		Unique transaction id	
	Order change type	Char(2)		'CH' (changed) or 'NW' (new)	
	Order number	Number(12)		Internal Oracle Retail order no	
	Supplier	Number(10)		Internal Oracle Retail supplier id	
	Vendor order id	Char(15)		External vendor_order_no (if available)	
	Order written date	Char(14)		Order created date in YYYYMMDDHH24MISS format	
	Original order approval date	Char(14)		Original order approval date in YYYYMMDDHH24MISS format	
	Old Currency Code	Char(3)		Old order currency_code (ISO standard)	
	New Currency Code	Char(3)		Changed order currency_code (ISO standard)	
	Old Shipment Method of payment	Char(2)		Old ship_pay_method	
	New Shipment Method of Payment	Char(2)		Changed ship_pay_method	
	Old Transportation Responsibility	Char(2)		Old fob_trans_res	
	Old Transportation Responsibility Description	Char(250)		Old fob_trans_res_desc	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
	New Transportation Responsibility	Char(2)		Changed fob_trans_res	
	New Trans. Resp. Description	Char(250)		New fob_trans_res_desc	
	Old Title Passage Location	Char(2)		Old fob_title_pass	
	New Title Passage Location	Char(2)		Changed fob_title_pass	
	Old Title Passage Description	Char(250)		Old fob_title_pass_desc	
	New Title Passage Description	Char(250)		Changed fob_title_pass_desc	
	Old not before date	Char(14)		Old not_before_date in YYYYMMDDHH24MISS format	
	New not before date	Char(14)		Changed not_before_date in YYYYMMDDHH24MISS format	
	Old not after date	Char(14)		Old not_after_date in YYYYMMDDHH24MISS format	
	New not after date	Char(14)		Changed not_after_date in YYYYMMDDHH24MISS format	
	Old Purchase type	Char(6)		Old Purchase type	
	New Purchase type	Char(6)		New Purchase type	
	Backhaul allowance	Char(20)		Backhaul allowance	
	Old terms description	Char(240)		Old terms description from terms table	
	New terms description	Char(240)		New terms description from terms table	
	Old pickup date	Char(14)		Old pickup date YYYYMMDDHH24MISS	
	New pickup date	Char(14)		New pickup date YYYYMMDDHH24MISS	
	Old ship method	Char(6)		Old ship method	
	New ship method	Char(6)		New ship method	
	Old comment description	Char(200)		Old comment description	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
	New comment description	Char(2000)		New comment description	
	Supplier DUNS number	Char(9)		Supplier DUNS number	
	Supplier DUNS location	Char(4)		Supplier DUNS location	
	Customer order number	Char(48)		Master customer order number from the Order Management System	
	Fulfillment order number	Char(48)		Master Fulfillment order number from the Order Management System	
	File ID	Char(20)		To support supplier pooling via the buyer worksheet, the recommended order quantities for the individual suppliers are linked together for logistical purposes while generating orders from the worksheet based on this identifier.	2
	Non-Self Invoice Indicator	Char(1)		To support vendor invoicing for Consignment POs, these POs will need to be communicated to the supplier. This will serve as an indicator to the supplier that the PO is being sent only for purposes of consignment invoicing and doesn't need to be processed.	3
TITEM	File record descriptor	Char(5)		TITEM	Item info
	Line id	Number(10)		Unique line id	
	Transaction id	Number(10)		Unique transaction id	
	Item Number Type	Char(6)		Item_number_type	
	Item	Char(25)		Item (For a pack item, this will be the pack number)	
	Old Ref Item Number type	Char(6)		Item_number_type for old ref_item	
	Old Ref Item	Char(25)		Old Ref_Item	
	New Ref Item Number type	Char(6)		Item_number_type for new ref_item	
	New Ref Item	Char(25)		Changed Ref_Item	
	Vendor catalog number	Char(30)		Supplier_item (VPN)	
	Free Form Description	Char(250)		Item_desc	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
TPACK	Supplier Diff 1	Char(120)		Supplier's diff 1	
	Supplier Diff 2	Char(120)		Supplier's diff 2	
	Supplier Diff 3	Char(120)		Supplier's diff 3	
	Supplier Diff 4	Char(120)		Supplier's diff 4	
	Pack Size	Number(12)		Supplier defined pack size * 10000 (4 implied decimal places)	
	Item_line_no	Number(10)		Indicates the detail item line number.	
	File record descriptor	Char(5)	TPACK	Pack component info	
	Line id	Number(10)		Unique line id	
	Transaction id	Number(10)		Unique transaction id	
	Pack id	Char(25)		Packitem_breakout.pack_no (same as item for the pack item)	
	Inner pack id	Char(25)		Inner pack identification	
	Pack Quantity	Number(12)		Packitem_breakout.pack_item_qty*10000 (4 implied decimal places)	
	Component Pack Quantity	Number(12)		Packitem_breakout.comp_pack_qty*10000 (4 implied decimal places)	
	Item Parent Part Quantity	Number(12)		Packitem_breakout.item_parent_pt_qty*10000 (4 implied decimal places)	
	Item Quantity	Number(12)		Packitem_breakout.item_qty*10000 (4 implied decimal places)	
	Item Number Type	Char(6)		Item number type	
	Item	Char(25)		Item	
	Ref Item Number Type	Char(6)		Ref_item_number_type	
	Ref Item	Char(25)		Ref_item	
	VPN	Char(30)		Supplier item (vpn)	
	Supplier Diff 1	Char(120)		Supplier's diff 1	
	Supplier Diff 2	Char(120)		Supplier's diff 2	
	Supplier Diff 3	Char(120)		Supplier's diff 3	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
TSHIP	Supplier Diff 4	Char(120)		Supplier's diff 4	
	Item Parent	Char(25)		Required when Pack Template is not NULL	
	Pack template	Number(8)		Pack template associated w/ style (packitem_breakout.pack_tmpl_id)	
	Template description	Char(250)		Description of pack template. sups_pack_tmpl_desc.suppack_desc	
	Record type	Char(5)	TSHIP	Describes the file record-shipment information	
	Line id	Number(10)		Unique file line number	
	Transaction id	Number(10)		Unique transaction number	
	Location type	Char(2)		'ST' store or 'WH' warehouse	
	Ship to location	Number(10)		Location value form ordloc (store or warehouse – For warehouse,if multichannel option is ON, physical warehouse value is taken from warehouse)	
	Old unit cost	Number(20)		Old unit cost*10000 (4 implied decimal places)	
	New unit cost	Number(20)		New unit cost*10000 (4 implied decimal places)	
	Old quantity	Number(12)		Old qty_ordered *10000 or qty_allocated*10000 (4 implied decimal places)	
	New quantity	Number(12)		Changed qty_ordered*10000 or qty_allocated*10000 (4 implied decimal places)	
	Old outstanding quantity	Number(12)		Old (qty_ordered-qty_received)*10000 or (qty_allocated-qty_transferred)*10000 for an allocation (4 implied decimal places)	
	New outstanding quantity	Number(12)		Changed qty_ordered-qty_received (4 implied decimal places)(or qty_allocated-qty_transferred, for an allocation)	
	Cancel code	Char(1)			
	Old cancelled quantity	Number(12)		Previous quantity cancelled (4 implied decimal places)	
	New cancelled quantity	Number(12)		Changed quantity cancelled (4 implied decimal places)	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
TCUST	Quantity type flag	Char(1)		'S'hip to 'A'locate	
	Store or warehouse indicator	Char(2)		'ST' (store) or 'WH' (warehouse)	
	Old x-dock location	Number(10)		Alloc_detail location (store or wh)	
	New x-dock location	Number(10)		Alloc_detail location (store or wh)	
	Case length	Number(12)		Case length (4 implied decimal places)	
	Case width	Number(12)		Case width (4 implied decimal places)	
	Case height	Number(12)		Case height (4 implied decimal places)	
	Case LWH unit of measure	Char(4)		Case LWH unit of measure	
	Case weight	Number(12)		Case weight (4 implied decimal places)	
	Case weight unit of measure	Char(4)		Case weight unit of measure	
	Case liquid volume	Number(12)		Case liquid volume (4 implied decimal places)	
	Case liquid volume unit of measure	Char(4)		Case liquid volume unit of measure	
	Location DUNS number	Char(9)		Location DUNS number	
	Location DUNS loc	Char(4)		Location DUNS loc	
	Old unit cost init	Number(20)		Old unit cost init (4 implied decimal places)	
	New unit cost init	Number(20)		New unit cost init (4 implied decimal places)	
	Item/loc discounts	Number(20)		Item/loc discounts (4 implied decimal places)	
	Record type	Char(5)	TCUST	Describes the file record-customer order information	
	Line id	Number(10)		Unique file line number	
	Transaction id	Number(10)		Unique transaction number	
	Delivery first name	Char(120)		First name for the delivery address on the order	
	Delivery phonetic first name	Char(120)		Phonetic first name for the delivery address on the order	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
	Delivery last name	Char(120)	)	Last name for the delivery address on the order	
	Delivery phonetic last name	Char(120)	)	Phonetic last name for the delivery address on the order	
	Delivery preferred name	Char(120)	)	Preferred name for the delivery address on the order	
	Delivery company name	Char(120)	)	Company name for the delivery address on the order	
	Delivery address Line 1	Char(240)	)	First line of the delivery address of the customer	
	Delivery address Line 2	Char(240)	)	Second line of the delivery address of the customer	
	Delivery address Line 3	Char(240)	)	Third line of the delivery address of the customer	
	Delivery county	Char(250)	)	County portion of the delivery address	
	Delivery city	Char(120)	)	City portion of the delivery address	
	Delivery state	Char(3)		State portion of the delivery address	
	Delivery country ID	Char(3)		Country portion of the delivery address	
	Delivery post	Char(30)		Postal code portion of the delivery address	
	Delivery jurisdiction	Char(10)		Jurisdiction code of the delivery country-state relationship	
	Delivery phone	Char(20)		Phone number in the delivery information	
	Billing first name	Char(120)	)	First name for the billing address on the order	
	Billing phonetic first name	Char(120)	)	Phonetic first name for the billing address on the order	
	Billing last name	Char(120)	)	Last name for the billing address on the order	
	Billing phonetic last name	Char(120)	)	Phonetic last name for the billing address on the order	
	Billing preferred name	Char(120)	)	Preferred name for the billing address on the order	
	Billing company name	Char(120)	)	Company name for the billing address on the order	
	Billing address Line 1	Char(240)	)	First line of the billing address of the customer	
	Billing address Line 2	Char(240)	)	Second line of the billing address of the customer	
	Billing address Line 3	Char(240)	)	Third line of the billing address of the customer	

Record Name	Field Name	Field Type	Default Value	Description	Added in version
	Billing county	Char(250)		County portion of the billing address	
	Billing city	Char(120)		City portion of the billing address	
	Billing state	Char(3)		State portion of the billing address	
	Billing country ID	Char(3)		Country portion of the billing address	
	Billing post	Char(30)		Postal code portion of the billing address	
	Billing jurisdiction	Char(10)		Jurisdiction code of the billing country-state relationship	
	Billing phone	Char(20)		Phone number in the billing information	
TTAIL	Record type	Char(5)	TTAIL	Describes file record – marks end of order	
	Line id	Number(10)		Unique file line id	
	Transaction id	Number(10)		Unique transaction id	
	#Lines in transaction	Number(10)		Number of lines in transaction	
FTAIL	Record type	Char(5)	FTAIL	Describes file record – marks end of file	
	Line id	Number(10)		Unique file line id	
	#lines	Number(10)		Total number of transaction lines in file (not including FHEAD and FTAIL)	

For a new order, the “old” fields should be blank. For a changed order, both old and new fields should hold values. If the value has changed, “old” values come from the revision tables for the latest revision before the current one (the last one sent), while new orders come from the ordering tables.

- FHEAD - REQUIRED: File identification, one line per file.
- TORDR - REQUIRED: Order level information, one line per order.
- TITEM - REQUIRED: Item description, multiple lines per order possible.
- TPACK - OPTIONAL: Pack contents, multiple lines per order possible. This line will be written only for pack items.
- TSHIP - REQUIRED: Ship to location and quantity, allocation location, multiple lines per item possible. Allocation information is optional on this line-will exist if premark_ind is 'Y'.
- TCUST - OPTIONAL: Customer order information, one line per order. This line will be written only for Drop Ship Customer Orders.
- TTAIL - REQUIRED: Order end, one line per order.
- FTAIL - REQUIRED: End of file marker, one line per file.

Design Assumptions

N/A

Download Summary of Outstanding Orders on OTB by Subclass (otbdlord)

Module Name	otbdlord.pc
Description	Download Summary of Outstanding Orders on OTB by Subclass
Functional Area	Open To Buy
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS13
Wrapper Script	rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program runs at the end of the half to delete rows from the OTB table that are at least one half old. The current and previous half's OTB data is retained. The number of days that OTB records are retained by Merchandising is not configurable via a system parameter.

Restart/Recovery

The logical unit of work for the otbdlord module is department/class/subclass. The `commit_max_ctr` field should be set to prevent excessive rollback space usage, and to reduce the overhead of the file I/O. The recommended commit counter setting is 10000 records. Each time the record counter equals the maximum recommended commit number, an application image array record will be written to the `restart_start_array` for restart/recovery if a fatal error occurs.

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000029

Output File Layout

Table 6-13 otbdlord.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type

Table 6-13 (Cont.) otbdlord.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
File Detail	File Line Sequence Number	Number(10)	N/A	Keeps track of the record's position in the file by line number
	File Type Definition	Char(4)	OOEX	Identifies file as 'OTB Out-standing Order Export'
	File Create Date	Char(14)	N/A	Date the file was created in YYYYMMDD format. Remaining six characters are blank.
	File Type Record Descriptor	Char(5)	FDETL	Identifies file record type
	File Line Sequence Number	Number(10)	N/A	Keeps track of the record's position in the file by line number
	Transaction Set Control Number	Number(14)	N/A	Sequence number used to force unique detail record check
	Department	Number(4)	N/A	The number of the department which contains the outstanding order quantity value
	Class	Number(4)	N/A	The number of the class which contains the outstanding order quantity value.
	Subclass	Number(4)	N/A	The number of the subclass which contains the outstanding order quantity value

Table 6-13 (Cont.) otbdlord.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
	N Outstanding Amt	Number(20)	N/A	The amount of outstanding non-basic orders (order type N/B) for past periods; value includes 4 implied decimal places
	B Outstanding Amt	Number(20)	N/A	The amount of outstanding buyer-replenished basic (order type BRB) orders for past periods; value includes 4 implied decimal places
	A Outstanding Amt	Number(20)	N/A	The amount of outstanding auto-replenished basic (order type ARB) orders for past periods; value includes 4 implied decimal places
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence Number	Number(10)	N/A	Keeps track of the record's position in the file by line number
	Control Number File Line Count	Control Number File Line Count Number(10)	N/A	Total number of all transaction lines, not including file header and trailer

Design Assumptions

N/A

On Order Publication API (BDI_OnOrder_Tx_PF_From_RMS_EOW_JOB)

This section describes the On Order Publication BDI.

Functional Area

Inventory Tracking

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of quantities On Order information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: **bdimfpb.pls**

```
BDI_MFP_SQL.ON_ORDER_UP(O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                        O_control_id      IN OUT NUMBER,
                        I_job_context     IN     VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Order tables/view.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
On Order	On Order upload to BDI	OnOrder_Tx_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
ON_ORDER_OUT	No	Yes	No	No
V_BDI_MFP_ON_ORDER	Yes	No	No	No

Replenishment Item Location Publication API (BDI_ReplItemLoc_Fnd_PF_From_RMS_JOB)

This section describes the Replenishment Item Location Publication BDI.

Functional Area

Item

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Replenishment Item Location information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: **bdiitemb.pls**

```
BDI_ITEM_SQL.REPL_ITEM_LOC_UP(O_error_message IN OUT VARCHAR2,
                                O_control_id    IN OUT NUMBER,
                                I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising REPL_ITEM_LOC table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Replenishment Item Location	Replenishment Item Location upload to BDI	ReplItemLoc_Fnd_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
REPL_ITEM_LOC_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
REPL_ITEM_LOC	Yes	No	No	No

Import Management

When using the Import Management features in Merchandising, there are several outbound integration processes that are available for customs entry and letter of credit functions. These integrations are only available if you are using not using Simplified Import Management (based on your system options configurations).

For additional information about import management, including detailed flow diagrams, see the *RTM Overview* white paper in the Merchandising Documentation Library (Doc ID: 1585843.1).

- [Download of Customs Entry Transactions to Brokers \(cednld\)](#)
- [Letter of Credit Amendment Download \(lcmdnld\)](#)
 - [SWIFT File Conversion – Letter of Credit Amendment \(lcmt707\)](#)
- [Letter of Credit Application Download \(lcadnld\)](#)
 - [SWIFT File Conversion - Letter of Credit Application \(lcmt700\)](#)

Download of Customs Entry Transactions to Brokers (cednld)

Module Name	cednld.pc
Description	Download of Customs Entry Transactions to Brokers
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS53
Wrapper Script	batch_cednld.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program is used to download custom entry information from the Merchandising database to brokers. Each night, this program reads all customs entry (CE) transactions that are in Sent status for a broker ID. These transactions are written to a flat file and the status is changed to Downloaded. One flat file is written per broker.

Restart/Recovery

The Logical Unit of Work for the program is a single row from the customs entry header table. Restart/Recovery will be used for init and commit.

Table based restart/recovery must be used. The commit max counter field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000050

Output File Layout

Table 6-14 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Identifier	Number(10)	Nine leading zeroes: 0000000001	ID of current line being processed by input file
	File Type Definition	Char(4)	CEDN	Identifies file as 'Customs Entry download'
	File Create Date	Date	Create date	Vdate in YYYYMMDDHH24MISS format
THEAD	File Type Descriptor	Char(5)	THEAD	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
	CE ID	Number(10)	ce_head.ce_id	N/A
	Entry No	Char (15)	ce_head.entry_no	N/A
	Entry Date	Char(14)	ce_head.entry_date	YYYYMMDDHH24MISS format
	Entry Status	Char(6)	ce_head.entry_status	N/A
	Entry Type	Char(6)	ce_head.entry_type	N/A
	Entry Port	Char(5)	ce_head.entry_port	N/A
	Summary Date	Char(14)	ce_head.summary date	YYYYMMDDHH24MISS format
	Broker ID	Char(10)	ce_head.broker_id	N/A
	Broker Ref. ID	Char(18)	ce_head.broker_ref_id	N/A
	File Number	Char(18)	ce_head.file_no	N/A
	Importer ID	Char(10)	ce_head.importer_id	N/A
	Import Country	Char(3)	ce_head.import_country_id	N/A
	Currency Code	Char(3)	ce_head.currency_code	N/A
	Exchange Rate	Number(20,10)	ce_head.exchange_rate*1000 0000000 (with 10 implied decimal places)	N/A

Table 6-14 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TSHIP	Bond Number	Char(18)	ce_head.bond_ no	N/A
	Bond Type	Char(6)	ce_head.bond_ type	N/A
	Surety Code	Char(6)	ce_head.surety_ _code	N/A
	Consignee ID	Char(10)	ce_head.consig nee_id	N/A
	Live Indicator	Char(1)	ce_head.live_in d	N/A
	Batch Number	Char(20)	ce_head.batch_ _no	N/A
	Entry Team	Char(3)	ce_head.entry_ team	N/A
	Liquidation Amount	Number(20,4)	ce_head.liquida tion_amt*10000 (4 implied decimal places)	N/A
	Liquidation Date	Date	ce_head.liquida tion_date	YYYYMMDDHH2 4MISS format
	Reliquidation Amount	Number(20,4)	ce_head.reliqui dation_amt*100 00 (4 implied decimal places)	N/A
	Reliquidation Date	Date	ce_head.reliqui dation_date	YYYYMMDDHH2 4MISS format
	Merchandise Loc	Char(40)	ce_head.merch andise_loc	N/A
	Location Code	Char(4)	ce_head.locatio n_code	N/A
	File Type Descriptor	Char(5)	TSHIP	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
	Vessel ID	Char(20)	ce_shipment.ve ssel_id	N/A
	Voyage Flt ID	Char(10)	ce_shipment.vo yage_flt_id	N/A
	Estimated Departure Date	Date	ce_shipment.es timated_depart _date	YYYYMMDDHH2 4MISS format
	Vessel SCAC Code	Char(6)	ce_shipment.ve ssel_scac_cod e	N/A
	Lading Port	Char(5)	ce_shipment.la ding_port	N/A

Table 6-14 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TORDI	Discharge Port	Char(5)	ce_shipment.di scharge_port	N/A
	Tran Mode ID	Char(6)	ce_shipment.tr an_mode_id	N/A
	Export Date	Date	ce_shipment.ex port_date	YYYYMMDDHH2 4MISS
	Import Date	Date	ce_shipment.im port_date	YYYYMMDDHH2 4MISS
	Arrival Date	Date	ce_shipment.ar rival_date	YYYYMMDDHH2 4MISS
	Export Country	Char(3)	ce_shipment.ex port_country_id	N/A
	Shipment Number	Number(10)	ce_shipment.sh ipment_no	N/A
	File Type Descriptor	Char(5)	TORDI	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
	Order Number	Number(8)	ce_ord_item.or der_no	N/A
	Item	Char (25)	ce_ord_item.ite m	N/A
	BL AWB ID	Char(30)	ce_ord_item.bl_ awb_id	'MULTI' – means multiple airway bills (otherwise a single airway bill will be retrieved)
	Invoice ID	Char(30)	ce_ord_item.inv oice_id	N/A
	Invoice Date	Date	ce_ord_item.inv oice_date	YYYYMMDDHH2 4MISS format
	Invoice Amount	Number(20,4)	ce_ord_item.inv oice_amt*1000 0 (4 implied decimal places)	N/A
	Currency Code	Char(3)	ce_ord_item.cu rrency_code	N/A
	Exchange Rate	Number(20,10)	ce_ord_item.ex change_rate*10 000000000 (10 implied decimal places)	N/A
	Manifest Item Quantity	Number(12,4)	ce_ord_item.m anifest_item_qt y*10000 (4 implied decimal places)	N/A

Table 6-14 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Manifest Item Quantity UOM	Char(4)	ce_ord_item.manifest_item_qty_uom	N/A
	Carton Quantity	Number (12,4)	ce_ord_item.carton_qty*10000 (4 implied decimal places)	N/A
	Carton Quantity UOM	Char(4)	ce_ord_item.carton_qty_uom	N/A
	Gross Weight	Number(12,4)	ce_ord_item.gross_wt*10000 (4 implied decimal places)	N/A
	Gross Weight UOM	Char(4)	ce_ord_item.gross_wt_uom	N/A
	Net Weight	Number(12,4)	ce_ord_item.net_wt*10000 (4 implied decimal places)	N/A
	Net Weight UOM	Char(4)	ce_ord_item.net_wt_uom	N/A
	Cubic	Number(12,4)	ce_ord_item.cubic*10000 (4 implied decimal places)	N/A
	Cubic UOM	Char(4)	ce_ord_item.cubic_uom	N/A
	Cleared Quantity	Number(12,4)	ce_ord_item.cleared_qty*10000 (4 implied decimal places)	N/A
	Cleared Quantity UOM	Char(4)	ce_ord_item.cleared_qty_uom	N/A
	In Transit Number	Char(15)	ce_ord_item.in_transit_no	N/A
	In Transit Date	Date	ce_ord_item.in_transit_date	YYYYMMDDHH24MISS format
	Rush Indicator	Char(1)	ce_ord_item.rush_ind	N/A
	Related Indicator	Char(1)	ce_ord_item.related_ind	N/A
	Tariff Treatment	Char(10)	ce_ord_item.tariff_treatment	N/A
	Ruling Number	Char(10)	ce_ord_item.ruling_no	N/A
	Do Number	Char(10)	ce_ord_item.do_no	N/A

Table 6-14 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TBLAW	Do Date	Date	ce_ord_item.do_date	YYYYMMDDHH24MISS format
	Manufacture ID	Char(18)	sup_import_attr.mfg_id	N/A
	File Type Descriptor	Char(5)	TBLAW	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
TCONT	BL AWB ID	Char(30)	Transportation.bl_awb_id	N/A
	File Type Descriptor	Char(5)	TCONT	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
TLICV	Container ID	Char(20)	Transportation.container_id	N/A
	Container SCAC Code	Char(6)	Transportation.container_scac_code	N/A
	File Type Descriptor	Char(5)	TLICV	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
	License/Visa Type	Char(6)	ce_lic_visa.license_visa_type	N/A
	License/Visa ID	Char(30)	ce_lic_visa.license_visa_id	N/A
	License/Visa Quantity	Number(12,4)	ce_lic_visa.license_visa_qty*10000 (4 implied decimal places)	N/A
	License/Visa Quantity UOM	Char(4)	ce_lic_visa.license_visa_qty_uom	N/A
	Quota Category	Char (6)	ce_lic_visa.quota_category	N/A
	Net Weight	Number(12,4)	ce_lic_visa.net_weight*10000 (4 implied decimal places)	N/A
	Net Weight UOM	Char(4)	ce_lic_visa.net_weight_uom	N/A
	Holder ID	Char(18)	ce_lic_visa.holder_id	N/A

Table 6-14 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TCHRG	File Type Descriptor	Char(5)	TCHRG	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
	Sequence Number	Number(6)	ce_charges.seq_no	N/A
	Pack Item	Char(25)	ce_charges.pack_item	N/A
	HTS	Char(10)	ce_charges.hts	N/A
	Effect From Date	Date	ce_charges.effect_from	YYYYMMDDHH24MISS format
	Effect To Date	Char(14)	ce_charges.effect_to	YYYYMMDDHH24MISS format
	Component ID	Date	ce_charges.component_id	N/A
	Component Rate	Number(20,4)	ce_charges.component_rate*10000 (4 implied decimal places)	N/A
	Per Count UOM	Char(3)	ce_charges.per_count_uom	N/A
TMDOC	Component Value	Number(20,4)	ce_charges.component_value * 10000 (4 implied decimal places)	N/A
	File Type Descriptor	Char(5)	TMDOC	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file
	Doc_id	Number(6)	Missing_doc.doc_id	N/A
FTAIL	Received_date	Date	Missing_doc.received_date	YYYYMMDDHH24MISS format
	File Type Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Identifier	Number(10)	Incremented internally	ID of current line being processed by input file.
	File Record Counter	Number(10)	Determined internally	Number of records/ transactions processed in current file (only records between head & tail)

Design Assumptions

N/A

Letter of Credit Amendment Download (lcmdnld)

Module Name	lcmdnld.pc
Description	Letter of Credit Amendment Download
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS56
Wrapper Script	rmswrap_dnld_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

lcmdnld.pc downloads amended letter of credit information to a bank, in the S.W.I.F.T. format.

Online user actions flag LCs for download by writing to the LC_DOWNLOAD table.

Restart/Recovery

Restart/recovery for this program is set up at the lc_ref_id level. The recommended commit counter setting is 1000 records (subject to change based on experimentation).

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integratin Contract	IntCon000053

Output File Layout

Table 6-15 File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Sequence Number	Number(10)	Line number in file	Keeps track of the record's position in the file by line number
	File Type Definition	Char(4)	LCAM	Identifies file as 'Letter of Credit Amendment'

Table 6-15 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
Transaction Header	File Create Date	Char(14)	Create date	Current date, formatted to 'YYYYMMDDHH24MISS'
	Filetype Record descriptor	Char(5)	THEAD	Identifies file record type
	File Line Sequence Number	Number (10)	Line number in file	Keeps track of the record's position in the file by line number
	Transaction Set Control Number	Number (10)	Sequence number	Used to force unique file check
	Issuing Bank	Char(10)	lc_head.issuing_bank	Used to sort the LCs into individualized bank SWIFT formatted files (using another program) – bank where LC application is headed
	Issuing Bank Name	Char(240)	partner.partner_desc	The description from the partner table where partner_id = issuing_bank and partner_type = 'BK'
	Issuing Bank Address 1	Char(240)	addr.add_1	Mandatory line of address
	Issuing Bank Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Issuing Bank Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Issuing Bank City	Char(120)	addr.city	City bank located in
	Issuing Bank State	Char(3)	addr.state	State, if applicable, where bank located in
	Issuing Bank Post Code	Char(30)	addr.post	Post code, if applicable, where bank located in
	Issuing Bank Country	Char(3)	addr.country_id	Country bank located in
	Letter of Credit	Number (8)	lc_detail.lc_ref_id	The LC_REF_ID off the LC_DETAIL table
	Bank Letter of Credit ID	Char(16)	lc_head.bank_lc_id	The BANK_LC_ID off the LC_HEAD table
	Currency Code	Char(3)	lc_head.currency_code	The CURRENCY_CODE off the LC_HEAD table
	Date of Issue/ Transfer of the Credit	Char(14)	lc_head.confirmed_date	Date the Issuing Bank thinks is the date of issue—when it was officially confirmed, formatted to 'YYYYMMDDHH24MISS'

Table 6-15 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
Transaction Detail	Current Amount of LC	Number (20,4)	N/A	This amount will be calculated in the get_current_amount() function and will be the net amount of the LC calculated only using amendments that have been downloaded. Normally, the net amount is calculated using amendments in the 'D'ownloaded status
	Beneficiary	Number (10)	lc.head.beneficiary	Party in favor of which the LC is being issued
	Beneficiary Name	Char(240)	sup.s.sup_name	Beneficiary (supplier) name from the SUPS table
	Beneficiary Address 1	Char(240)	addr.add_1	Mandatory line of address
	Beneficiary Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Beneficiary Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Beneficiary City	Char(120)	addr.city	City beneficiary located in
	Beneficiary State	Char(3)	addr.state	State, if applicable, where beneficiary located in
	Beneficiary Post Code	Char(30)	addr.post	Post code, if applicable, where beneficiary located in
	Beneficiary Country	Char(3)	addr.country_id	Country beneficiary located in
	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type
	File Line Sequence Number	Number (10)	line number in file	Keeps track of the record's position in the file by line number
	Transaction Set Control Number	Number (10)	sequence number	Used to force unique file check
Transaction Detail	Amendment Number	Number (8)	lc_amendments.amend_no	Holds the amendment number for the amendment
	Order_no	Number (8)	lc_amendments.order_no	Order_no, if applicable, that is attached to the LC that is being amended
	Item	Char(25)	lc_amendments.item	Item being amended, either a Style or Staple sku

Table 6-15 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Value Being Amended	Char(6)	lc_amendment s.amended_value	LC Field being amended. Can be any of the following code_types: CODE CODE_DESC AI Add Item AO Add PO ARQD Add Reqd Doc. C Cost ED Expiration Date ESD Earliest Ship Date LSD Latest Ship Date NA Net Amount ND Negotiation Days OC Origin Country OQ Order Quantity PE Place of Expiry PRT Presentation Terms PSF Partial Ship Flag RI Remove Item RO Remove PO RRQD Remove Reqd Doc TFF Transferable Flag TSF Transshipment Flag
	Value Being Amended Description	Char(40)	code_detail.co de_desc	The Value Being Amended decoded (see the above list). Will possibly be used when printing to the SWIFT file MT 707 for clarity
	Original Value of Amended Field	Char(45)	lc_amendment s.original_value	Current value of field that is being amended
	New Value of Amended Field	Char (2000)	lc_amendment s.new_value	New value of the field that is being amended
	Description of New Value	Char(40)	code_detail.co de_desc	The new value decoded (or fetched from a table, as in the origin_country case)– only applicable to the following amended values: place of expiry, title_pass_location, origin_country, presentation terms, purchase type
	Sign	Char(1)	N/A	If the effect is negative it will be “-“ if the effect is positive it will be “ “
	Effect	Number (20,4)	lc.amendment s.effect	Effect that amendment will have on LC if amendment to change qty or cost of a PO or amount of LC itself

Table 6-15 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
Transaction Text	Date of Amendment	Char(14)	Lc_amendments.accept_date	Date on which Issuing Bank (or issuing party, in this case the retailer) considers the credit as being amended, formatted to 'YYYYMMDD HH24MISS'
	File Type Record Descriptor	Char(5)	TTEXT	Identifies file record type
	File Line Sequence Number	Number (10)	line number in file	Keeps track of the record's position in the file by line number
	Transaction Set Control Number	Number (10)	sequence number	Used to force unique file check
Transaction Trailer	Amendment Text	Char (2000)	text description	A text description of the individual amendment (for each TDETL line of the output file) built by the package LC_AMEND_SQL.AMEND_TEXT.
	File Type Record Descriptor	Char (5)	TTAIL	Identifies File Record Type
	File Line Sequence Number	Number (10)	Line Number in file	ID of current line being created for output file
	Transaction set control number	Number (10)	Sequence number	Used to force unique file check
File Trailer	Transaction detail line count	Number (10)	ID of current line being created for output file	Sum of the detail lines within a transaction
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence Number	Number (10)	line number in file	Keeps track of the record's position in the file by line number
	Control Number File Line Count	Number (10)	total detail lines	Sum of all transaction lines, not including the file header and trailer

Letter of Credit Application Download (lcadnld)

Module Name	Lcadnld.pc
Description	Letter of Credit Application Download
Functional Area	Retail Trade Management

Module Type	Integration
Module Technology	ProC
Catalog ID	RMS57
Wrapper Script	rmswrap_dnlld_in.ksh

Schedule

See Oracle Merchandising Batch Schedule.

Design Overview

Lcadnld sends letter of credit (LC) applications to partner banks. Online user actions flag LCs for download by writing to the LC_DOWNLOAD table.

Restart/Recovery

Restart/recovery for this program is set up at the lc_ref_id level. The recommended commit counter setting is 10000 records (subject to change based on experimentation).

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integratin Contract	IntCon000052

Output File Layout

Table 6-16 File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Identifier	Number(10)	line number in file	ID of current line being created for output file
	File Type Definition	Char(4)	LCAP	Identifies file as 'Letter of Credit Application'
	File Create Date	Char(14)	create date	Current date, formatted to 'YYYYMMDDHH24MISS'
File Detail	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file.
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Issuing Bank	Char(10)	lc_head.issuing_bank	Used to sort the LCs into individualized bank SWIFT formatted files (using another program) - bank where LC application is headed
	Issuing Bank Name	Char(240)	partner.partner_desc	The description from the partner table where partner_id = issuing_bank and partner_type = 'BK'
	Issuing Bank Address 1	Char(240)	addr.add_1	Mandatory line of address
	Issuing Bank Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Issuing Bank Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Issuing Bank City	Char(120)	addr.city	City bank located in
	Issuing Bank State	Char(3)	addr.state	State, if applicable, where bank located in
	Issuing Bank Post Code	Char(30)	addr.post	Post code, if applicable, where bank located in
	Issuing Bank Country	Char(3)	addr.country_id	Country bank located in
	Advising Bank	Char(10)	lc_head.advising_bank	Used to sort the LCs into individualized bank SWIFT formatted files (using another program) - bank where LC application is headed
	Advising Bank Name	Char(240)	Partner.partner_desc	The description from the partner table where partner_id = advising_bank and partner_type = 'BK'

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Advising Bank Address 1	Char(240)	Addr.add_1	Mandatory line of address
	Advising Bank Address 2	Char(240)	Addr.add_2	Non-mandatory line of address (can be null)
	Advising Bank Address 3	Char(240)	Addr.add_3	Non-mandatory line of address (can be null)
	Advising Bank City	Char(120)	Addr.city	City bank located in
	Advising Bank State	Char(3)	Addr.state	State, if applicable, where bank located in
	Advising Bank Post Code	Char(30)	Addr.post	Post code, if applicable, where bank located in
	Advising Bank Country	Char(3)	Addr.country_id	Country bank located in
	Letter of Credit	Number(8)	lc_head.lc_ref_id	The LC_REF_ID off the LC_HEAD table
	Form Type	Char(6)	lc_head.form_type	The level of detail that the LC will send to the issuing bank
	Form Type Description	Char(40)	code_detail.code_desc	Describes the form type: Long or Short
	Letter of Credit Type	Char(6)	lc_head.lc_type	Describes the form type: Long or Short
	Letter of Credit Type Description	Char(40)	code_detail.code_desc	Describes the LC type: Master, Normal, Revolving
	Form of Letter of Credit – I	Char(1)	sup_import_attr.revocable_ind	The REVOCABLE_IND from the SUP_IMPORT_ATTR table
	Form of Letter of Credit – II	Char(1)	lc_head.transferable_ind	Indicates if LC transferable
	Application Date	Char(14)	lc_head.application_date	Date the LC is created within Import Management/ Merchandising, formatted to 'YYYYMMDD HH24MISS'

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Expiration Date	Char(14)	lc_head.expiration_date	The date the LC expires, formatted to 'YYYYMMDD HH24MISS'
	Place of Expiry	Char(6)	lc_head.place_of_expiry	Code for the place the LC will expire
	Place of Expiry Description	Char(40)	desc is retrieved through a decode	The description of the place the LC will expire
	Applicant	Char(10)	lc_head.applicant	Party on whose behalf the LC is being issued
	Applicant Name	Char(240)	partner.partner_desc	The description from the partner table where partner_id = applicant and partner_type = 'AP'
	Applicant Address 1	Char(240)	addr.add_1	Mandatory line of address
	Applicant Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Applicant Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Applicant City	Char(120)	addr.city	City applicant located in
	Applicant State	Char(3)	addr.state	State, if applicable, where applicant located in
	Applicant Post Code	Char(10)	addr.post	Post code, if applicable, where applicant located in
	Applicant Country	Char(3)	addr.country_id	Country applicant located in
	Beneficiary	Number(10)	lc.head.beneficiary	Party in favor of which the LC is being issued
	Beneficiary Name	Char(240)	supp.sup_name	Beneficiary (supplier) name from the SUPS table

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Beneficiary Address 1	Char(240)	addr.add_1	Mandatory line of address
	Beneficiary Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Beneficiary Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Beneficiary City	Char(120)	addr.city	City beneficiary located in
	Beneficiary State	Char(3)	addr.state	State, if applicable, where beneficiary located in
	Beneficiary Post Code	Char(30)	addr.post	Post code, if applicable, where beneficiary located in
	Beneficiary Country	Char(3)	addr.country_id	Country beneficiary located in
	Currency Code	Char(3)	lc_head.currency_code	The country of origin for the orders on the LC
	Exchange Rate	Number (20,10)	lc_head.exchange_rate	Exchange_rate to convert LC currency to Merchandising currency
	Origin Country ID	Char(3)	lc_head.origin_country_id	Origin country of the orders associated with the LC
	Presentation Terms	Char(6)	lc_head.presentation_terms	Code for the terms of presentation
	Presentation Terms Description	Char(40)	desc is retrieved through a decode	Description of the terms of presentation
	Purchase Type	Char(6)	lc_head.purchase_type	Code for the purchase type
	Purchase Type Description	Char(40)	desc is retrieved through a decode	Description of the purchase type
	Advice Method	Char(6)	lc_head.advice_method	Code for the advice method

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Advice Method Description	Char(40)	desc is retrieved through a decode	Description of the advice method (eg. Full Wire, Mail, and so on)
	Issuance	Char(6)	lc_head.issuance	Code for the issuance
	Issuance Description	Char(40)	desc is retrieved through a decode	Description of the issuance (for example Cable, Telex, and so on)
	Amount Type	Char(6)	lc_head.amount_type	If 'E'xact, then amount must be exact, if 'A'pproximate then amount can be within variance percent
	Amount Type Description	Char(40)	desc is retrieved through a decode	Description of amount_type
	Amount	Number (20,4)	lc_head.amount	The total amt of the Letter of Credit
	Variance Percent	Number (12,4)	lc_head.variance_pct	Allowed currency variance percent for the LC
	Specification	Char(6)	lc_head.specification	Code for any condition for the credit, such as, "maximum", and so on
	Specification Description	Char(40)	desc is retrieved through a decode	Description of condition for the credit, such as, "maximum", and so on
	Credit Available With	Char(10)	lc_head.credit_avail_with	Code for bank with which credit is available
	Credit With Bank Name	Char(40)	partner.partner_desc	The description from the partner table where partner_id = credit_avail_with and partner_type = 'BK'
	Credit With Address 1	Char(240)	addr.add_1	Mandatory line of address

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Credit With Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Credit With Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Credit With City	Char(120)	addr.city	City creditor located in
	Credit With State	Char(3)	addr.state	State, if applicable, where creditor located in
	Credit With Post Code	Char(30)	addr.post	Post code, if applicable, where creditor located in
	Credit With Country	Char(3)	addr.country_id	Country creditor located in
	Drafts At	Char(6)	lc_head.drafts_at	Specifies the terms of the drafts to be drawn under the LC
	Drafts At Description	Char(40)	desc is retrieved through a decode	Description of the terms of the drafts to be drawn under the LC
	Drawee	Char(10)	lc_head.paying_bank	Identifies drawee of drafts to be drawn under LC (paying bank)
	Drawee Name	Char(240)	partner.partner_desc	The description from the partner table where partner_id = paying_bank and partner_type = 'BK'
	Drawee Address 1	Char(240)	addr.add_1	Mandatory line of address
	Drawee Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Drawee Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Drawee City	Char(120)	addr.city	City bank located in

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Drawee State	Char(3)	addr.state	State, if applicable, where bank located in
	Drawee Post Code	Char(30)	addr.post	Post code, if applicable, where bank located in
	Drawee Country	Char(3)	addr.country_id	Country bank located in
	Negotiating Bank	Char(10)	lc_head.negotiating_bank	Identifies the negotiating bank
	Negotiating Bank Name	Char(240)	partner.partner_desc	The description from the partner table where partner_id = negotiating_bank and partner_type = 'BK'
	Negotiating Bank Address 1	Char(240)	addr.add_1	Mandatory line of address
	Negotiating Bank Address 2	Char(240)	addr.add_2	Non-mandatory line of address (can be null)
	Negotiating Bank Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Negotiating Bank City	Char(120)	addr.city	City bank located in
	Negotiating Bank State	Char(3)	addr.state	State, if applicable, where bank located in
	Negotiating Bank Post Code	Char(30)	addr.post	Post code, if applicable, where bank located in
	Negotiating Bank Country	Char(3)	addr.country_id	Country bank located in
	Confirming Bank	Char(10)	lc_head.confirming_bank	
	Confirming Bank Name	Char(240)	partner.partner_desc	Identifies the confirming bank
	Confirming Bank Address 1	Char(240)	addr.add_1	The description from the partner table where partner_id = confirming_bank and partner_type = 'BK'
	Confirming Bank Address 2	Char(240)	addr.add_2	Mandatory line of address

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Confirming Bank Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Confirming Bank City	Char(120)	addr.city	Non-mandatory line of address (can be null)
	Confirming Bank State	Char(3)	addr.state	City bank located in
	Confirming Bank Post Code	Char(30)	addr.post	State, if applicable, where bank located in
	Confirming Bank Country	Char(3)	addr.country_id	Post code, if applicable, where bank located in
	Transferring Bank	Char(10)	lc_head.transferring_bank	Country bank located in
	Transferring Bank Name	Char(240)	partner.partner_desc	Identifies the transferring bank
	Transferring Bank Address 1	Char(240)	addr.add_1	The description from the partner table where partner_id = transferring_bank and partner_type = 'BK'
	Transferring Bank Address 2	Char(240)	addr.add_2	Mandatory line of address
	Transferring Bank Address 3	Char(240)	addr.add_3	Non-mandatory line of address (can be null)
	Transferring Bank City	Char(120)	addr.city	Non-mandatory line of address (can be null)
	Transferring Bank State	Char(3)	addr.state	City bank located in
	Transferring Bank Post Code	Char(30)	addr.post	State, if applicable, where bank located in
	Transferring Bank Country	Char(3)	addr.country_id	Post code, if applicable, where bank located in
	Partial Shipment Indicator	Char(1)	lc_head.partial_ship_ind	Country bank located in
	Transshipment Indicator	Char(1)	lc_head.transshipment_ind	Indicates whether goods covered by LC can be partially shipped or not

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Fob Title Pass	Char(6)	lc_head.fob_title_pass	Indicates whether goods can be transferred to another vessel midway through the voyage
	Fob Title Pass Decode	Char(40)	desc is retrieved through a decode	Indicates where the title for goods is passed from the vendor to the purchaser
	Fob Title Pass Description	Char(250)	lc_head.ob_title_pass_desc	Decode of where the title for goods is passed from the vendor to the purchaser
	Transportation to	Char(5)	lc_head.transportation_to	Describes the FOB_TITLE_PASS - could be city name and so on
	Transportation to description	Char(150)	outloc.outloc_desc	Transportation to location
	With Recourse Indicator	Char(1)	lc_head.with_recourse_ind	Description of transportation to location
	Latest Shipment Date	Char(14)	lc_head.latest_ship_date	Indicates conditional payment on the part of the bank as instructed by the buyer
	Earliest Shipment Date	Char(14)	lc_head.earliest_ship_date	Latest ship date for all Pos included in the LC, formatted to 'YYYYMMDD HH24MISS'
	Letter of Credit Negotiation Days	Number(3) replaces x in the string "DOCUMENTS TO BE PRESENTED WITHIN x DAYS AFTER ISSUANCE OF THE SHIPPING DOCUMENTS BUT WITHIN THE VALIDITY OF THIS CREDIT"	lc.head.lc_neg_days	The number of days to negotiate documents

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Bank's LC reference id	Number(8)	lc_head.bank_lc_id	Bank's LC ref id
	File Type Record Descriptor	Char(5)	THDCM	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check
	Header Level Comments	Char(2000)	lc_head.comments	Holds any comments that you added to the Letter of Credit.
	File Type Record Descriptor	Char(5)	TDOCS	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check
	Swift Tag	Char(6)	doc.swift_tag	Identifies individual document types that can be associated with an LC
	Document ID	Number(6)	req_doc.doc_id	Uniquely identifies the individual documents associated with an LC
	Body Text	Char(2000)	req_doc.doc_text	Documents associated with a given LC Description of Goods and Services OR Documents Required OR Additional Conditions OR Narrative
	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check
	Order Number	Number(8)	lc_detail.order_no	PO associated with the LC
	Item	Char(25)	lc_detail.item	Item on the PO - item is rolled up to the item_level of 1, if possible
	Cost	Number (20,4)	lc_detail.cost	If form_type = 'S'hort then cost is the total cost of the order; if the form_type = 'L'ong then the cost is the unit cost of the item
	Quantity	Number (12,4)	lc_detail.qty	Total qty of the item for the order on the LC
	Standard UOM	Char(4)	Item_master.standard_uom	Standard unit of measure of the quantity of the item for the order on the LC
	Earliest Ship Date	Char(14)	lc_detail.earliest_ship_date	The earliest date an order on the LC can be shipped, formatted to 'YYYYMMDDHH24MISS'
	Latest Ship Date	Char(14)	lc_detail.latest_ship_date	The latest date an order on the LC can be shipped, formatted to 'YYYYMMDDHH24MISS'
	item description	Char(250)	Item_master.description	Item's description
	File Type Record Descriptor	Char(5)	TMERC	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check

Table 6-16 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Merchandise Description	Char(2000)	lc_detail.merch_desc	Contains the merchandise description of the field.
	File Type Record Descriptor	Char(5)	TDTCM	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check
	Detail Level Comments	Char(2000)	lc_detail.comments	Holds any comments that you added to the Letter of Credit detail record.
File Trailer	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type
	File Line Sequence Number	Number(10)	line number in file	ID of current line being created for output file
	Transaction Set Control Number	Number(10)	sequence number	Used to force unique file check
	Transaction detail line count	Number(10)	ID of current line being created for output file	Sum of the detail lines within a transaction
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Identifier	Number(10)	Sequential number Created by program.	ID of current line being created for output file.
	File Record Counter	Number(10)	N/A	Number of records/ transactions processed in current file (only records between head & tail)

SWIFT File Conversion – Letter of Credit Amendment (lcmt707)

Module Name	lcmt707
Description	SWIFT File Conversion – Letter of Credit Amendment

Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	Perl
Catalog ID	RMS137
Wrapper Script	rmswrap_perl.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This Perl script converts the Oracle retail standard interface file format for Amendments to Letters of Credit download to the corresponding S.W.I.F.T file format (MT 707). The input file for this Perl script is the output of the lcmdnld.pc Merchandising batch.

I/O Specification

Integration Type	Download to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000053 (input) IntCon000138 (output)

Output

The SWIFT MT 707 output file should be in the following format:

- Most output fields are contained in their own line (or 3-4 line for addresses).
- Each amendment consists of only one part, the MT 707. There may be several MT 707s at any given time associated to an LC because they are grouped by amendment number at the time of creation. All TDETL records with the same amend_no will be grouped together in one MT 707.
- Each record starts with a colon and a SWIFT field identifier, followed by another colon: for example, ':40A:':
- Each amendment is separated by a line with only the ASCII 3 symbol (a heart) on it.

Logic Setup:

The input file will be in standard Merchandising file format. It will potentially have numerous TDETL lines per each THEAD line. There may be numerous TDETL records for one amendment. MT 707 will write one record for each amendment, so if there are multiple TDETL records they need to be combined. There is one TTEXT for each TDETL.

There are three values that need to be calculated. 32B, 33B, 34B. 32B is the total increment or the sum of the positive effect values for each amendment. 33B is the total decrement or the sum of all the negative effect values for each amendment. 32B and 33B are separate totals for each amendment. 34B is the total difference, so it is the sum of the total increment and total decrement. 34B is not just for one amendment though; it is for all amendments of a THEAD record, so this total will run through each TDETL in a THEAD.

For example: if the input file contains:

- THEAD

- TDETL amendment 1, effect +1000
- TTEXT
- TDETL amendment 1, effect +500
- TTEXT
- TDETL amendment 2, effect -2500
- TTEXT
- TDETL amendment 3, effect +4000
- TTEXT
- TDETL amendment 3, effect -1000
- TTEXT
- TDETL amendment 3, effect +500
- TTEXT
- TDETL amendment 4, effect -1000
- TTEXT
- TDETL amendment 4 , effect –2500
- TTEXT
- TTAIL

32B for amendment 1 = 1500

33B for amendment 1 = 0

34B for amendemnt 1 = 1500

32B for amendment 2 = 0

33B for amendment 2 = 2500

34B for amendemnt 2 = -1000

32B for amendment 3 = 4500

33B for amendment 3 = 1000

34B for amendemnt 3 = 4500

32B for amendment 4 = 0

33B for amendment 4 = 3500

34B for amendemnt 4 = 1000

Examples of how individual lines of the M T 707 should look:

APPLICANT:
 OPERATOR:
 OPERATION DATE:
 OPERATION TIME:
 TEST KEY:
 BATCH TOTAL:
 SEGMENT TOTAL:
 MT/PRIORITY: 707 02
 :27:1/1
 :20:10001981
 :21:1981
 :52D:Bank One
 100 Bank One Way
 Columbus ,OH 41984 US
 :31C:990204
 :30:990204

```
:26E:1
:59:David Fashion Creations P/L Pack
Wholesale Division
109 Ackland St.
St. Kilda      ,VA  30280-1234 US
:32B:USD500,0
:33B:USD0,0
:34B:USD500,0
:79:Letter of Credit: has been changed from 25 to 30
for Style 10049369, resulting in an effect of 500
(USD).
```

The layout of the S.W.I.F.T MT 707 (Amendment to a Documentary Credit) file is as follows:

SWIFT I.D. DATA TYPE CODES (refer to SWIFT User Handbook – Standards General Information – October 1998 release for formatting information):

① Note

The field lengths and types in the Oracle Retail Standard Download Format of the MT 707 are important because sometimes they are different from the information that is being placed in them and the fields may have to be truncated, rounded, and so on.

There is always a new line (nl) after every individual SWIFT ID (and there may be more than one line within an individual field (example 59 - Beneficiary, four lines to hold address information)).

In some situations, certain fields will be blank. These fields should be skipped over. In other words, no blank line or tag should be printed indicating the field is blank. Simply ignore it.

SWIFT File Conversion - Letter of Credit Application (lcmt700)

Module Name	lcmt700
Description	SWIFT File Conversion – Letter of Credit Application
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	Perl
Catalog ID	RMS136
Wrapper Script	rmswrap_perl.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This Perl script will convert the standard Merchandising flat file into the bank specific S.W.I.F.T. MT 700 output files. The input file for this Perl script is the output of the lcadnld.pc Merchandising batch. One output file will be created for each issuing bank in the lcadnld.pc output file.

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000052 (input) IntCon000137 (output)

Output

All files layouts input and output the SWIFT MT 700. The output file should be in the following format:

- Most output fields are contained in their own line (or 3-4 line for addresses).
- Each application consists of four parts, one MT 700 and three MT 701s, which are ordered through the Sequence of Total field: for example, ':27:1/4 MT 700' is the first (MT 700) part of the application.
- MT 700 and MT 701s will be mingled in the same file.
- Each record starts with a colon and a SWIFT field identifier, followed by another colon: for example, ':40A:'-
- Each application is separated by a line with only the ASCII 3 symbol (a heart) on it.

Examples of how individual lines of the MT 700 or MT 701 should look:

```
:27:1/4
:40A:IRREVOCABLE
:20:29893098
:23:NOREF
:31C:910906
:31D:911022DALLAS
:51D:NORTHERN TRUST INT'L BANKING CORP.
      ONE WORLD TRADE CENTER
SUITE 3941
NY, NY 10048 USA
```

The layout of the S.W.I.F.T MT 700 (Issue of a Documentary Credit) file is as follows:

SWIFT I.D. DATA TYPE CODES (refer to SWIFT User Handbook - Standards general Information - October 1998 release for formatting information):

Note

There is always a new line (nl) after every individual SWIFT ID (and there may be more than one line within an individual field [for example, 59 – Beneficiary, four lines to hold address information]).

In some situations, certain fields will be blank. These fields should be skipped over. In other words, no blank line or tag should be printed indicating the field is blank. Simply ignore it.

Invoices

Merchandising has scheduled integration for the following invoices for communication with Oracle Retail Invoice Matching Cloud Service:

- [Stage Complex Deal Invoice Information \(vendinvc\)](#)
- [Stage Fixed Deal Invoice Information \(vendinvf\)](#)
- [Download of Invoice for Invoice Matching \(edidlinv\)](#)

Download of Invoice for Invoice Matching (edidlinv)

Module Name	edidlinv.pc
Description	Download of Invoice For Invoice Matching
Functional Area	Invoice Matching
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS127
Wrapper Script	rmswrap_multi_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The EDIDLINV program extracts invoice information from Merchandising invoice tables (INVC_HEAD, INVC_DETAIL) to a flat file. This flat file is used by Invoice Matching to upload invoice data into tables such as IM_DOC_HEAD, IM_INVOICE_DETAIL and IM_DOC_NON_MERCH. This batch program is run daily, extracting invoice records whose invoice date falls on the current vdate.

If the batch is run ad hoc, there may be issues when consignment invoices are generated as the sales process can also run multiple times a day. Invoice information can potentially be not the latest when the extract is generated.

Restart/Recovery

Restart/recovery for this program is set up at the invoice ID and line sequence level. The program resumes writing to file starting on the next line where the previous process ended.

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000024

Output File Layout

Table 6-17 edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Describes file record type. Valid value is FHEAD.
	Line id	Number(10)	0000000001	Sequential file line number.
	Gentran ID	Char(5)	UPINV	The type of transaction this file represents. Valid value is UPINV.
	Current date	Char(14)		Vdate in YYYYMMDDHH24MISS format.
	File Version	Number(2)	05	Numeric number of the file version to backward compatibility. Starting with 2, incremented by 1 each time the file format changes
THEAD	Record descriptor	Char(5)		Describes file record type. Valid value is THEAD.
	Line id	Number (10)		Sequential file line number.
	Transaction number	Number(10)		Sequential transaction number. All records within this transaction will also have this transaction number.
	Document Type	Char(6)		Describes the type of document being uploaded. The document type will determine the types of detail information that are valid for the document upload. Invoice types are held on the codes table under a code type of 'IMIT'.
	Vendor Document Number	Char (50)		Vendor's document number.
	Group ID	Char(10)	NULL	The Group ID is an informational field, which can be used to identify groups of invoices that were transmitted to Invoice Matching together. This is not populated by Merchandising.
	Vendor Type	Char(6)		Type of vendor (either supplier or partner) for this document. Valid values include Bank 'BK', Agent 'AG', Freight Forwarder 'FF', Importer 'IM', Broker 'BR', Factory 'FA', Applicant 'AP', Consolidator 'CO', Consignee 'CN', Supplier Hierarchy Level 1 'S1', Supplier Hierarchy Level 2 'S2', and Supplier Hierarchy Level 3 'S3'. These partner types will be held on the codes table under the code_type 'PTAL'.
	Vendor ID	Char(10)		Vendor for this document.
	Vendor Document Date	Char(14)		Date document was issued by the vendor (in YYYYMMDDHH24MISS format).

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Order Number / RTV order number	Number(12)		Merchandising system order number for this document. Required for merchandise invoices and optional for others. This field can also contain the RTV order number if the RTV flag is 'Y'
	Location	Number(10)		Merchandising system location for this document.
	Location Type	Char(1)		Merchandising system location type (either 'S'tore or 'W'arehouse) for this document. Required for merchandise invoices and optional for others.
	Terms	Char(15)		Terms of this document. If terms are not provided, the vendor's default terms will be associated with this record.
	Due Date	Char(14)		Date the amount due is due to the vendor (YYYYMMDDHH24MISS format). If due date is not provided, default due date is calculated based on vendor and terms.
	Payment method	Char(6)		Method for paying this document.
	Currency code	Char(3)		Currency code for all monetary amounts on this document.
	Exchange rate	Number(20,4)		Exchange rate *10000 (implied 4 decimal places) for conversion of document currency to the primary currency.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) total cost amount.
	Total Cost	Number(20,4)		Total document cost *10000 (implied 4 decimal places), including all items and costs on this document. This value is in the document currency.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) total tax amount.
	Total Tax Amount	Number(20,4)		Total TAX amount *10000 (implied 4 decimal places), including all items and costs on this document. This value is in the document currency.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) total quantity amount.
	Total Quantity	Number(12,4)		Total quantity of items *10000 (implied 4 decimal places) on this document. This value is in EACHES (no other units of measure are supported in Invoice Matching).
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) total discount amount.

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Total Discount	Number(12,4)		Total discount *10000 (implied 4 decimal places) applied to this document. This value is in the document currency.
	Freight Type	Char(6)	NULL	The freight method for this document. Always blank.
	Paid Ind	Char(1)		Indicates if this document has been paid.
	Multi-Location	Char(1)	N	Indicates if this invoice goes to multiple locations.
	Merchandise Type	Char(1)		Indicates if this invoice is a consignment invoice.
	Deal Id	Number(10)	NULL	Deal Id from Merchandising if this invoice is a deal bill back invoice. Always blank.
	Deal Detail Id	Char(10)	NULL	Complex Deal Component Id. Always blank from Merchandising.
	Ref CNR Ext Doc Id	Char(50)	NULL	Reference to the External Id of Credit Note Request associated with this document. Always blank from Merchandising.
	Ref INV Ext Doc Id	Char(50)	NULL	Reference to the External Id of Invoice associated with this document. Always blank from Merchandising.
	Deal Approval Indicator	Char(1)	NULL	Indicates if the document on IM_DOC_HEAD is to be created in Approved or Submitted status. Always blank from Merchandising.
	RTV indicator	Char(1)		Indicates if this invoice is a RTV invoice.
	Custom Document Reference 1	Char(90)	NULL	This optional field is included in the upload file for client customization. No validation will be performed on this field. Always blank from Merchandising.
	Custom Document Reference 2	Char(90)	NULL	This optional field is included in the upload file for client customization. No validation will be performed on this field. Always blank from Merchandising.
	Custom Document Reference 3	Char(90)	NULL	This optional field is included in the upload file for client customization. No validation will be performed on this field. Always blank from Merchandising.

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TDETL	Custom Document Reference 4	Char(90)	NULL	This optional field is included in the upload file for client customization. No validation will be performed on this field. Always blank from Merchandising.
	Cross-reference document number	Number(10)		Document that a credit note is for. Blank for all document types other than merchandise invoices.
	Third-Party Payee	Number(10)	NULL	This optional field is included in the upload file for client customization. No validation will be performed on this field. Always blank from Merchandising.
	Fiscal ID	Char(240)	NULL	This is fiscal document ID. Always blank from Merchandising.
	Best Terms Date	Char(14)	NULL	Basis Date for calculating the Payment Date. Always blank from Merchandising.
	Record descriptor	Char(5)		Describes file record type. Valid value is TDETL.
	Line id	Number(10)		Sequential file line number.
	Transaction number	Number(10)		Transaction number for this item detail record.
	UPC	Char(25)	NULL	UPC for this detail record. Valid item number will be retrieved for the UPC. Always blank from Merchandising.
	UPC Supplement	Number(5)	NULL	Supplement for the UPC. Always blank from Merchandising.
	Item	Char(25)		Item for this detail record.
	VPN	Char(30)	NULL	Vendor Product Number which can (optionally) be used instead of the Oracle Retail Item Number.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) Original Document Quantity amount.
	Original Document Quantity	Number(12,4)		Quantity *10000 (implied 4 decimal places), in EACHES, of the item on this detail record.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) Original Unit Cost amount.
	Original Unit cost	Number(20,4)		Unit cost *10000 (implied 4 decimal places), in document currency, of the item on this detail record.
	Original Tax Code	Char (6)		Tax code for item. This will be Blank if the SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Original Tax rate	Number (20,10)		Tax Rate for the Tax code/item. This will be Blank if the SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) total allowance. Default is "+" if no allowances exist for this detail record.
	Total Allowance	Number(20,4)		Sum of allowance details for this item detail record *10000 (implied 4 decimal places). If no allowances exist for this item detail record, value will be 0.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Tax Basis. Always blank from Merchandising.
	Tax Basis	Number (20,4)		Specify the basis for which this tax should be calculated. If not provided and tax calc type is 'P', use the qty * unit cost as basis. Always blank from Merchandising.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Per Unit Tax. Per Unit tax will be negative on Debit Memos, Credit Note Requests, and Credit Notes since quantity is always positive. Always blank from Merchandising.
	Per Unit Tax	Number (20,4)		Used to calculate the tax amount when tax rate type is 'U'. Tax amount = quantity * per unit tax. (Quantity is the quantity of the item). Always blank from Merchandising.
TDTLT	Record descriptor	Char(5)		Describes file record type. Valid value is TDTLT. This record will be populated only if SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.
	Line id	Number(10)		Sequential file line number.
	Transaction number	Number(10)		Transaction number for this item tax detail record.
	UPC	Char(25)	NULL	UPC for this detail record. Valid item number will be retrieved for the UPC. Always blank from Merchandising.
	UPC Supplement	Number(5)	NULL	Supplement for the UPC. Always blank from Merchandising.
	Item	Char(25)		Item for this tax detail record.
	VPN	Char(30)	NULL	Vendor Product Number which can (optionally) be used instead of the Oracle Retail Item Number.

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Original Tax Code	Char (6)		Tax code for item.
	Original Tax rate	Number (20,10)		Tax Rate for the Tax code/item
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Tax Basis.
	Tax Basis	Number (20,4)		Specify the basis for which this tax should be calculated. If not provided and tax calc type is 'P', use the qty * unit cost as basis.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Per Unit Tax. Per Unit tax will be negative on Debit Memos, Credit Note Requests, and Credit Notes since quantity is always positive.
	Per Unit Tax	Number (20,4)		Used to calculate the tax amount when tax rate type is 'U'. Tax amount = quantity * per unit tax. (Quantity is the quantity of the item).
	TNMRC	Char(5)		Describes file record type. Valid value is TNMRC.
	Line id	Number (10)		Sequential file line number.
	Transaction number	Number(10)		Transaction number for this non-merchandise record.
	Non Merchandise Code	Char(6)		Non-Merchandise code that describes this cost.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) Non Merchandise Amt.
	Non Merchandise Amt	Number(20,4)		Cost *10000 (implied 4 decimal places) in the document currency.
	Non Merch TAX Code	Char (6)		Tax Code for Non-Merchandise. This will be Blank if the SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.
	Non Merch Tax Rate at this TAX code	Number (20, 10)		Tax Rate corresponding to the Tax code. This will be Blank if the SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.
	Service Performed Indicator	Char(1)		Indicates if a service has actually been performed.
	Store	Number(10)		Store at which the service was performed.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Tax Basis. Always blank from Merchandising.

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TNMRT	Tax Basis	Number (20,4)		Specify the basis for which this tax should be calculated. Always blank from Merchandising.
	Record descriptor	Char(5)		Describes file record type. Valid value is TNMRT. This record will be populated only if SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.
	Line id	Number (10)		Sequential file line number.
	Transaction number	Number(10)		Transaction number for this non-merchandise record.
	Non Merchandise Code	Char(6)		Non-Merchandise code that describes this cost.
	Non Merch TAX Code	Char (6)		Tax Code for Non-Merchandise.
	Non Merch Tax Rate at this TAX code	Number (20, 10)		Tax Rate corresponding to the Tax code.
TVATS	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Tax Basis.
	Tax Basis	Number (20,4)		Specify the basis for which this tax should be calculated.
	File record descriptor	Char(5)		Marks costs at tax rate line. Valid value is TVATS.
	Line id	Char(10)		Sequential file line number
	Transaction number	Number(10)		Transaction number for this tax detail record.
	TAX code	Char(6)		Tax code that applies to cost
	TAX rate	Number (20,10)		Tax Rate corresponding to the Tax code. In case of SYSTEM_OPTIONS.DEFAULT_TAX_TYPE being GTS, this will be populated only when the tax calc type is 'P'.
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) Original Document Quantity amount. In case of SYSTEM_OPTIONS.DEFAULT_TAX_TYPE being GTS, this will be populated only when the tax calc type is 'P'.
	Cost at this TAX code	Number (20,4)		Total amount *10000 (implied 4 decimal places) that must be taxed at the above TAX code. In case of SYSTEM_OPTIONS.DEFAULT_TAX_TYPE being GTS, this will be populated only when the tax calc type is 'P'.

Table 6-17 (Cont.) edidlinv.pc - Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Sign Indicator	Char(1)		Indicates either a positive (+) or a negative (-) for Tax Amount. This will be calculated only if the SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS.
	Total Tax Amount	Number (20,4)		Invoice level total tax amount * 1000 (implied 4 decimal places) for the tax code. It includes both merchandise and non-merchandise taxes. This will be calculated only if the SYSTEM_OPTIONS.DEFAULT_TAX_TYPE is GTS. RMS will sum up the values in the Merch Tax Detail (i.e. TDTLT) and the Non-Merch Tax Detail (i.e. TNMRT) and aggregate it by Tax Code and Tax Rate if the tax calc type is 'P' and aggregate it by Tax Code if the tax calc type is 'U'.
TTAIL	Record descriptor	Char(5)		Describes file record type. Default value is TTAIL.
	Line id	Number(10)		Sequential file line number.
	Transaction number	Number(10)		Transaction number for the transaction that this record is closing.
	Transaction lines	Number(6)		Total number of detail lines within this transaction.
FTAIL	Record descriptor	Char(5)		Describes file record type.
	Line id	Number(10)		Sequential file line number.
	Number of lines	Number(10)		Total number of lines within this file excluding FHEAD and FTAIL.

Design Assumptions

N/A

Stage Complex Deal Invoice Information (vendinvc)

Module Name	vendinvc.pc
Description	Stage Complex Deal Invoice Information
Functional Area	Deals
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS122
Wrapper Script	rmswrap.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The batch module creates records in invoice match staging tables dealing for complex type deals.

The invoicing logic will be driven from the billing period estimated next invoice date for complex deals. The amount to be invoiced will be the sum of the income accruals of the deal since the previous invoice date (or the deal start date for the first collection).

prepost vendinvc pre - truncates STAGE_COMPLEX_DEAL_HEAD and STAGE_COMPLEX_DEAL_DETAIL tables to remove previous days records.

prepost vendinvc post - calls the process_deal_head() function to update est_next_invoice_date of the deal to NULL.

Restart/Recovery

When the max commit point is reached, the data is updated.

I/O Specification

Integration Type	Download from Merchandising
File Name	N /A
Integration Contract	IntCon000009

Records are written to the stage_complex_deal_head and stage_complex_deal_detail tables.

Design Assumptions

N/A

Stage Fixed Deal Invoice Information (vendinvc)

Module Name	vendinvc.pc
Description	Stage Complex Deal Invoice Information
Functional Area	Deals
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS123
Wrapper Script	rmswrap_multi.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The batch module creates records in staging tables dealing for fixed type deals.

The invoicing logic will be driven by the collection dates for fixed deals. The amount to be invoiced will be retrieved directly from fixed deal tables for a given deal date.

prepost vendinvf pre - truncates STAGE_FIXED_DEAL_HEAD and STAGE_FIXED_DEAL_DETAIL tables to remove previous days records.

prepost vendinvf post – calls the process_fixed_deal function to update the status of the fixed deal claim to 'I' (inactive)

Restart/Recovery

Data is committed to the database once the number of transactions processed reaches or exceeds the max_commit_ctr.

I/O Specification

Integration Type	Download from Merchandising
File Name	N /A
Integration Contract	IntCon000009

Records are written to the stage_complex_deal_head and stage_complex_deal_detail tables.

Design Assumptions

N/A

Inventory

Merchandising has scheduled integration for inventory and sales data through the following processes:

- [Download Sales and Stock on Hand to Suppliers \(edidlprd\)](#)
- [Future Available Inventory Publication API \(BDI_COFutureAvail Tx PF From RMS_JOB\)](#)
- [Inventory Publication API \(BDI_Inventory Tx PF From RMS_EOW_JOB\)](#)
- [Item Location History \(BDI_ItemLocHist Tx PF From RMS_JOB\)](#)
- [Reject POSU Transactions \(salesgenrej.ksh\)](#)
- [Store Available Inventory Publication API \(BDI_InvAvailStore Tx PF From RMS_JOB\)](#)
- [Warehouse Inventory Publication API \(BDI_InvAvailWh Tx PF From RMS_JOB\)](#)

Download Sales and Stock on Hand to Suppliers (edidlprd)

Module Name	edidlprd.pc
Description	Download Sales and Stock On Hand to Suppliers
Functional Area	Inventory
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS47
Wrapper Script	rmswrap_multi_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program is used to transmit item-level sales and stock-on-hand information to vendors. The report is a summary that will be sent to specified suppliers through EDI, giving sales details, as well as current stock on hand and in transit for all locations for each of the items supplied by that supplier. Only those suppliers which have an EDI sales reporting frequency of either daily or weekly will have files generated by this program. The system parameter EDI Daily Report Lag is used for suppliers receiving daily updates to determine the day lag for sales data sent, to account for late posting sales.

Restart/Recovery

Restart/recovery in this program is achieved through utilizing a global temporary table. Once a supplier is processed, it is deleted from the temporary table to prevent the same supplier from being processed again during recovery.

I/O Specification

Integration Type	Download from Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000013

Output File Layout

Table 6-18 edidlprd.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File record descriptor	Char(5)	FHEAD	Describes record type
	Line number	Number(10)	0000000001	Sequential file line number
	File source	Char(5)	DLPRD	File Type
	File create date	Char(8)	N/A	Date that the file was created in YYYYMMDD format
THEAD	File record descriptor	Char(5)	THEAD	Identifies record type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number

Table 6-18 (Cont.) edidlprd.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
TITEM	Report date	Char(8)	N/A	For weekly reporting, this will contain the current date. For daily reporting, it will be the date represented by the sales, current date – lag days. Both will be in the YYYYMMDD format
	Supplier	Number(10)	N/A	Merchandising Supplier Number
	File record descriptor	Char(5)	TITEM	Identifies file record type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number
	Item	Char(25)	N/A	Transaction level item to which with the data is related
	Item_Num_Type	Char(6)	N/A	Contains the item number type for the item on ITEM_MASTER
	Ref_Item	Char(25)	N/A	Contains the primary reference item for the item in the file, if defined
	Ref_Item_Num_Type	Char(6)	N/A	Contains the item number type for the reference item from ITEM_MASTER
	Vendor catalog number	Char(30)	N/A	Contains the VPN (Vendor Product Number), if defined for the item/supplier
TQUTY	Item description	Char(250)	N/A	Contains the transaction level item description from ITEM_MASTER
	File record descriptor	Char(5)	TQUTY	Identifies record type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number

Table 6-18 (Cont.) edidlprd.pc - Output File

Record Name	Field Name	Field Type	Default Value	Description
	Quantity descriptor	Char(15)	N/A	Indicates what the quantity represents, either 'On-hand' (stock), 'Sold'(sales), or 'In transit'
	Location type	Char(2)	N/A	Indicates the type of location represented in the file: 'ST' for store or 'WH' warehouse
	Location	Number(10)	N/A	Contains the store or warehouse number for which the information applies
	Unit cost	Number(20)	N/A	Contains the current unit cost for the item/ location with 4 implied decimal places. This value will be in the supplier's currency
	Quantity	Number(12)	N/A	Indicates the quantity of the item sold, on hand or in transit to the location; the quantity is represented with 4 implied decimal places
TTAIL	File record descriptor	Char(5)	TTAIL	Identifies record type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction lines	Number(6)	N/A	Number of lines for this transaction
FTAIL	File record descriptor	Char(5)	FTAIL	Identifies record type
	Line number	Number(10)	N/A	Total number of lines in file
	Number of transaction lines	Number(10)	N/A	Number of transaction lines in file

Design Assumptions

A data translator will be used to convert the flat file produced by Merchandising to the required EDI data format.

Only data for items where the supplier is indicated as the primary supplier/origin country for the item will be included in the report.

Future Available Inventory Publication API (BDI_COFutureAvail_Tx_PF_From_RMS_JOB)

This section describes the Future Available Inventory Publication BDI.

Functional Area

Inventory

Design Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of on-order quantity for all item/location combinations that are flagged as back-orderable in Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

The following packages are impacted:

Bulk Interface Module

In the bulk interface module:

Filename: bdiavinvb.pls

```
BDI_AV_INV_SQL.CO_FUTURE_AVAIL_UP(O_error_message IN OUT VARCHAR2,
                                   O_control_id     IN OUT NUMBER,
                                   I_job_context     IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Item Inventory tables/view.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
CO Future Avail	CO Future Availability	COFutureAvail_Tx_BdiInterfaceModule .xml

Tables

TABLE	SELECT	INSERT	UPDATE	DELETE
CO_FUTURE_AVAIL_OUT	No	Yes	No	No
V_BDI_CO_FUTURE_AVAIL	Yes	No	No	No

Inventory Publication API (BDI_Inventory_Tx_PF_From_RMS_EOW_JOB)

This section describes the Item Inventory Publication BDI.

Functional Area

Inventory

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of inventory from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

Package Impact

Filename: bdimfpb.pls

```
BDI_MFP_SQL.INVENTORY_UP(O_error_message IN OUT RTK_ERRORS.RTK_TEXT%TYPE,
                          O_control_id    IN OUT NUMBER,
                          I_job_context   IN    VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Item Inventory tables/view.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Inventory	Inventory upload to BDI	Inventory_Tx_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
INVENTORY_OUT	No	Yes	No	No
V_BDI_MFP_INVENTORY	Yes	No	No	No

Item Location History (BDI_ItemLocHist_Tx_PF_From_RMS_JOB)

Module Name BDI_ItemLocHist_Tx_PF_From_RMS_JOB

Description	Extracts Sales History
Functional Area	Sales
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A
Runtime Parameters	ItemLocHist_Tx_ProcessFlow_From_RMS ItemLocHist_Tx_Extractor

Design Overview

Merchandising extracts item-location sales history on a weekly basis. It utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement from Merchandising to an external solution.

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	Select	Insert	Update	Delete
ITEM_LOC_HIST	Yes	No	No	No
ITEM_LOC_HIST_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

Refer to ItemLocHist_Tx_BdiInterfaceModule.xml.

Reject POSU Transactions (salesgenrej.ksh)

Module Name	salesgenrej.ksh
Description	Reject POSU Transactions
Functional Area	Sales Posting
Module Type	Business Processing

Module Technology	KSH
Catalog ID	RMS338
Wrapper Script	batch_salesgenrej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this module is to archive the rejected transactions and create a reject file based on the recently processed POSU file which is still in the staging table. It will also generate a retry file based on input parameter (Retry Indicator¹) if error was due to locking and if the number of attempts does not exceed the retry lock attempt configuration (RMS_PLSQL_BATCH_CONFIG.RETRY_LOCK_ATTEMPT).

Restart/Recovery

N/A

Performance Considerations

The number of threads, the amount of waiting time, number for retries, and average volume of data should be considered. RETRY_WAIT_TIME shouldn't be increased significantly.

Reject File:

The module will have the ability to re-process the reject file directly. The file format will therefore be identical to the input file layout. A reject line counter will be kept in the program and is required to ensure that the file line count in the trailer record matches the number of rejected records. If no errors occur, no reject files would be generated.

Retry File:

If the retry input indicator is set to Y, then a retry file is going to be generated if the error is due to locking only. This will be automatically be placed at the input directory ready to be picked up by the next sales upload and processing.

Store Available Inventory Publication API (BDI_InvAvailStore_Tx_PF_From_RMS_JOB)

This section describes the Store Available Inventory Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Store Address information from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API that will pull data from Merchandising and deliver these to the BDI integration layer. This API will be in the form of a PLSQL function inside a PLSQL package.

¹ Input parameter is not required and is defaulted to N that will disable the generate the retry file functionality for locked records.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiavinvb.pls

```
BDI_AV_INV_SQL.ST_AVAIL_INV_UP(0_error_message IN OUT VARCHAR2,
                                0_control_id    IN OUT NUMBER,
                                I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Merchandise Hierarchy tables.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition

Data Flow	Description	XML Schema Definition (XSD)
Store Inventory	Store inventory upload to BDI	InvAvailStore_Tx_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
INV_AVAIL_STORE_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
STORE	Yes	No	No	No

Warehouse Inventory Publication API (BDI_InvAvailWh_Tx_PF_From_RMS_JOB)

This section describes the Warehouse Inventory Publication BDI.

Functional Area

Foundation

Business Overview

BDI (Bulk Data Integration) is an integration layer that facilitates the bulk transfer of Warehouse Inventory positions from Merchandising to other Oracle Retail Applications. On this particular integration stream, the data flow is from Merchandising to BDI, and then BDI to downstream applications. To accomplish this data transfer, BDI will be calling a Merchandising-owned API

that will pull data from Merchandising and deliver these to the BDI integration layer. This API is in the form of a PLSQL function inside a PLSQL package.

Package Impact

This section describes the package impact.

Bulk Interface Module

Filename: bdiavinvb.pls

```
BDI_AV_INV_SQL.WH_AVAIL_INV_UP(0_error_message IN OUT VARCHAR2,
                                0_control_id    IN OUT NUMBER,
                                I_job_context    IN      VARCHAR2)
```

This function begins by calling a BDI function that signals the start of the interface process. The BDI function will update the internal BDI control tables to track the progress of the API.

A DML insert statement is then executed to populate the BDI outbound table that resides in the BDI_RMS_INT_SCHEMA schema. This outbound table is loaded with records from the Merchandising Item Location table.

After the insert, another call to a BDI function is performed to signify the successful loading of records. This will update the internal BDI control tables.

A database commit is issued, and the control Id is returned by the API.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Warehouse Inventory Avail	Wh Available Inventory	InvAvailWh_Tx_BdiInterfaceModule.xml

Table Impact

TABLE	SELECT	INSERT	UPDATE	DELETE
INV_AVAIL_WH_OUT	No	Yes	No	No
ITEM_MASTER	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	No	No
WH	Yes	No	No	No

Planning and Forecasting

Merchandising provides critical foundation and transactional information to the Oracle Retail planning and forecasting solutions. Because the planning and forecasting solutions are built on the same platform, several of the integrations from Merchandising are used by more than one of the solutions. The tables below summarize the key outbound integration points by solution.

Table 6-19 Integration to Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)

Description	Program
Calendar Extract to Planning and Forecasting	BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB
Currency Rates Extract to Planning and Forecasting	BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB
Inventory Extract to Planning	BDI_MFP_Inventory_Tx_PF_From_RMS_JOB
Merchandise Hierarchy and Item Extract to Planning and Forecasting	BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB
On Order Extract to Planning	BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB
Organization Hierarchy Extract to Planning and Forecasting	BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB
Store Extract to Planning and Forecasting	BDI_RPAS_Store_Fnd_PF_From_RMS_JOB
Transaction Data Extract to Planning	BDI_MFP_TranData_Tx_PF_From_RMS_JOB

Table 6-20 Integration to Oracle Retail Assortment and Item Planning for Fashion/Softlines Cloud Service (A&IP CS)

Description	Program
Brand Extract to Planning	BDI_RPAS_Brand_Fnd_PF_From_RMS_JOB
Calendar Extract to Planning and Forecasting	BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB
Currency Rates Extract to Planning and Forecasting	BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB
Differentiator Extract to Planning	BDI_RPAS_Diff_Fnd_PF_From_RMS_JOB
Inventory Extract to Planning	BDI_MFP_Inventory_Tx_PF_From_RMS_JOB
Merchandise Hierarchy and Item Extract to Planning and Forecasting	BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB
On Order Extract to Planning	BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB
Organization Hierarchy Extract to Planning and Forecasting	BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB
Store Extract to Planning and Forecasting	BDI_RPAS_Store_Fnd_PF_From_RMS_JOB
Supplier Extract to Planning	BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB
Transaction Data Extract to Planning	BDI_MFP_TranData_Tx_PF_From_RMS_JOB
UDA Extract to Planning	BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB
UDA Item Extract to Planning and Forecasting	BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB

Table 6-21 Integration to Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting

Description	Program
Calendar Extract to Planning and Forecasting	BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB
Merchandise Hierarchy and Item Extract to Planning and Forecasting	BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB
Organization Hierarchy Extract to Planning and Forecasting	BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB
Out of Stock Extract to Forecasting	BDI_RDF_StockOut_Tx_PF_From_RMS_JOB
Store Extract to Planning and Forecasting	BDI_RPAS_Store_Fnd_PF_From_RMS_JOB
UDA Item Extract to Planning and Forecasting	BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB
Weekly Sales Extract to Forecasting	BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB

Brand Extract to Planning (BDI_RPAS_Brand_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_Brand_Fnd_PF_From_RMS_JOBbdi_merch_extract_to_file_wrapper.shbdi_rpas_brand_extract.ksh
Description	Extracts Brand information to Planning
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	Brand_Fnd_ProcessFlow_From_RMSBrand_Fnd_ExtractorDatabase connection, download file location, filename, trigger filename

Design Overview

This process extracts its brand data to Planning on a weekly basis.

Key assumptions for this integration:

- The full set of brands is included in this integration each time it runs.
- Retailers will not create a Diff with an ID of 'BRAND'.
- In order to meet the format required by Planning, the UDA description in this extract is hard coded to "Brand" and does not take into account the primary language configuration in Merchandising.
- The intended targets for this integration are
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to Planning. The batch job BDI_RPAS_Brand_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_Brand_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts Brand information and writes it out to
a flat file for processing by AP and IP."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateds" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_RPAS_Brand_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (Brand_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (Brand_Fnd_Extractor) calls BDI_FOUNDATION_SQL.BRAND_UP function to extract data from Merchandising table BRAND to BDI outbound staging table BRAND_OUT.
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rpas_brand_extract.ksh to write brand information from the BRAND_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Two separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A

Schedule Information	Description
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
BRAND	Yes	No	No	No
BRAND_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
UDA_ID	Char(6)	Yes	Hardcoded to 'BRAND'
UDA_DESC	Char(120)	Yes	Hardcoded to 'Brand'
BRAND_NAME	Char(30)	Yes	The brand ID from the Merchandising Brand table.
BRAND_DESCRIPTION	Char(120)	Yes	The brand description in the primary language from the Merchandising Brand table.

Calendar Extract to Planning and Forecasting (BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB)

Note

This module replaces the ftmednld.pc module from previous releases.

Module Name	BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB
Description	Extracts calendar information to RPAS from RMS
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A

Runtime Parameters Calendar_Fnd_ProcessFlow_From_RMS Calendar_Fnd_Extractor

Design Overview

This program extracts calendar data to planning and forecasting on a weekly basis.

Key assumptions for this integration:

- The last two years, current year, and two years into the future are extracted each time this process is run.
- A data set is sent each time the extract runs.
- This extract supports a 4-5-4 calendar only.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This program utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement from Merchandising to the target applications.

The batch job BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts calendar information and writes it
out to a flat file for processing by both MFP and RDF."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_RPAS_Calendar_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (Calendar_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (Calendar_Fnd_Extractor) calls BDI_FOUNDATION_SQL.CALENDAR_UP function to extract data from Merchandising view V_BDI_DAY_LEVEL_CALENDAR to BDI outbound staging table CALENDAR_OUT.

- A generic BDI Downloader file creator job writes calendar information from the CALENDAR_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - MFP_outboundLocation
 - RDF_outboundLocation
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
V_BDI_DAY_LEVEL_CALENDAR	Yes	No	No	No
CALENDAR_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
DAY	Date	Yes	The date for which the data was derived, in YYYYMMDD format
WEEK	Date	Yes	The end of week date for the day, in YYYYMMDD format

Field Name	Field Type	Required	Description
MONTH	Number(2)	Yes	The month number of the day in the year; valid values 1-12
QUARTER	Number(1)	Yes	The quarter of the year for the day; valid values 1-4
HALF	Number(1)	Yes	The half of the year for the day; valid values are 1 or 2
YEAR	Number(4)	Yes	The year for the day (YYYY format).
WEEK_OF_YEAR	Number(2)	Yes	The week of the year for the day; valid values 1-53
DAY_OF_WEEK	Number(1)	Yes	The day number within the week; valid values 1-7.

Currency Rates Extract to Planning and Forecasting (BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rpas_curr_conv_rates_extract.ksh
Description	Extracts currency rates information to RPAS
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	CurrConvRates_Fnd_ProcessFlow_From_RMS CurrConvRates_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This program extracts its currency rates data to planning and forecasting on a weekly basis.

Key assumptions for this integration:

- Only currency rates for which stores and warehouse exist will be included in the extract.
- Either the consolidated or operational rate will be sent based on the setting of the Consolidation system option. If Y, then the consolidation rates will be sent. If N, then the operational rates are used.
- All applicable currency rates are sent each time this process is run.
- The rates sent in this integration are based on a materialized view. The process that refreshes this view (batch_rfmvcrrconv.ksh) must be scheduled to ensure that the latest currency information is sent each week.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting

- Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This program utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement from Merchandising to the target applications.

The batch job BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts currency conversion rate
information and writes it out to a flat file for processing by both MFP and RDF."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_RPAS_CurrConvRates_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (CurrConvRates_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (CurrConvRates_Fnd_Extractor) calls BDI_FOUNDATION_SQL.CURR_CONV_RATES_UP function to extract data from Merchandising view MV_CURRENCY_CONVERSION_RATES to BDI outbound staging table CURR_CONV_RATES_OUT.
 - Only the currencies for which stores and warehouses exist in Merchandising will be extracted.
 - Either consolidated or operational rates will be included based on Merchandising system options (consolidation_ind).
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rpas_curr_conv_rates_extract.ksh to write currency rates information from the CURR_CONV_RATES_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - MFP_outboundLocation
 - RDF_outboundLocation

- AP_outboundLocation
- IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date
Scheduling Considerations	N/A
Pre-Processing	batch_rfmvcrrconv.ksh
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
MV_CURRENCY_CONVERSION_RATES	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
CURR_CONV_RATES_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
EFFECTIVE_DATE	Date	Yes	Holds the effective date of the exchange rate for the currencies and the exchange type
FROM_CURRENCY_CODE	Char(3)	Yes	Holds the convert from currency code.
TO_CURRENCY_CODE	Char(3)	Yes	Holds the convert to currency code.
EXCHANGE_TYPE	Char(1)	Yes	Identifies the type of exchange rate. This will be either C (consolidation) or O (operational).

Field Name	Field Type	Required	Description
EXCHANGE_RATE	Number(20,10)	Yes	Contains the exchange rate between the from and to currencies for the specified exchange type on the next effective date. It is expressed in terms of the to-currency.

Differentiator Extract to Planning (BDI_RPAS_Diff_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_Diff_Fnd_PF_From_RMS_JOBbdi_merch_extract_to_file_wrapper.shbdi_rpas_diff_extract.ksh
Description	Extracts Diff Types and Diff ID information to Planning
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	Diff_Fnd_ProcessFlow_From_RMS Diff_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This process extracts its differentiator data to Planning on a weekly basis.

Key assumptions for this integration:

- The full set of differentiators and diff types are included in this integration each time it runs.
- The intended targets for this integration are
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to Planning. The batch job BDI_RPAS_Diff_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_Diff_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts Diff Types and Diff ID information and
writes it out to a flat file for processing by AP and IP."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
  </step>
</job>
```

```

        <end on="COMPLETED"/>
    </step>
</job>

```

When the batch job BDI_RPAS_Diff_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (Diff_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (Diff_Fnd_Extractor) calls BDI_CROSS_PILLAR_SQL.DIFF_UP function to extract data from DIFF_IDS and DIFF_TYPE to BDI outbound staging table DIFF_OUT.
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rpas_diff_extract.ksh to write differentiator information from the DIFF_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
DIFF_IDS	Yes	No	No	No
DIFF_TYPE	Yes	No	No	No
DIFF_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
DIFF_TYPE_ID	Char(6)	Yes	The ID of the diff type (for example, C for color).
DIFF_TYPE_DESC	Char(120)	Yes	The description of the diff type (for example, Color) in the primary language.
DIFF_ID	Char(10)	Yes	The ID of the diff (for example, S for Small).
DIFF_DESC	Char(120)	Yes	The description of the diff (for example, Small) in the primary language.

Inventory Extract to Planning (BDI_MFP_Inventory_Tx_PF_From_RMS_JOB)

Module Name	BDI_MFP_Inventory_Tx_PF_From_RMS_JOB
Description	Extracts inventory information to Planning
Functional Area	Inventory
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A
Runtime Parameters	Inventory_Tx_ProcessFlow_From_RMS Inventory_Tx_Extractor

Design Overview

This process extracts owned inventory information for inventoried, non-pack approved transaction items to planning on a weekly basis, at the end of the week. The integration captures the current on-hand and in-transit for all the included item/locations at the point in time that the integration is run.

Key assumptions for this integration:

- Only inventoried, approved transaction items are included in the integration.
- Any inventory for pack items is aggregated with inventory for the component items.
- Only stockholding stores are included in the integration.
- Cost values are based on system configuration for cost:
 - For a cost department with the system configured for average cost, the cost basis is the item/location's weighted average cost, converted to primary currency.
 - For a cost department with the system configured for standard cost, the cost basis is the item/locations unit cost, converted to primary currency.
 - For a retail department, the cumulative mark-on percentage is used to calculate cost based on the retail price, converted to primary currency.

- Retail values sent are based on the current item/location retail price, converted to primary currency. The retail will include VAT if the system option to include VAT in the stock ledger is set to include VAT so that the retail values in this integration are consistent with other data sent to planning.
- All unit values are sent in terms of the standard unit of measure for the item.
- Planning will interpret inventory as being clearance if the clearance flag sent in this integration shows the item/location to be on clearance at the end of the week.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement from Merchandising to the target applications. The batch job BDI_MFP_Inventory_Tx_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_MFP_Inventory_Tx_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts information regarding inventory for
use by the MFP application"/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_MFP_Inventory_Tx_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (Inventory_Tx_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (Inventory_Tx_ExtractorJob) calls BDI_MFP_SQL.INVENTORY_UP function to extract data from Merchandising view V_BDI_MFP_INVENTORY to BDI outbound staging table INVENTORY_OUT.
- A generic BDI Downloader file creator job writes inventory quantities information from the INVENTORY_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Two separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:

- MFP_outboundLocation
- AP_outboundLocation
- IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
V_BDI_MFP_INVENTORY	Yes	No	No	No
INVENTORY_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DA TA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CT L	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
EOW	Date	Yes	Indicates the end of week date that the on order information pertains to.
ITEM	Varchar2(25)	Yes	Transaction level item only.
LOCATION	Number(10)	Yes	Could be a store or virtual warehouse.
LOC_TYPE	Varchar2(1)	Yes	Indicates if the location is a store or warehouse - S = Store; W = Warehouse.
CLEAR_IND	Number(1)	Yes	Indicates if the item/location is currently on clearance.

Field Name	Field Type	Required	Description
REGULAR_INVENTORY_UNITS	Number(12,4)	Yes	Current owned inventory for the item/location in units based on the standard unit of measure; calculated as stock on hand + pack component stock on hand + in transit + pack component in transit.
REGULAR_INVENTORY_COST	Number(20,4)	Yes	The cost value of current owned inventory for the item/location; calculated based on unit inventory and the cost basis of the item's department, as described above.
REGULAR_INVENTORY_RETAIL	Number(20,4)	Yes	The retail value of current owned inventory for the item/location; calculated based on the unit inventory value shown above and the current item/location unit retail.
UNIT_COST	Number(20,4)	Yes	The current supplier purchase cost for the item/location.
AV_COST	Number(20,4)	Yes	The current weighted average cost for the item/location.
UNIT_RETAIL	Number(20,4)	Yes	The current unit retail for the item/location. If the item is on clearance, this would be the clearance price.

Merchandise Hierarchy and Item Extract to Planning and Forecasting (BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rpas_merchhier_extract.ksh
Description	Extracts merchandise hierarchy and item information to RPAS
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	ItemHdrAndMerchHier_Fnd_ProcessFlow_From_RMS ItemHdr_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This program extracts the merchandise hierarchy from company to transaction level item to planning and forecasting on a weekly basis. Additional key attributes about the items are also included, such as the primary supplier, brand, and any differentiators (for example, colors, sizes, and so on) that exist for the item.

Key assumptions for this integration:

- The full merchandise hierarchy and all items are sent each time this process is run.
- Only approved, inventoried and sellable transaction-level items will be included in the integration. Pack items are not included.
- All descriptions are sent in the primary language as defined in Merchandising.
- For transaction items that do not have a parent item, then the transaction item is also displayed as the parent item, as well as the parent/diff level.
- For a parent item that is not marked as an aggregate item or does not have any of its diffs flagged as aggregates, the parent item is sent as the parent/diff level for all of its transaction items.
- A single unit of measure is assumed for all items and therefore the standard units of measure for the items are not sent.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This program utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to the target applications. The batch job BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts Merch Hierarchy information and
writes it out to a flat file for processing by both MFP and RDF."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_RPAS_MerchHier_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (ItemHdrAndMerchHier_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to the target applications:

- Extractor jobs (MerchHier_Fnd_Extractor, ItemHdr_Fnd_Extractor) call respective BDI_MERCH_SQL and BDI_ITEM_SQL functions to extract data from Merchandising tables to BDI outbound staging tables MERCH_HIER_OUT and ITEM_HDR_OUT.

- Downloader file creator job calls the wrapper script, `bdi_merch_extract_to_file_wrapper.sh`, to set the runtime parameters on environment variables. This script will then call `bdi_rpas_merchhier_extract.ksh` to write merchandise hierarchy and item information from the `MERCH_HIER_OUT` and `ITEM_HDR_OUT` tables into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - `MFP_outboundLocation`
 - `RDF_outboundLocation`
 - `AP_outboundLocation`
 - `IP_outboundLocation`

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
COMPHEAD	Yes	No	No	No
DIVISION	Yes	No	No	No
GROUPS	Yes	No	No	No
DEPS	Yes	No	No	No
CLASS	Yes	No	No	No
SUBCLASS	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
DIFF_GROUP_HEAD	Yes	No	No	No
DIFF_IDS	Yes	No	No	No
SYSTEM_OPTIONS	Yes	No	No	No
MERCH_HIER_OUT	Yes	Yes	No	Yes
ITEM_HDR_OUT	Yes	Yes	No	Yes

Table	SELECT	INSERT	UPDATE	DELETE
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No
ITEM_SUPPLIER_OUT	Yes	Yes	No	Yes

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
ITEM	Char(25)	Yes	The transaction level item ID.
ITEM_DESC	Char(250)	Yes	The transaction level item description.
ITEM_PARENT_DIFF	Char(30)	Yes	Concatenated value consisting of item parent ID with the composite diff aggregate. If there is no item parent, this will contain the transaction level item.
ITEM_PARENT_DIFF_DESC	Char(250)	Yes	Description of the item parent diff. Concatenated value consisting of the item parent description and the diff IDs for all diffs associated to the parent marked as aggregates. If there is no item parent, it will contain the transaction level item description.
ITEM_PARENT	Char(25)	Yes	If there is no item parent, it will contain the transaction level item.
ITEM_PARENT_DESC	Char(250)	Yes	If there is no item parent, it will contain the transaction level item description.
SUBCLASS_ID	Number(10)	Yes	Unique subclass ID
SUBCLASS_NAME	Char(120)	Yes	Concatenated value consisting of the subclass number with name.
CLASS_ID	Number(10)	Yes	Unique class ID
CLASS_NAME	Char(120)	Yes	Concatenated value consisting of the class number with name.
DEPT	Number(4)	Yes	Department ID
DEPT_NAME	Char(120)	Yes	Concatenated value consisting of the department ID and name.
GROUP_NO	Number(4)	Yes	Group ID
GROUP_NAME	Char(120)	Yes	Group name
DIVISION	Number(4)	Yes	Division ID
DIV_NAME	Char(120)	Yes	Division name
COMPANY	Number(4)	Yes	Company ID
COMPANY_NAME	Char(120)	Yes	Company name

Field Name	Field Type	Required	Description
FORECAST_IND	Char(1)	Yes	Indicates whether or not the item should be forecasted. Valid values are Y or N.
CLASS	Number(10)	Yes	The class ID that is displayed in the Merchandising screens.
SUBCLASS	Number(10)	Yes	The subclass ID that is displayed in the Merchandising screens.
BRAND_NAME	Char(30)	Yes	If a brand is not assigned, this is defaulted to 'NA'.
BRAND_DESCRIPTION	Char(120)	Yes	The brand description for the transaction item. If a brand is not assigned, this is defaulted to 'Not Assigned'.
SUPPLIER	Number(10)	Yes	The ID of the primary supplier for the transaction item.
SUPPLIER_NAME	Char(240)	Yes	The name of the primary supplier for the transaction item.
DIFF_1	Char(10)	No	The ID of the first diff for the transaction level item. If a diff is not assigned, this is defaulted to 'NA'.
DIFF_1_DESC	Char(120)	No	The name of the first diff for the transaction item. If a diff is not assigned, this is defaulted to 'unassigned'.
DIFF_2	Char(10)	No	The ID of the second diff for the transaction item. If a diff is not assigned, this is defaulted to 'NA'.
DIFF_2_DESC	Char(120)	No	The name of the second diff for the transaction item. If a diff is not assigned, this is defaulted to 'unassigned'.
DIFF_3	Char(10)	No	The ID of the third diff for the transaction item. If a diff is not assigned, this is defaulted to 'NA'.
DIFF_3_DESC	Char(120)	No	The name of the third diff for the transaction item. If a diff is not assigned, this is defaulted to 'unassigned'.
DIFF_4	Char(10)	No	The ID of the fourth diff for the transaction item. If a diff is not assigned, this is defaulted to 'NA'.
DIFF_4_DESC	Char(120)	No	The name of the fourth diff for the transaction item. If a diff is not assigned, this is defaulted to 'unassigned'.

On Order Extract to Planning (BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB)

Note

This module replaces the onordext.pc and onorddnld.pc modules from previous releases.

Module Name	BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB
Description	Extracts inventory information to Planning
Functional Area	Inventory Tracking
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A
Runtime Parameters	OnOrder_Tx_ProcessFlow_From_RMS OnOrder_Tx_Extractor

Design Overview

This process extracts its quantities on order to planning and forecasting on a weekly basis, at the end of the week. The integration sends any open on order quantities aggregated by week, grouped by the open to buy end of week date. Any on order quantity that is still open and has an OTB EOW date in the past will be combined with the current week's on order.

Key assumptions for this integration:

- Only orderable, inventoried, approved transaction items are included in the integration.
- Any on order for pack items is sent based on the component items.
- Purchase orders flagged to not be included in "on order" are not included in the integration.
- Cost and retail values sent are based on the purchase order's cost and retail value, converted to primary currency.
- Retail values will include VAT if the system option to include VAT in the stock ledger is set to include VAT so that the retail values in this integration are consistent with other data sent to planning.
- All unit values are sent in terms of the standard unit of measure for the item.
- Planning will interpret the on order as being clearance if the clearance flag sent in this integration shows the item/location to be on clearance at the end of the week.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to the target applications.

The batch job BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```

<job id="BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts information regarding quantities on
order for use by the MFP application"/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicatedDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>

```

When the batch job BDI_MFP_OnOrder_Tx_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end of week date. If the vdate is an end of week date, it invokes a BDI process flow (OnOrder_Tx_ProcessFlow_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (OnOrder_Tx_Extractor) calls BDI_MFP_SQL. ON_ORDER_UP function to extract data from Merchandising view V_BDI_MFP_ON_ORDER to BDI outbound staging table ON_ORDER_OUT.
- A generic BDI Downloader file creator job writes quantities on order information from the ON_ORDER_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - MFP_outboundLocation
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
V_BDI_MFP_ON_ORDER	Yes	No	No	No
ON_ORDER_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
EOW	Date	Yes	Indicates the end of week date that the on order information pertains to.
ITEM	Varchar2(25)	Yes	Transaction level item only.
LOCATION	Number(10)	Yes	Could be a store or virtual warehouse.
LOC_TYPE	Varchar2(1)	Yes	Indicates if the location is a store or warehouse - S = Store; W = Warehouse.
CLEAR_IND	Number(1)	Yes	Indicates if the item/location is currently on clearance.
ON_ORDER_UNITS	Number(12)	Yes	Indicates the total quantity of the item in the order in standard unit of measure.
ON_ORDER_COST	Number(20,4)	Yes	on order * PO cost in primary currency
ON_ORDER_RETAIL	Number(20,4)	Yes	on order * PO retail in primary currency

Organization Hierarchy Extract to Planning and Forecasting
(BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rpas_orghier_extract.ksh
Description	Extracts organizational hierarchy information to RPAS
Functional Area	Foundation
Module Type	Integration

Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	StoreAndWhAndOrgHier_Fnd_ProcessFlow_From_RMS Store_Fnd_Extractor Wh_Fnd_Extractor OrgHier_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This program extracts the organization hierarchy data from company to location, which can be stores or warehouses to planning and forecasting on a weekly basis. Additional key attributes about the organizational hierarchy will also be sent to assist in building alternate hierarchies for planning, such as channel.

Key assumptions for this integration:

- MFPCS will use the third level of the Merchandising hierarchy (area) to represent channel.
- The full organizational hierarchy is sent each time this process is run.
- All names and descriptions are sent in the primary language only.
- The location in the file can represent either a store or a virtual warehouse location.
- Because warehouses live outside the organization hierarchy, for the levels of the organizational hierarchy above location (chain through district) when the location is a warehouse, the warehouse ID and description will be repeated.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This program utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to the target applications. The batch job BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts Org Hierarchy information and
writes it out to a flat file for processing by both MFP and RDF."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
  <end on="COMPLETED"/>
</job>
```



```
</step>
</job>
```

When the batch job BDI_RPAS_OrgHier_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (StoreAndWhAndOrgHier_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor jobs (Store_Fnd_Extractor, Wh_Fnd_Extractor, OrgHier_Fnd_Extractor) call respective BDI_ORG_SQL functions to extract data from Merchandising tables to BDI outbound staging tables ORG_HIER_OUT, STORE_OUT, and WH_OUT.
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rpas_organhier_extract.ksh to write organization hierarchy information from the ORG_HIER_OUT, STORE_OUT, and WH_OUT tables into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - MFP_outboundLocation
 - RDF_outboundLocation
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
STORE	Yes	No	No	No
WH	Yes	No	No	No
AREA	Yes	No	No	No

Table	SELECT	INSERT	UPDATE	DELETE
CHAIN	Yes	No	No	No
DISTRICT	Yes	No	No	No
REGION	Yes	No	No	No
COMPHEAD	Yes	No	No	No
CHANNELS	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
STORE_FORMAT	Yes	No	No	No
LANG	Yes	No	No	No
VAT_REGION	Yes	No	No	No
TSFZONE	Yes	No	No	No
ORG_HIER_OUT	Yes	Yes	No	Yes
STORE_OUT	Yes	Yes	No	Yes
WH_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
LOCATION	Number(10)	Yes	Store or virtual warehouse ID
LOC_NAME	Char(150)	Yes	Store or warehouse name
DISTRICT	Number(10)	Yes	District ID; for warehouses, repeat the warehouse ID with the prefix "WH"
DISTRICT_NAME	Char(120)	Yes	District name; for warehouses, repeat the warehouse name
REGION	Number(10)	Yes	Region ID; for warehouses, repeat the warehouse ID with the prefix "WH"
REGION_NAME	Char(120)	Yes	Region name; for warehouses, repeat the warehouse name
AREA	Number(10)	Yes	Area ID; for warehouses, repeat the warehouse ID with the prefix "WH"
AREA_NAME	Char(120)	Yes	Area name; for warehouses, repeat the warehouse name
CHAIN	Number(10)	Yes	Chain ID; for warehouses, repeat the warehouse ID with the prefix "WH"
CHAIN_NAME	Char(120)	Yes	Chain name; for warehouses, repeat the warehouse name

Field Name	Field Type	Required	Description
COMPANY	Number(4)	Yes	Company ID
COMPANY_NAME	Char(120)	Yes	Company name
COMPANY_CURRENCY	Char(3)	Yes	The currency code for the base currency defined in system options
LOC_TYPE	Char(1)	Yes	'S' for store, 'W' for warehouse
LOC_TYPE_NAME	Char(120)	Yes	Store or Warehouse depending on location type
PHYSICAL_WH	Number(10)	Yes	Physical warehouse ID for warehouses, repeat store ID for store
PHYSICAL_WH_NAME	Char(120)	Yes	Physical warehouse name for warehouse, repeat store name for stores
CHANNEL_ID	Number(4)	Yes	Channel ID for the store or virtual warehouse; if no channel is defined, then NA
CHANNEL_NAME	Char(120)	Yes	Channel name; if no channel is defined, then 'unassigned'
STORE_CLASS	Char(1)	Yes	For stores, the store class ID; for warehouses or if no store class is defined; then NA.
STORE_CLASS_DESCRIPTION	Char(250)	Yes	For stores, the description of the store class, if defined; for warehouses or if not defined for a store, then 'unassigned'.
STORE_FORMAT	Number(4)	Yes	For stores, the store format ID; for warehouses or if no store class is defined; then NA.
STORE_FORMAT_NAME	Char(60)	Yes	For stores, the description of the store format, if defined; for warehouses or if not defined for a store, then 'unassigned'.

Out of Stock Extract to Forecasting (BDI_RDF_StockOut_Tx_PF_From_RMS_JOB)

Note

This module replaces the southnld.pc module from previous releases.

Module Name	BDI_RDF_StockOut_Tx_PF_From_RMS_JOB
Description	Extracts out of stock item location information to Forecasting
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A

Runtime Parameters StockOut_Tx_ProcessFlow_From_RMS StockOut_Tx_Extractor

Design Overview

This process extracts items which are out of stock for use by Forecasting on a weekly basis. This integration sends all item/store combinations that meet the criteria for review and have a stock-on-hand position of less than or equal to zero at the end of the week.

Key assumptions for this integration:

- Only stockholding stores are included in this integration.
- Only forecasted items are included in this integration.
- Only item/store combinations that have a status of Active and a ranged flag of Yes are reviewed for stock out conditions.
- Only item/store combinations that have a last sold date that is between the end of week date and x number of days back are reviewed for stock out conditions, where x is the value reports system option value Days Since Last Transaction.
- The intended targets for this integration are
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to the target applications. The batch job BDI_RDF_StockOut_Tx_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RDF_StockOut_Tx_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts information for items which are out
of stock for use by the RDF application"/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_RDF_StockOut_Tx_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (StockOut_Tx_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (StockOut_Tx_ExtractorJob) calls BDI_RDF_SQL.STOCKOUT_UP function to extract data from the Merchandising view V_BDI_RDF_STOCKOUT to outbound staging table STOCKOUT_OUT.

- A generic BDI Downloader file creator job writes out of stock item information from the STOCKOUT_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful.
- The downloaded data files and trigger files are written to designated location as configured through BDI system options:
 - RDF_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
V_BDI_RDF_STOCKOUT	Yes	No	No	No
STOCKOUT_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DA TA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CT L	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
ITEM	Varchar2(25)	Yes	Item that is out of stock at the store.
STORE	Number(10)	Yes	Store that is out of stock for the item.
EOW_DATE	Date	Yes	Indicates the end of week date for which the data applies.
OUT_OF_STOCK	Number(1)	Yes	Flag to indicate if the item/store is out of stock at end of week. This will always be 1, as only out-of-stock items are sent.

Store Extract to Planning and Forecasting (BDI_RPAS_Store_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_Store_Fnd_PF_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rpas_store_extract.ksh
Description	Extracts store information to RPAS
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI, shell scripts
Catalog ID	N/A
Runtime Parameters	Store_Fnd_ProcessFlow_From_RMS Store_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This program extracts store data to planning and forecasting on a weekly basis. This data supplements the store information included in the organizational hierarchy feed.

Key assumptions for this integration:

- Both stockholding and non-stockholding stores are included.
- Both company and franchise types of stores are included.
- All stores are sent each time this process is run.
- Planning will derive the status of the store (e.g. open or closed) based on the dates sent in this integration. For example, if the open date is in the past and there is no close date defined or it is a future date, then the store is considered open.
- All descriptions are sent in the primary language as defined in Merchandising.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This program utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement from Merchandising to the target applications.

The batch job BDI_RPAS_Store_Fnd_PF_From_RMS_JOB is defined in the Merchandising JCS batch job admin as follows:

```
<job id="BDI_RPAS_Store_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts store information and writes it out
to a flat file for processing by both MFP and RDF."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
```

```

        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
    </properties>
</batchlet>
<end on="COMPLETED"/>
</step>
</job>

```

When the batch job `BDI_RPAS_Store_Fnd_PF_From_RMS_JOB` is executed, a batchlet (`BDIInvokerBatchlet`) starts the execution flow. It calls a PLSQL function (`RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL`) to ensure the process flow is only executed on an end-of-week date. If the `vdate` is an end-of-week date, it invokes a BDI process flow (`Store_Fnd_ProcessFlow_From_RMS`) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (`Store_Fnd_Extractor`) calls `BDI_ORG_SQL.STORE_UP` function to extract data from Merchandising tables to BDI outbound staging table `STORE_OUT`.
- Downloader file creator job calls the wrapper script, `bdi_merch_extract_to_file_wrapper.sh`, to set the runtime parameters on environment variables. This script will then call `bdi_rpas_store_extract.ksh` to write store information from the `STORE_OUT` table into a comma-delimited flat file, which will be consumed by the target application. A zero-byte trigger file is also generated to signal that the extract process was successful. Two separate copies of the data file and the trigger file are sent to the target application.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - `MFP_outboundLocation`
 - `RDF_outboundLocation`
 - `AP_outboundLocation`
 - `IP_outboundLocation`

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
STORE	Yes	No	No	No
CHANNELS	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
STORE_FORMAT	Yes	No	No	No
LANG	Yes	No	No	No
VAT_REGION	Yes	No	No	No
TSFZONE	Yes	No	No	No
STORE_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
STORE	Number(10)	Yes	Store ID
STORE_NAME	Char(150)	Yes	Store name
DISTRICT	Number(10)	Yes	District in which the store is a member.
STORE_CLOSE_DATE	DATE	Yes	Date on which the store closed. If NULL, set to NA.
STORE_OPEN_DATE	DATE	Yes	Date on which the store opened
REMODEL_DATE	DATE	Yes	Date on which the store was last remodeled. If NULL, set to NA.
STORE_CLASS	Char(1)	Yes	ID for the store class of which the store is a member.
STORE_CLASS_DESCRIPTION	Char(250)	Yes	Store class description
STORE_FORMAT	Number(4)	Yes	Store format. If NULL, set to NA.
STORE_FORMAT_NAME	Char(60)	Yes	Store format name. If NULL, set to 'unassigned'.
CURRENCY	Char(3)	Yes	Currency under which the store operates.
STORE_TYPE	Char(6)	Yes	Indicates whether the store is a franchise (F) or company store (C).
STOCKHOLDING_IND	Char(1)	Yes	Indicates whether the store can hold stock. Valid values are Y or N.

Supplier Extract to Planning (BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rpas_supplier_extract.ksh
Description	Extracts Supplier information to Planning
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	Supplier_Fnd_ProcessFlow_From_RMS Supplier_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This process extracts supplier data to Planning on a weekly basis.

Key assumptions for this integration:

- All active, orderable supplier sites will be included in this integration each time it runs.
- Retailers will not create a Diff with an ID of 'SUP'.
- In order to meet the format required by Planning, the UDA description in this extract is hard coded to "Supplier" and does not take into account the primary language configuration in Merchandising.
- The intended targets for this integration are
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to Planning.

The batch job BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts Supplier information and writes it out
to a flat file for processing by AP and IP."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateDS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
  </end on="COMPLETED"/>
</job>
```

```
</step>
</job>
```

When the batch job BDI_RPAS_Supplier_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (Supplier_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (Supplier_Fnd_Extractor) calls BDI_FOUNDATION_SQL.SUPS_UP function to extract data from the Merchandising table SUPS to BDI outbound staging table SUPS_OUT. Only supplier sites will be extracted.
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rpas_supplier_extract.ksh to write supplier information from the SUPS_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Two separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
SUPS	Yes	No	No	No
SUPS_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
UDA_ID	Char(6)	Yes	Hardcoded 'SUP'
UDA_DESC	Char(120)	Yes	Hardcoded 'Supplier'
SUPPLIER	Char(30)	Yes	The supplier site ID.
SUP_NAME	Char(120)	Yes	The supplier site name in the primary language.

Transaction Data Extract to Planning (BDI_MFP_TranData_Tx_PF_From_RMS_JOB)

Module Name	BDI_MFP_TranData_Tx_PF_From_RMS_JOB
Description	Extracts Transaction data to Planning from RMS
Functional Area	Transactional Data
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A
Runtime Parameters	TranData_Tx_ProcessFlow_From_RMS TranData_Tx_Extractor

Design Overview

This process extracts transactional data to planning on a weekly basis, aggregating all transactions that posted in the last week, which could include transactions for previous weeks that posted late.

Key assumptions in this integration:

- Only orderable, inventoried, approved transaction items are included in the integration.
- Pack items are not included in this integration; any transactions involving pack items will be sent in terms of the pack's component items.
- Cost and retail values sent in primary currency.
- Sales sent will always be net sales. If gross sales are needed in Planning, then net sales can be combined with returns.
- Retail values will include VAT if the system option to include VAT in the stock ledger is set to include VAT so that the retail values in this integration are consistent with other data sent to planning.
- All unit values are sent in terms of the standard unit of measure for the item.
- Late posted transactions included in this integration may be for any week in the open stock ledger month, as well as any week in the previous month that posted during the week but before the previous month closed, if the month close ran during the current week.
- The intended targets for this integration are
 - Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to the target applications. The batch job BDI_MFP_TranData_Tx_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_MFP_TranData_Tx_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts information regarding transaction
data for use by the MFP application"/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicatedS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_MFP_TranData_Tx_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (Trandata_Tx_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (TranData_Tx_Extractor) calls BDI_MFP_SQL. TRAN_DATA_UP function to extract data from the Merchandising view V_BDI_MFP_TRAN_DATA to BDI outbound staging table TRAN_DATA_OUT.
- A generic BDI Downloader file creator job writes transactional information from the TRAN_DATA_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated MFP location as configured via BDI system options:
 - MFP_outboundLocation
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date
Scheduling Considerations	N/A
Pre-Processing	N/A

Schedule Information	Description
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
V_BDI_MFP_TRAN_DATA	Yes	No	No	No
TRAN_DATA_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
EOW	Date	Yes	Indicates the end of week date that the information pertains to.
ITEM	Varchar2(25)	Yes	Transaction level item only.
LOCATION	Number(10)	Yes	Could be a store or virtual warehouse.
LOC_TYPE	Varchar2(1)	Yes	Indicates if the location is a store or warehouse - S = Store; W = Warehouse.
CLEAR_IND	Number(1)	Yes	If Y, item/location is currently on clearance.
NET_SALES_REG_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 1 and sales type = R
NET_SALES_REG_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 1 and sales type = R
NET_SALES_REG_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 1 and sales type = R
NET_SALES_PROMO_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 1 and sales type = P
NET_SALES_PROMO_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 1 and sales type = P
NET_SALES_PROMO_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 1 and sales type = P
NET_SALES_CLEAR_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 1 and sales type = C

Field Name	Field Type	Required	Description
NET_SALES_CLEAR_CO ST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 1 and sales type = C
NET_SALES_CLEAR_RE TAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 1 and sales type = C
NET_SALES_REG_RETAI L_VAT_EXCL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 2 and sales type = R
NET_SALES_PROMO_R TL_VAT_EXCL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 2 and sales type = P
NET_SALES_CLR_RETAI L_VAT_EXCL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 2 and sales type = C
RETURNS_REG_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 4 and sales type = R
RETURNS_REG_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 4 and sales type = R
RETURNS_REG_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 4 and sales type = R
RETURNS_PROMO_UNI TS	Number(12,4)	No	tran_data_history.units: tran_code = 4 and sales type = P
RETURNS_PROMO_COS T	Number(20,4)	No	tran_data_history.total_cost: tran_code = 4 and sales type = P
RETURNS_PROMO_RET AIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 4 and sales type = P
RETURNS_CLEAR_UNIT S	Number(20,4)	No	tran_data_history.units: tran_code = 4 and sales type = C
RETURNS_CLEAR_COS T	Number(20,4)	No	tran_data_history.total_cost: tran_code = 4 and sales type = C
RETURNS_CLEAR_RETA IL	Number(20,4)	No	tran_data_history.total_cost: tran_code = 4 and sales type = C
REG_MARKDOWN_RETA IL	Number(20,4)	No	tran_data_history.total_retail: tran_code 13 - tran_code 14 (Markdown Cancel) - tran_code 11 (Markup)
PROMO_MARKDOWN_R ETAIL_REG	Number(20,4)	No	tran_data_history.total_retail: tran_code = 15 - if the item is not on clearance EOW
PROMO_MARKDOWN_R ETAIL_CLEAR	Number(20,4)	No	tran_data_history.total_retail: tran_code = 15 - if the item is on clearance EOW
CLEAR_MARKDOWN_RE TAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 16
WF_MARKDOWN_RETAI L	Number(20,4)	No	tran_data_history.total_retail: tran_code = 85
WF_MARKUP_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 84
SHRINK_UNITS	Number(12,4)	No	tran_data_history.units: tran_code 22
SHRINK_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code 22

Field Name	Field Type	Required	Description
SHRINK_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code 22
DEAL_INCOME_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code 6 & 7
RECEIPT_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 20 + 44
RECEIPT_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 20 + 44
RECEIPT_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 20 + 44
NON_SHRINK_ADJ_UNIT S	Number(12,4)	No	tran_data_history.units: tran_code = 23
NON_SHRINK_ADJ_COS T	Number(20,4)	No	tran_data_history.total_cost: tran_code = 23
NON_SHRINK_ADJ_RET AIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 23
DEAL_INCOME_PURCH ASES	Number(20,4)	No	tran_data_history.total_cost: tran_code 7
MARKUP	Number(20,4)	No	tran_data_history.total_retail: tran_code 11
MARKDOWN_CANCEL	Number(20,4)	No	tran_data_history.total_retail: tran_code 14
INTERCOMPANY_MARK UP	Number(20,4)	No	tran_data_history.total_retail: tran_code 17
INTERCOMPANY_MARK DOWN	Number(20,4)	No	tran_data_history.total_retail: tran_code 18
RTV_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 24
RTV_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 24
RTV_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 24
TSF_IN_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 30
TSF_IN_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 30
TSF_IN_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 30
TSF_IN_UNITS_BOOK	Number(12,4)	No	tran_data_history.units: tran_code = 31
TSF_IN_COST_BOOK	Number(20,4)	No	tran_data_history.total_cost: tran_code = 31
TSF_IN_RETAIL_BOOK	Number(20,4)	No	tran_data_history.total_retail: tran_code = 31
TSF_OUT_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 32
TSF_OUT_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 32

Field Name	Field Type	Required	Description
TSF_OUT_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 32
TSF_OUT_UNITS_BOOK	Number(12,4)	No	tran_data_history.units: tran_code = 33
TSF_OUT_COST_BOOK	Number(20,4)	No	tran_data_history.total_cost: tran_code = 33
TSF_OUT_RETAIL_BOOK	Number(20,4)	No	tran_data_history.total_retail: tran_code = 33
RECLASS_IN_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 34
RECLASS_IN_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 34
RECLASS_IN_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 34
RECLASS_OUT_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 36
RECLASS_OUT_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 36
RECLASS_OUT_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 36
TSF_IN_UNITS_ICK	Number(12,4)	No	tran_data_history.units: tran_code = 37
TSF_IN_COST_ICK	Number(20,4)	No	tran_data_history.total_cost: tran_code = 37
TSF_IN_RETAIL_ICK	Number(20,4)	No	tran_data_history.total_retail: tran_code = 37
TSF_OUT_UNITS_ICK	Number(12,4)	No	tran_data_history.units: tran_code = 38
TSF_OUT_COST_ICK	Number(20,4)	No	tran_data_history.total_cost: tran_code = 38
TSF_OUT_RETAIL_ICK	Number(20,4)	No	tran_data_history.total_retail: tran_code = 38
INTERCOMPANY_MARGIN	Number(20,4)	No	tran_data_history.total_retail: tran_code = 39
TSF_RECEIPT_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 44
TSF_RECEIPT_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 44
TSF_RECEIPT_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 44
RTV_RESTOCK_FEE	Number(20,4)	No	tran_data_history.total_cost: tran_code = 65
FRANCHISE_SALES_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 82
FRANCHISE_SALES_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 82
FRANCHISE_SALES_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 82

Field Name	Field Type	Required	Description
FRANCHISE_RETURNS_UNITS	Number(12,4)	No	tran_data_history.units: tran_code = 83
FRANCHISE_RETURNS_COST	Number(20,4)	No	tran_data_history.total_cost: tran_code = 83
FRANCHISE_RETURNS_RETAIL	Number(20,4)	No	tran_data_history.total_retail: tran_code = 83
FRANCHISE_RESTOCK_FEE	Number(20,4)	No	tran_data_history.total_cost: tran_code = 86

UDA Extract to Planning (BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB)

Module Name	BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rpas_uda_extract.ksh
Description	Extracts LOV Type UDA information to Planning
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	UdaAndUdaValues_Fnd_ProcessFlow_From_RMS Uda_Fnd_Extractor UdaValues_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This process extracts its UDA data to Planning on a weekly basis.

Key assumptions for this integration:

- The full set of user defined attributes (UDAs) is included in this integration each time it runs.
- Only list of value type UDAs will be included in the integration.
- The intended targets for this integration are
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to Planning. The batch job BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts LOV Type UDA information and writes it
out to a flat file for processing by AP and IP."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
```

```

    <properties>
      <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
      <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
      <property name="predicateDS" value="RmsDBDS"/>
      <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
    </properties>
  </batchlet>
  <end on="COMPLETED"/>
</step>
</job>

```

When the batch job BDI_RPAS_UdaAndUdaValues_Fnd_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an and-of-week date, it invokes a BDI process flow (UdaAndUdaValues_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor jobs (Uda_Fnd_Extractor, UdaValues_Fnd_Extractor) call respective BDI_FOUNDATION_SQL functions to extract data from Merchandising tables UDA and UDA_VALUES to BDI outbound staging tables UDA_OUT and UDA_VALUES_OUT.
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rpas_uda_extract.ksh to write UDA information from the UDA_OUT and UDA_VALUES_OUT tables into a comma-delimited flat file, which will be consumed by the target applications. Only LOV type UDAs will be extracted. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated locations as configured via BDI system options:
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
UDA	Yes	No	No	No
UDA_VALUES	Yes	No	No	No
UDA_OUT	Yes	Yes	No	Yes
UDA_VALUES_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
UDA_ID	Number(5)	Yes	The ID of the UDA assigned to the item.
UDA_DESC	Char(120)	Yes	The description of the UDA (for example, Fabric Content).
UDA_VALUE	Number(5)	Yes	The ID of the UDA value for the UDA assigned to the item.
UDA_VALUE_DESC	Char(250)	Yes	The description of the UDA value (for example, Cotton).

UDA Item Extract to Planning and Forecasting (BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB)

Module Name	BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB bdi_merch_extract_to_file_wrapper.sh bdi_rdf_itemuda_extract.ksh
Description	Extracts information for LOV type of UDAs to Planning and Forecasting
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job, shell scripts
Catalog ID	N/A
Runtime Parameters	ItemHdrAndUdaltemLov_Fnd_ProcessFlow_From_RMS ItemHdr_Fnd_Extractor UdaltemLov_Fnd_Extractor Database connection, download file location, filename, trigger filename

Design Overview

This process extracts user-defined attributes (UDAs) assigned to item to Planning and Forecasting on a weekly basis.

Key assumptions for this integration:

- Only list of value (LOV) type UDAs will be included.
- Both forecasted and non-forecasted items are included in this extract, with the forecast flag included.
- Planning and Forecasting can only support a specific UDA being associated with an item once. Merchandising has a configuration that allows the same UDA to be associated with an item more than one time. However, when implementing with Planning or Forecasting, this should be avoided for LOV-type UDAs to prevent issues with interpreting the data. If more than one is associated with the item, then only the last UDA with a particular ID will be visible in Planning and Forecasting.
- The intended targets for this integration are
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting
 - Assortment & Item Planning for Fashion/Softlines Cloud Service and Assortment & Item Planning Enterprise Edition Cloud Service (referred to jointly as APCS)

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement from Merchandising to the target applications. The batch job BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RDF_UdaItemLov_Fnd_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts UDA item LOV information and writes
it out to a flat file for processing by RDF."/>
  </properties>
  <step id="batchlet-step">
    <batchlet ref="BDIInvokerBatchlet">
      <properties>
        <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
        <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
        <property name="predicateBS" value="RmsDBDS"/>
        <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
      </properties>
    </batchlet>
    <end on="COMPLETED"/>
  </step>
</job>
```

When the batch job BDI_RDF_UdaltemLov_Fnd_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (ItemHdrAndUdaltemLov_Fnd_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to the target applications:

- Extractor jobs (ItemHdr_Fnd_Extractor, UdaltemLov_Fnd_Extractor) call respective BDI_ITEM_SQL functions to extract data from Merchandising tables to BDI outbound staging tables ITEM_HDR_OUT and UDA_ITEM_LOV_OUT.
- Downloader file creator job calls the wrapper script, bdi_merch_extract_to_file_wrapper.sh, to set the runtime parameters on environment variables. This script will then call bdi_rdf_itemuda_extract.ksh to write LOV type of UDA information from the ITEM_HDR_OUT and UDA_ITEM_LOV_OUT tables into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to

signal that the extract process was successful. The data file and the trigger file are then sent to the target applications.

- The downloaded data file and trigger file are written to designated locations as configured through BDI system options:
 - RDF_outboundLocation
 - AP_outboundLocation
 - IP_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
ITEM_MASTER	Yes	No	No	No
CLASS	Yes	No	No	No
SUBCLASS	Yes	No	No	No
DIFF_GROUP_HEAD	Yes	No	No	No
DIFF_IDS	Yes	No	No	No
SYSTEM_OPTION	Yes	No	No	No
UDA_ITEM_LOV	Yes	No	No	No
UDA	Yes	No	No	No
UDA_VALUES	Yes	No	No	No
UDA_ITEM_LOV_OUT	Yes	Yes	No	Yes
ITEM_HDR_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DATA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CTL	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
ITEM	Char(25)	Yes	The ID of the item.
UDA_ID	Number(5)	Yes	The ID of the UDA assigned to the item.
UDA_DESC	Char(120)	Yes	The description of the UDA (for example, Fabric Content)
UDA_VALUE	Number(5)	Yes	The ID of the UDA value for the UDA assigned to the item.
UDA_VALUE_DESC	Char(250)	Yes	The description of the UDA value (for example, Cotton).
FORECAST_IND	Char(1)	Yes	Indicates whether or not the item is to be forecasted. Valid values are Y or N.

Weekly Sales Extract to Forecasting (BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB)

Module Name	BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB
Description	Extracts weekly sales information to Forecasting
Functional Area	Foundation
Module Type	Integration
Module Technology	BDI job
Catalog ID	N/A
Runtime Parameters	WeeklySales_Tx_ProcessFlow_From_RMS WeeklySales_Tx_Extractor

Design Overview

This process extracts weekly sales for use by Forecasting on a weekly basis. It sends only the sales from the last week.

Key assumptions for this integration:

- This integration sends gross sales. Returns are not netted out of the sales values.
- Warehouse issues are not included in this integration. Only sales for stores.
- Only forecasted items are included in this integration.
- The intended targets for this integration are
 - Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting

This process utilizes BDI (Bulk Data Integration) to facilitate the bulk data movement to the target applications.

The batch job BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB is defined in the Merchandising JOS batch job admin as follows:

```
<job id="BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB" version="1.0" xmlns="http://
xmlns.jcp.org/xml/ns/javaee">
  <properties>
    <property name="description" value="Extracts weekly sales information for use by
```

```

the RDF application"/>
</properties>
<step id="batchlet-step">
  <batchlet ref="BDIInvokerBatchlet">
    <properties>
      <property name="bdiProcessFlowUrl" value="#SysOpt.bdiProcessFlowUrl"/>
      <property name="bdiProcessFlowCredential"
value="#SysOpt.bdiProcessFlowUrlUserAlias"/>
      <property name="predicateDS" value="RmsDBDS"/>
      <property name="predicateFunction"
value="RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL"/>
    </properties>
  </batchlet>
  <end on="COMPLETED"/>
</step>
</job>

```

When the batch job BDI_RDF_WeeklySales_Tx_PF_From_RMS_JOB is executed, a batchlet (BDIInvokerBatchlet) starts the execution flow. It calls a PLSQL function (RMS_BATCH_STATUS_SQL.GET_EOW_RUN_SIGNAL) to ensure the process flow is only executed on an end-of-week date. If the vdate is an end-of-week date, it invokes a BDI process flow (WeeklySales_Tx_ProcessFlow_From_RMS) to perform a series of steps to extract, download, and transport the downloaded files to target applications:

- Extractor job (WeeklySales_Tx_ExtractorJob) calls BDI_RDF_SQL.WEEKLY_SALES_UP function to extract data from a Merchandising view V_BDI_RDF_WEEKLY_SALES to outbound staging table WEEKLY_SALES_OUT.
- A generic BDI Downloader file creator job writes weekly sales information from the WEEKLY_SALES_OUT table into a comma-delimited flat file, which will be consumed by the target applications. A zero-byte trigger file is also generated to signal that the extract process was successful. Separate copies of the data file and the trigger file are sent to the target applications.
- The downloaded data files and trigger files are written to designated location as configured via BDI system options:
 - RDF_outboundLocation

Scheduling Constraints

Schedule Information	Description
Processing Cycle	End of Day
Frequency	Scheduled daily but files will only be generated weekly on End of Week date.
Scheduling Considerations	N/A
Pre-Processing	N/A
Post-Processing	N/A
Threading Scheme	N/A

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
V_BDI_RDF_WEEKLY_SALES	Yes	No	No	No
WEEKLY_SALES_OUT	Yes	Yes	No	Yes
BDI_DWNLDR_IFACE_MOD_DA TA_CTL	Yes	No	No	No
BDI_DWNLDR_IFACE_DATA_CT L	Yes	No	No	No

Integration Contract

The flat file will contain the following information:

Field Name	Field Type	Required	Description
ITEM	Varchar2(25)	Yes	Indicates the item.
STORE	Number(10)	Yes	Indicates the store.
EOW_DATE	Date	Yes	Indicates the end of week date for which the data applies.
SALES_UNITS	Number(12,4)	No	This value will be the total sales units for the item/location for the week.
SALES_TYPE	Varchar2(1)	Yes	Indicates the sales type. For example, R (Regular Sales), P (Promotional Sales) or C (Clearance Sales).

Sales Audit

The purpose of Sales Audit is to accept transaction data from point-of-sale (POS) and order management (OMS) solutions and move the data through a series of processes that culminate in "clean" data. Data that Sales Audit finds to be inaccurate is brought to the attention of the auditors who can use the features in Sales Audit to correct the exceptions.

For more information on Sales Audit processing see *Merchandising Operations Guide Volume 1*.

This section contains details about the following integration processes used to export data from Sales Audit to other solutions:

- [Download from Sales Audit to Account Clearing House \(ACH\) System \(saexpach\)](#)
- [Download of Escheated Vouchers from Sales Audit for Payment \(saescheat\)](#)
- [Export DSD and Escheatment from Sales Audit to Invoice Matching \(saexpim\)](#)
- [Export from Sales Audit to Oracle Retail Insights \(saexpdw\)](#)
- [Export Inventory Reservation/Release for In Store Customer Order & Layaway Transactions \(saordinvexp\)](#)
- [Export of POS transactions from Sales Audit to Merchandising \(saexprms\)](#) (file-based integration)

- [Export of POS Transactions from Sales Audit to Merchandising Based on Direct Table Load \(saexprocsales\)](#) (table-based integration)
- [Export of Revised Sale/Return Transactions from ReSA to SIM/SIOCS \(saexpsim\)](#)
- [Export of Revised Sale/Return Transactions from Sales Audit to Store Inventory Management \(saexport_sales_to_sim\)](#)
- [Export to Universal Account Reconciliation System from Sales Audit \(saexpuar\)](#)
- [Extract of POS Transactions by Store/Date from Sales Audit for Web Search \(ang_sapngen\)](#)
- [Post User Defined Totals from Sales Audit to General Ledger \(saexpgl\)](#)

Download from Sales Audit to Account Clearing House (ACH) System (saexpach)

Module Name	saexpash.pc
Description	Download from Sales Audit to Account Clearing House (ACH) System
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA03
Wrapper Script	rmswrap_out.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This module will post store/day deposit totals to the SA_STORE_ACH table and bank deposit totals for a given day in a file formatted for export to an ACH (Account Clearing House). The ACH export deviations from the typical Sales Audit export in that store/days must be exported even though errors may have occurred for a given day or store (depending on the unit of work defined), and also, the store/day does not need to be closed for the export to occur. The nature of the ACH process is such that as much money as possible must be sent as soon as possible to the consolidating bank. Any adjustments to the amount sent can be made using the sabnkach screen in the online system.

Deposits for store/days that have not been Fully (F) loaded will not be transferred to the consolidating bank. After they are fully loaded, their deposits will be picked up by the next run of this program.

Restart/Recovery

This module is in two distinct parts, with two different logical units of work. Thus, restart/recovery has to be implemented so that the first part does not get reprocessed in case the program is being restarted. Details on the implementation follow.

The first driving cursor in this module retrieves a store/day to generate ACH totals. Once the first cursor is complete, the second retrieves bank locations by account numbers.

The first Logical Unit of Work (LUW) is defined as a unique store/day combination. Records will be fetched, using the first driving cursor, in batches of commit_max_ctr, but processed and committed one store/day at a time.

The first driving cursor will fetch all store/days that have been Fully Loaded (F), whose audit status is Audited (A), HQ Errors Pending (H), or Store Errors Pending (S) and that are ready to be exported to ACH. Before processing starts, a write lock is obtained using `get_lock ()`. This driving cursor only fetches store/days with a `sa_export_log.status` of `SAES_R`. After a store/day is processed, `sa_export_log.status` is set to `SAES_P` so that this store/day will not be selected again if the program is restarted. The commit is performed using `retek_force_commit` after each store/day has been processed and `sa_export_log` updated, so as to release the lock.

In case a store/day could not be processed due to locking, the store/day information is placed on a list (called locked store/day list) and the next store/day is processed. This list is kept in memory and is available only during processing. If the store for a store/day obtained from the first driving cursor, is on the locked store/day list, then this store/day cannot be processed. This is the case because there is a data dependency such that data from a particular store/day is dependent on data for the same store but at an earlier date. Thus, if a store/day cannot be processed, then subsequent store/days for the same store cannot be processed either. After the driving cursor returns no more data, the program attempts to process each store/day on the list two more times. If the store/day is still locked, then it is skipped entirely and a message is printed to the error log.

The second LUW is a bank account number. Again, records will be fetched in batches of `commit_max_ctr`. The second driving cursor cannot retrieve information by the LUW because it is possible for the store's currency to be different from the local bank's currency. In that case, a currency conversion is needed.

For each store/day, the query should retrieve the required ACH transfer. The latter is determined by adding the estimated deposit for the next day, the adjustment to the estimate for the current day, and any manual adjustment to the estimate.

Since a store can be associated with different accounts at different banks, only accounts that are consolidated should be retrieved. Since it is possible for the local bank to be in a different country than the consolidating bank, the currency of the partner should also be fetched.

Since processing is dependent on the type of account at the RDFI, the account type should be fetched by this cursor.

Due to differences in transaction processing in cases when the bank is outside the United States, the partner's country should also be fetched. The results of the query should be sorted by partner country. The results of the query should also be ordered by accounts.

Security Considerations

The fact that this program automates the transfer of funds on behalf of the user makes it a likely target for electronic theft. It must be made clear that the responsibility of electronic protection lies with the users themselves.

Following are some tips and recommendation to users:

- A specific user should be used to run the program. This user would be the only one (or one of a few) who has access to this program.
- The umask for this user should be set up so as to prevent other users from reading/writing its files. This would ensure that when the output file is created, it would not be accessible to other users.
- The appropriate permissions should be set up on the directory, which holds the ACH files. The most restrictive decision would be to not allow any other user to view the contents of the directory.

- A secure means of communication should be implemented for transferring the file from where it has been created to the ACH network. This may be done through encryption, or by copying the file to a disk and trusting the courier to deliver the files intact.
- The ACH network needs to be secure.

I/O Specification

Integration Type	Download from Sales Audit
File Name	ACH_ appended with the consolidating routing number, consolidating account number, and current system date.
Integration Contract	IntCon000040

Output File

Table 6-22 Output File

Record Name	Field Name	Field Type	Default Value	Description
ACH File Header	Section No.	Number(3)	101	Constant number.
	Console Route No	Number(10)	N/A	The routing number of the consolidating bank.
	Sender ID	Char(10)	N/A	ID used by the Originator to identify itself.
	Current Date	Char(6)	N/A	Vdate in YYMMDD format.
	Day Time	Char(4)	N/A	Time of file creation in HH24MM format.
	File Header No.	Number(7)	0094101	Constant number.
	Console Bank Name	Char(23)	N/A	Name of the Originating Financial Depository Institution.
	Company Name	Char(23)	N/A	The name of the company name.
ACH CCD Batch Header	Ref Code	Char (8)	N/A	Reference code.
	Section No.	Number(4)	5225	Constant number.
	Company Name	Char(16)	N/A	The name of the company.
	Comp Disc Data	Char(20)	NULL	Any kind of data specific to the company.
	Comp Id	Char(10)	N/A	Alphanumeric code to identify the company.
	CCD Header Id	Char(3)	CCD	Constant value.
	Comp Entry Desc	Char(10)	CONSOL	A short description from the Originator about the purpose of the entry.
	Tomorrow	Char(6)	N/A	Vdate+1 in YYMMDD format.
	Tomorrow	Char(6)	N/A	Vdate+1 in YYMMDD format.
	Settle Date	Char(3)	NULL	This is inserted by receiving the ACH Operator.
	Reserved	Number(1)	1	Constant number.
	Odfi Id	Number(8)		8-digit routing number of the ODFI.

Table 6-22 (Cont.) Output File

Record Name	Field Name	Field Type	Default Value	Description
ACH CBR Batch Header	Batch No	Number(7)		Batch number.
	Section No.	Number(4)	5225	Constant number.
	Company Name	Char(16)	N/A	The name of the company.
	Reserved	Char(3)	FV1	Constant value.
	Exch Rate	Number(15)		Exchange rate for the specified currency.
	Reserved	Char(2)	US	Constant value.
	Comp Id	Char(10)		Alphanumeric code to identify the company
	CBR Header Id	Char(3)	CBR	Constant value.
	Comp Entry Desc	Char(10)	"CONSOL "	A short description from the Originator about the purpose of the entry.
	Partner Curr Code	Char(3)	N/A	Code identifying the currency the partner uses for business transactions.
	Reserved	Char(3)	USD	Constant value.
	Tomorrow	Char(6)	N/A	Vdate+1 in YYMMDD forma.
	Settle Date	Char(3)	NULL	This is inserted by the receiving ACH Operator.
	Reserved	Number(1)	1	Constant number.
	Odfi Id	Number(8)	N/A	8-digit routing number of the ODFI.
ACH CCD Entry	Batch No	Number(7)	N/A	Batch number.
	Section No.	Number(1)	6	Constant number.
	Trans Code	Char(2)		Code used to identify the type of debit and credit. Value accepted are 27 and 37.
	Routing No	Number(9)		Routing number for the bank account.
	Acct No	Char(17)		Account number of the bank.
	Deposit	Number(10)		The amount involved in the transaction* 10000 (4 implied decimal places).
	Id	Char(15)	Null	Identification number. Optional field containing a number used by the Originator to insert its own number for tracing purposes.
	Store Name	Char(22)		Name of the local store.
	Disc Data	Char(2)	Null	Discretionary data. Any kind of data specific to the transaction.
	Reserved	Number(1)	0	Constant number.

Table 6-22 (Cont.) Output File

Record Name	Field Name	Field Type	Default Value	Description
ACH CBR Entry	Trace No	Number(15)		Used to uniquely identify each entry within a batch. The first 8 digits contain the routing number of the ODFI and the other 7 contains a sequence number.
	Section No.	Number(1)	6	Constant number.
	Trans Code	Char(2)	N/A	Code used to identify the type of debit and credit. Values accepted are 27 and 37.
	Routing No	Number(9)	N/A	Routing number for the bank account.
	Acct No	Char(17)	N/A	Account number of the bank
	Deposit	Number(10)	N/A	The amount involved in the transaction* 10000 (4 implied decimal places).
	Id	Char(15)	NULL	Identification number. Optional field containing a number used by the Originator to insert its own number for tracing purposes.
	Store Name	Char(22)	N/A	Name of the local store.
	Disc Data	Char(2)	NULL	Discretionary data. Any kind of data specific to the transaction.
	Reserved	Number(1)	1	Constant number.
ACH CBR Addendum	Trace No	Number(15)	N/A	Used to uniquely identify each entry within a batch. The first 8 digits contain the routing number of the ODFI and the other 7 contains a sequence number.
	Section No.	Number(3)	701	Constant number.
	Payment Info	Char(80)	Null	Payment related information.
	Reserved	Number(4)	0001	Constant number
	Trace Seq No	Number(7)	N/A	Sequence number part of the Trace Number of the entry record to which this addendum is referring.
ACH Batch Control	Section No.	Number(4)	8225	Constant number.
	Batch Line Count	Number(6)	N/A	The number of entries and addenda in the batch.
	Hash Count	Number(10)	N/A	Sum of the RDFI IDs in the detail records.
	Total Batch Debit	Number(12)	N/A	Contains the accumulated debit and debit for the file * 10000 (4 implied decimal places).
	Total Batch Credit	Number(12)	N/A	Contains the accumulated credit and credit for the file * 10000 (4 implied decimal places).

Table 6-22 (Cont.) Output File

Record Name	Field Name	Field Type	Default Value	Description
ACH File Control	Comp Id	Char(10)	N/A	An alphanumeric code identifying the company.
	Auth	Char(19)	Null	Message Authentication Code. The first 8 characters represent a code from the Data Encryption Standard (DES) algorithm. The remaining eleven characters are blanks.
	Reserved	Char(6)	Null	Reserved.
	ODFI Id	Number(8)	N/A	8-digit routing number of the ODFI.
	Batch No	Number(7)	N/A	Batch number.
	Section No.	Number(1)	9	Constant number.
	Batch count	Number(6)	N/A	The number of batches sent in the file.
	Block count	Number(6)	N/A	The number of physical blocks in the file, including both File Header and File Control Records. This is the ceiling of the number of records divided by the blocking factor, which is 10.
	Entry count	Number(8)	N/A	The number of entries and addenda in the file.
	Total hash count	Number(10)	N/A	Sum of the Entry Hash fields on the Batch Control Records.
ACH Completed Block	Total file debit	Number(12)	N/A	Contains the accumulated debit and debit for the file * 10000 (4 implied decimal places).
	Total file credit,	Number(12)	N/A	Contains the accumulated credit and credit for the file * 10000 (4 implied decimal places).
	Reserved	Char(39)	NULL	Reserved.
	End string	Char(94)	N/A	Mark the end of the file: a string of 94 '9' characters. The number of end lines with a string of 94 '9' characters is identified by the following equation: $10 - \text{mod}(\text{number of lines in the file}, 10)$.

Design Assumptions

N/A

Download of Escheated Vouchers from Sales Audit for Payment (saescheat)

Module Name	saescheat.pc

Description	Download of Escheated Vouchers from Sales Audit for Payment
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA05
Wrapper Script	rmswrap.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The laws of individual states and countries may require a retailer to return monies for aged, unclaimed gift certificates, and vouchers. This process is called escheatment. This program writes records for this data to tables that are read into Invoice Matching by the program saexpim.pc. The data can then be sent as invoices approved for payment to a financial application.

The saescheat batch program will set the status of vouchers that have met certain state's escheats rules or have expired to the proper status and produce a total for later export to Invoice Matching. The rules for escheatment are defined on the sa_escheatment_options table.

Restart/Recovery

The logical unit of work is a store/day. The program commits when the number of store/day records processed has reached the commit_max_ctr.

I/O Specification

Integration Type	Download from Sales Audit
File Name	N/A
Integration Contract	IntCon000039

Design Assumptions

N/A

Export DSD and Escheatment from Sales Audit to Invoice Matching (saexpim)

Module Name	saexpim.pc
Description	Export DSD and Escheatment from Sales Audit to Invoice Matching
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA04
Wrapper Script	rmswrap.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this program is to support interfacing invoices from Direct Store Delivery and Escheatment sales audit transactions to the Invoice Matching application. Direct Store Delivery invoices refer to products or services that are delivered to the store and paid for at the store. This program will take DSD invoices that have been staged to the transaction header table by the saimptlog.pc program and move them into the invoice header table. All DSD transactions will be assumed paid. They can be assumed received if there is a proof of delivery number listed on them. Transactions with a vendor invoice ID or a proof of delivery number should be matched to any existing invoice in the invoice header, and that invoice updated with the new information being interfaced. Invoices that do not match an existing invoice in the invoice head table will need to be inserted. Each transaction will be exported to the invoice head table only once.

The Sales Audit Transaction type used to identify invoices for Direct Store Delivery transactions will be "Paid Out". The Paid Out transaction has a code of 'PAIDOU'. The Sales Audit sub-transaction types will be used to identify whether the invoice is an "Expense Vendor Payout" or a "Merchandise Vendor Payout". The codes are 'EV' for Expense Vendor Payout and 'MV' for Merchandise Vendor Payout. Any Paid Out transaction with a sub transaction type of Expense Vendor will create a non-merchandise invoice and cause a record to be written to the invoice non-merchandise table. Sales Audit will store non-merchandise codes in the reason_code field on sa_tran_head. Valid values for these reason codes should correspond to the codes stored on the non_merch_code_head table.

In addition to DSD invoices, this program will also interface Escheatment totals to Invoice Matching. Escheatment is the process where an unredeemed gift certificate/voucher or credit voucher will, after a set period of time, be paid out as income to the issuing retailer, or in some states, the state receives this escheatment income. Sales Audit will be the governing system that determines who receives this income, but Invoice Matching will send the totals, with the related Partner, to an Accounts Payable system. Escheatment information will be stored on the Sales Audit SA_TOTALS table and will be used to create non-merchandise invoices in Invoice Matching. These invoices will be assumed not paid.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, updated, and inserted based on the commit_max_ctr specified on the restart_control table. Only two commits will be done, one to establish the store/day lock and another at the end, to release the lock after a store/day has been completely processed. In case of failure, all work done will be rolled back to the point right after the call to get_lock and releases the lock. Thus, the rollback segment should be large enough to hold all inserts into sa_exported for one store_day.

Integration Contract

Integration Type	Download from Sales Audit
File Name	N/A
Integration Contract	IntCon00004 INVC_HEAD table

Design Assumptions

N/A

Export from Sales Audit to Oracle Retail Analytics (saexport_sales_to_dw)

Module Name	saexport_sales_to_dw.ksh
Description	Export from ReSA to Oracle Retail Analytics
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	KSH
Catalog ID	RSA02
Wrapper Script	N/A

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to fetch all sales and return transactions that do not have Retail Analytics errors from the Oracle Retail Sales Audit (ReSA) database tables and export them into the interface tables SA_EXPDW_RDW*. Retail Data Store (RDS) views have been written on the top of these interface tables so that the Oracle Retail Analytics application can fetch the exported sales and return data through Retail Data Store (RDS). The data will be exported at the store-day level. If the transaction has a status of Deleted and if it has been previously exported, a reversal of the transaction will be exported.

The batch has been designed to integrate with either Oracle Retail Insights or a 3rd Party Retail Analytics system. For integration with Oracle Retail Insights, this batch will export RDWT and RDWF data to three tables at the store day level:

- SA_EXPDW_RDWT_HEAD – Transaction head and transaction items are exported to this table.
- SA_EXPDW_RDWT_DETAIL – Transaction discounts are exported to this table.
- SA_EXPDW_RDWF_DETAIL – Form of payment (tender) is exported to this table.

For integration with a third-party Retail Analytics system, this batch will *additionally* export RDWS and RDWC data to two more tables at the store day level:

- SA_EXPDW_RDWS_DETAIL – Store Totals are exported to this table.
- SA_EXPDW_RDWC_DETAIL – Cashier/Register Totals are exported to this table.

The export of RDWS and RDWC data to SA_EXPDW_RDWS_DETAIL and SA_EXPDW_RDWC_DETAIL is controlled by a Y/N flag defined in the MERCH_BATCH_PARAM table with a batch_name of SAEXPDW_EXPORT_JOB and param_key of EXPORT_DW_S_C. The default value of the flag is N.

For customers using Oracle Retail Insights, EXPORT_DW_S_C should be set to N to prevent the batch from writing to SA_EXPDW_RDWS_DETAIL and SA_EXPDW_RDWC_DETAIL, because Oracle Retail Insights doesn't need the RDWS and RDWC data.

For customers using a third-party Retail Analytics and requiring Store Totals or Cashier/Register Totals, the EXPORT_DW_S_C flag needs to be set to Y to extract RDWS and RDWC data.

Note

This batch can run in two modes – trickle mode and batch mode. If Y is passed as a parameter while running the batch, then the batch runs in trickle mode. If N or the no parameter is passed, it runs in normal batch mode.

Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
SA_EXPDW_RDWT_HEAD	No	Yes	No	No
SA_EXPDW_RDWT_DETAIL	No	Yes	No	No
SA_EXPDW_RDWF_DETAIL	No	Yes	No	No
SA_STORE_DAY	Yes	No	No	No
SA_EXPORT_LOG	Yes	No	Yes	No
SA_EXPORTED	Yes	Yes	Yes	No
SA_EXPORTED_REV	Yes	No	No	No
SA_TRAN_HEAD	Yes	No	No	No
SA_TRAN_HEAD_REV	Yes	No	No	No
SA_TRAN_ITEM	Yes	No	No	No
SA_TRAN_ITEM_REV	Yes	No	No	No
SA_TRAN_DISC	Yes	No	No	No
SA_TRAN_DISC_REV	Yes	No	No	No
SA_TRAN_TENDER	Yes	No	No	No
SA_TRAN_TENDER_REV	Yes	No	No	No
SA_VOUCHER	Yes	No	No	No
SA_STORE_PRICE_HIST_TEMP	Yes	Yes	No	No
ORDCUST	Yes	No	No	No
SA_EXPDW_RDWS_DETAIL	No	Yes	No	No
SA_EXPDW_RDWC_DETAIL	No	Yes	No	No
SA_TOTAL	Yes	No	No	No
SA_TOTAL_HEAD	Yes	No	No	No
SA_TOTAL_USAGE	Yes	No	No	No
SA_HQ_VALUE	Yes	No	No	No
SA_STORE_VALUE	Yes	No	No	No
SA_SYS_VALUE	Yes	No	No	No
SA_POS_VALUE	Yes	No	No	No
SA_STORE_EMP	Yes	No	No	No
SA_BALANCE_GROUP	Yes	No	No	No
SA_ERROR	Yes	No	No	No
SA_ERROR_IMPACT	Yes	No	No	No
IF_ERRORS	No	Yes	No	No
SA_EXPDW_WC_WS_TEMP	Yes	Yes	Yes	Yes

Design Assumptions

N/A

Export from Sales Audit to Oracle Retail Insights (saexpdw)

Module Name	saexpdw.pc
Description	Export from Sales Audit to Oracle Retail Analytics
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA02
Wrapper Script	batch_resa2dw.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to fetch all sales and return transactions that do not have Retail Analytics errors from the Sales Audit database tables for transmission to the Oracle Retail Analytics application. The data will be sent at the store day level. If the transaction has a status of Deleted, and if it has been previously Transmitted, a reversal of the transaction will be sent.

Note

This batch program can be run in two modes - trickle mode and batch mode. If 'Y' is passed as a parameter while running the batch program, then the batch runs in trickle mode. If 'N' or no parameter is passed, it runs in normal batch mode.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, updated, and inserted based on the commit_max_ctr. Only two commits will be done: one to establish the store/day lock and another at the end, to release the lock after a store/day has been completely processed. The RDWT, RDWF, RDWS, and RDWC formatted output files will be created with temporary names and renamed just before the end of store/day commit.

In case of a failure, all the work done will be rolled back to the point right after the call to get_lock() and the lock is released. Thus, the rollback segment should be large enough to hold all inserts into sa_exported for one store/day.

I/O Specification

Integration Type	Download from Sales Audit

File Name	RDWT_ appended with store number, business date, and system date. RDWF_ appended with store number, business date, and system date. RDWS_ appended with store number, business date, and system date. RDWC_ appended with store number, business date, and system date.
Integration Contract	IntCon000041 (RDWT) IntCon000156 (RDWF) IntCon000157 (RDWS) IntCon000158 (RDWC)

Four output files will be created for each store_day:

- RDWT - Transaction File
- RDWF - Form of Payment (Tender) file
- RDWS - Store Totals output file
- RDWC - Cashier output File

Each output file is converted into a format for loading into Retail Analytics by the resa2dw Perl script.

Sales Audit - File Layout - Retail Analytics

- File layouts for the interface between sales audit and Retail Analytics.
- Char fields are left justified and blank filled.
- Number fields are right justified and zero filled. They can contain only numbers.
- Numeric fields are left justified and blank filled. They can contain only numbers.

RDWT File

Table 6-23 RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Type Definition	Char(4)	RDWT	Identifies file as 'Retail Analytics Transaction file'	Yes
	File Create Date	Number(14)	create date	Date file was written by external system. Format YYYYMMDDHH24 MISS	Yes
Transaction Header	File Type Record Descriptor	Char(5)	THEAD	Identifies transaction record type	

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	Business date	Number(8)		Format YYYYMMDD (Note, This is the date the Retail Analytics will consider the transaction date)	Yes
	Transaction Date	Number(14)	transaction date	Date sale/return transaction was processed at the POS. Format YYYYMMDDHH24MISS (Note, the Retail Analytics only uses the HH24MI part of this date)	Yes
	Location	Number(10)	specified by external system	Store or warehouse identifier. This value is now being determined based on either the Account for Sale or Account for Return system option.	Yes
	Register ID	Char(5)		The register identifier	Yes, -1 for null
	Banner ID	Char(4)		The unique identifier of the banner.	Yes, -1 for null
	Line Media ID	Char(10)		The identifier of the media for the order line. For non-merchandise items, such as Shipping & Handling, Service Lines and gift certificates, the media code will be that of the order line it's associated.	Yes, -1 for null
	Selling Item ID	Char(25)		The unique identifier of a selling item.	Yes, -1 for null
	Customer Order Header ID	Char(48)		The unique identifier of a customer order.	Yes, -1 for null

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Customer Order Line ID	Char(30)		The identifier of a customer order line. For a Value Added Service, like monogramming, this will be the line number for the item which the service was applied.	Yes, -1 for null
	Customer Order Create Date	Char(8)		The date when the customer order was created/placed.	Yes, -1 for null
	Cashier Identifier	Char(10)		The cashier number. This will be the unique employee number.	Yes, -1 for null
	Salesperson Identifier	Char(10)		The salesperson number. This will be the unique employee number.	Yes, -1 for null
	Customer ID Type	Char(6)		The type of ID number used by this customer.	Yes, -1 for null
	Customer ID Number	Char(16)		Customer id associated with the transaction.	Yes, -1 for null
	Transaction Number	Number(10)		The unique transaction reference number generated by the POS.	Yes
	Original Register ID	Char(5)		Register ID of the original transaction.	Yes for a transaction type of 'PVOID'.
	Original Transaction Number	Number(10)		Transaction number of the original transaction.	Yes for a transaction type of 'PVOID', 'EEXCH' and 'RETURN'
	Transaction Header Number	Numeric(20)		Unique reference used within sales audit to represent the date/store/register/tran_no	No
	Revision number	Number(3)		Number used to identify the version of the transaction being sent.	Yes

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Sales Sign	Char(1)	'P' - positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Transaction Type	Char(6)		Transaction type code	Yes
	Sub Transaction Type	Char(6)		The Sub Transaction type	Yes, -1 for null
	Retail Type	Char(1)	'R'egular, 'P'romo, or 'C'learance		Yes
	Item_Seq_No	Number(4)		The order in which items were entered during the transaction.	No
	Employee Number (Cashier)	Char(10)		Employee identification number. This will only be populated if the sub transaction type is 'EMP'.	Yes, -1 for null
	Receipt Indicator	Char(1)		Flag that identifies returns that have been processed without a receipt. This field will only be populated if the transaction type is 'RETURN'.	No
	Reason Code	Char(6)		A reason is required with a Paid In/Out transaction type, and optional with a return transaction.	Yes, -1 for null
	Vendor number	Number(10)		This will only get populated when the paid in code is Expense Vendor	No
	Item Type	Char(6)	item type identifier	Type of item sold, 'ITEM', 'REF', 'GCN' (gift certificate number), or 'NMITEM'	No
	Item	Char(25)		ID number of the item or gift certificate.	No. Required if Item Type is not null.

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Ref Item	Char(25)		Sub-transaction level item	No. Also, this field can never be populated without a transaction level item in the item field.
	Taxable Indicator	Char(1)		Taxable/non-taxable status indicator	No
	Entry/mode	Char(6)		Indicator that identifies whether the item was scanned or manually entered	No
	Department	Number(4)		Department of item sold or returned. Yes need to validate if using ReSA.	No
	Class	Number(4)		Class of item sold or returned. Yes need to validate if using ReSA.	No
	Subclass	Number(4)		Subclass of item sold or returned. Yes need to validate if using ReSA.	No
	Total Sales Quantity	Number(12)		Number of units sold at a particular location with 4 implied decimal places.	No
	Total Transaction Value	Number(20)		Sales value, net sales value of goods sold/returned with 4 implied decimal places.	No
	Override Reason	Char(6)		This column will be populated when an item's price has been overridden at the POS to define why it was overridden. This will also always be sent if the transaction originated in RCOM.	Yes, -1 for null
	Return Reason	Char(6)		The reason an item was returned.	Yes, -1 for null
	Total original sign	Char(1)	'P'- positive 'N' – negative		No

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Total Original Sales Value	Number(20)		This column will be populated when the item's price was overridden at the POS and the item's original unit retail is known. This will always be written when the transaction originated in RCOM. This has 4 implied decimals.	No
	Weather	Char(6)		For transaction types of 'COND', this field will store the type of weather for the store-day.	No
	Temperature	Char(6)		For transaction types of 'COND', this field will store the type of temperature for the store-day.	No
	Traffic	Char(6)		For transaction types of 'COND', this field will store the type of traffic for the store-day.	No
	Construction	Char(6)		For transaction types of 'COND', this field will store info regarding any construction on that store-day.	No
	Drop Shipment Indicator	Char(1)	'Y' or 'N'	Indicates whether item is involved in a drop shipment.	No
	Item Status	Char(6)		The status of the item, required for voided or exchanged items. Valid values are found in the code_detail table under code_type SASI.	Y, -1 for null

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Tran Process Sys	Char(3)		This column holds the name of the system that processed the transaction. This will be used for filtering duplicate transactions coming from the different systems for export to downstream systems. Expected values are POS – Point of Sale, OMS – Order Management System and SIM – Store Inventory Management.	Y, -1 for null
	Return Wh	Number(10)		This column contains the physical warehouse ID for the warehouse identifier where the item was returned.	N, -1 for null
	Fulfill Order No	Char(48)		This column holds the number from OMS related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS. This column is required if the order is a cross channel order (i.e. Sales Type = 'E') and the item status is 'ORD'.	N, -1 for null
	No Inventory Return Ind	Char(1)		This column contains an indicator that identifies a return without inventory. This is generally a non-required column, but in case of Returns, this is required.	N

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Sales Type	Char(1)		This column indicates whether the line item is a Regular Sale, a customer order serviced by OMS (External CO) or a customer order serviced by a store (In Store CO).	Y
	Return Disposition	Char(10)		This column will contain the disposition code published by RWMS as part of the Returns upload to OMS.	N, -1 for null
	Original Store	Char(10)		This column contains the store ID for the original store.	N
	Original Transaction Number	Number(10)		Original transaction number for the returned item.	N
	Reference Number 1	Char(30)			No
	Reference Number 2	Char(30)			No
	Reference Number 3	Char(30)			No
	Reference Number 4	Char(30)			No
	Reference Number 5	Char(30)			No
	Reference Number 6	Char(30)			No
	Reference Number 7	Char(30)			No
	Reference Number 8	Char(30)			No
	Reference Number 25	Char(30)			No
	Reference Number 26	Char(30)			No
	Reference Number 27	Char(30)			No
	Reference Number 28	Char(30)			No

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Reference Number 29	Char(30)			No
	Reference Number 30	Char(30)			No
	Reference Number 31	Char(30)			No
	Fulfill Loc Type	Char(2)		This column contains the fulfillment location type of the customer order. Valid values are 'S' for physical store and 'V' for virtual store.	N, -1 for null
	Fulfill Loc ID	Number(10)		This column contains the fulfillment location of the customer order. It can only be either a physical store or a virtual store.	N, -1 for null
	Posting Store	Number(10)		This column contains the store at which the item sale/return should be accounted for in case of cross-store sales happening at co-located stores. It is expected that this field will be populated only for items that are checked out at a different store from the one at which they are originally managed.	N, -1 for null
Transaction Detail	File Type Record Descriptor	Char(5)	TDETL	Identifies transaction record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	Discount Type	Char(6)		Code for discount type from code_detail, code_type = 'SADT'	No

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
Transaction	Promotional Transaction Type	Char(6)		Code for promotional type from code_detail, code_type = 'PRMT'	Yes
	Promotion Number	Number(10)	promotion number	Promotion number from Merchandising	No
	Promotion Component Number	Number(10)	Offer ID from Pricing		Required if it is a promotional sale.
	Coupon Number	Char(40)			Yes if Discount Type is 'SCoup'.
	Coupon Reference Number	Char(16)			No
	Sales Quantity	Number(12)		Number of units sold in this prom type with 4 implied decimal places.	No
	Transaction Sign	Char(1)	'P' - positive 'N' - negative		Yes
	Transaction Value	Number(20)		Value of units sold in this promotion type with 4 implied decimal places.	Yes
	Discount Value	Number(20)		Value of discount given in this prom type with 4 implied decimal places.	Yes
	Reference Number 13	Char(30)			No
	Reference Number 14	Char(30)			No
	Reference Number 15	Char(30)			No
	Reference Number 16	Char(30)			No
	Transaction Trailer	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	Transaction Count	Number(6)	specified by external system	Number of TDETL records in this transaction set	Yes

Table 6-23 (Cont.) RDWT File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Record Counter	Number(10)		Number of records/ transactions processed in current file (only records between head & tail)	Yes

Transaction Item Information Produced by saexpdw.pc after Translation by resa2dw

Table 6-24 File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Business date	Number(8)		Format YYYYMMDD	Yes
	Transaction Date	Number(14)	transaction date	Date sale/return transaction was processed at the POS. Format YYYYMMDDHH24M ISS	Yes
	Location	Number(10)	specified by external system	Store or warehouse identifier. This value is now being determined based on either the Account for Sale or Account for Return system option.	Yes
	Register ID	Char(5)		The register identifier	Yes, -1 for null
	Banner ID	Char(4)		The unique identifier of the banner.	Yes, -1 for null
	Line Media ID	Char(10)		The identifier of the order line media. For non-merchandise items, such as Shipping & Handling, Service Lines and gift certificates, the media code will be that of the order line it's associated.	Yes, -1 for null

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Selling Item ID	Char(25)		The unique identifier of a selling item.	Yes, -1 for null
	Customer Order Header ID	Char(48)		The unique identifier of a customer order.	Yes, -1 for null
	Customer Order Line ID	Char(30)		The identifier of a customer order line. For a Value Added Service, like monogramming, this will be the line number for the item, which the service was applied.	Yes, -1 for null
	Customer Order Create Date	Number(8)		The customer order creation date	Yes, 'transaction date' for null
	Cashier Identifier	Char(10)		The cashier number. This will be the unique employee number.	Yes, -1 for null
	Salesperson Identifier	Char(10)		The salesperson number. This will be the unique employee number.	Yes, -1 for null
	Customer ID Type	Char(6)		The type of ID number used by this customer.	Yes, -1 for null
	Customer ID Number	Char(16)		Customer id associated with the transaction.	Yes, -1 for null
	Transaction Number	Number(10)		The unique transaction reference number generated by the POS.	Yes
	Original Register ID	Char(5)		Register ID of the original transaction.	Yes for a transaction type of 'PVOID'.
	Original Transaction Number	Number(10)		Transaction number of the original transaction.	Yes for a transaction type of 'PVOID'.
	Transaction Header Number	Numeric(20)		Unique reference used within sales audit to represent the date/store/register/tran_no	Yes

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Revision number	Number(3)		Number used to identify the version of the transaction being sent.	Yes
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Transaction Type	Char(6)		Transaction type code	Yes
	Sub Transaction Type	Char(6)		The Sub Transaction type	Yes, -1 for null
	Retail Type	Char(1)	'R'egular, 'P'romo, or 'C'learance		Yes
	Item_Seq_No	Number(4)		The order in which items were entered during the transaction.	No
	Employee Number (Cashier)	Char(10)		Employee identification number. This will only be populated if the sub transaction type is 'EMP'.	Yes, -1 for null
	Receipt Indicator	Char(1)		Flag that identifies returns that have been processed without a receipt. This field will only be populated if the transaction type is 'RETURN'.	No
	Reason Code	Char(6)		A reason is required with a Paid In/Out transaction type, and optional with a return transaction.	Yes, -1 for null
	Vendor number	Numeric(10)		This will only get populated when the paid in code is Expense Vendor	No
	Item Type	Char(6)	item type identifier	Type of item sold, 'ITEM', 'REF', 'GCN' (gift certificate number), or 'NMITEM'	No

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Item	Char(25)		ID number of the item or gift certificate.	No. Required if Item Type is not null.
	Ref Item	Char(25)		Sub-transaction level item	No. Also, this field can never be populated without a transaction level item in the item field.
	Taxable Indicator	Char(1)		Taxable/non-taxable status indicator	No
	Entry/mode	Char(6)		Indicator that identifies whether the item was scanned or manually entered	No
	Department	Number(4)		Department of item sold or returned. Yes need to validate if using ReSA.	No
	Class	Number(4)		Class of item sold or returned. Yes need to validate if using ReSA.	No
	Subclass	Number(4)		Subclass of item sold or returned. Yes need to validate if using ReSA.	No
	Total Sales Quantity	Number(12)		Number of units sold at a particular location with 4 implied decimal places.	No
	Total Transaction Value	Number(20)		Sales value, net sales value of goods sold/returned with 4 implied decimal places.	No
	Override Reason	Char(6)		This column will be populated when an item price has been overridden at the POS to define why it was overridden. This will always be sent if the transaction originated in RCOM.	Yes, -1 for null
	Return Reason	Char(6)		The reason an item was returned.	Yes, -1 for null

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Total original sign	Char(1)	'P'- positive 'N' – negative		No
	Total Original Sales Value	Number(20)		This column will be populated when the item's price was overridden at the POS and the item's original unit retail is known. This will always be sent if the transaction originated in RCOM. This has 4 implied decimals.	No
	Weather	Char(6)		For transaction types of 'COND', this field will store the type of weather for the store-day.	No
	Temperature	Char(6)		For transaction types of 'COND', this field will store the type of temperature for the store-day.	No
	Traffic	Char(6)		For transaction types of 'COND', this field will store the type of traffic for the store-day.	No
	Construction	Char(6)		For transaction types of 'COND', this field will store info regarding any construction on that store-day.	No
	Drop Shipment Indicator	Char(1)	'Y' or 'N'	Indicates whether item is involved in a drop shipment.	No
	Item Status	Char(6)		The status of the item, required for voided or exchanged items. Valid values are found in the code_detail table under code_type SASI.	Y, -1 for null

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Tran Process Sys	Char(3)		This column holds the name of the system that processed the transaction. This will be used for filtering duplicate transactions coming from the different systems for export to downstream systems. Expected values are POS – Point of Sale, OMS – Order Management System and SIM – Store Inventory Management.	Y, -1 for null
	Return Wh	Number(10)		This column contains the physical warehouse ID for the warehouse identifier where the item was returned.	N, -1 for null
	Fulfill Order No	Char(48)		This column holds the number from OMS related to the fulfillment details. One or more fulfillment orders could relate back to a single customer order in OMS. This column is required if the order is a cross channel order (i.e. Sales Type = 'E') and the item status is 'ORD'.	N, -1 for null
	No Inventory Return Ind	Char(1)		This column contains an indicator that identifies a return without inventory. This is generally a non-required column, but in case of Returns, this is required.	N

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Sales Type	Char(1)		This column indicates whether the line item is a Regular Sale, a customer order serviced by OMS (External CO) or a customer order serviced by a store (In Store CO).	Y
	Return Disposition	Char(10)		This column will contain the disposition code published by RWMS as part of the Returns upload to OMS.	N, -1 for null
	Original Store	Char(10)		This column contains the store ID for the original store.	No
	Original Transaction Number	Number(10)		Original transaction number for the returned item.	No
	Discount Type	Char(6)		Code for discount type from code_detail, code_type = 'SADT'	No
	Promotional Transaction Type	Char(6)		Code for promotional type from code_detail, code_type = 'PRMT'	Yes
	Promotion Number	Number(10)	promotion number	Promotion number from Merchandising	No
	Promotion Component Number	Numbr(10)	Offer ID from Pricing		Required if it is a promotional sale.
	Coupon Number	Char(40)			Yes if Discount Type is 'SCOUP'.
	Coupon Reference Number	Char(16)			No
	Sales Quantity	Number(12)		Number of units sold in this prom type with 4 implied decimal places.	No
	Transaction Sign	Char(1)	'P'- positive 'N' – negative		Yes

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Transaction Value	Number(20)		Value of units sold in this promotion type with 4 implied decimal places.	Yes
	Discount Value	Number(20)		Value of discount given in this prom type with 4 implied decimal places.	Yes
	Reference Number 1	Char(30)			No
	Reference Number 2	Char(30)			No
	Reference Number 3	Char(30)			No
	Reference Number 4	Char(30)			No
	Reference Number 5	Char(30)			No
	Reference Number 6	Char(30)			No
	Reference Number 7	Char(30)			No
	Reference Number 8	Char(30)			No
	Reference Number 13	Char(30)			No
	Reference Number 14	Char(30)			No
	Reference Number 15	Char(30)			No
	Reference Number 16	Char(30)			No
	Reference Number 25	Char(30)			No
	Reference Number 26	Char(30)			No
	Reference Number 27	Char(30)			No
	Reference Number 28	Char(30)			No
	Reference Number 29	Char(30)			No
	Reference Number 30	Char(30)			No
	Reference Number 31	Char(30)			No

Table 6-24 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description	Required
	Fulfill Loc Type	Char(2)		This column contains the fulfillment location type of the customer order. Valid values are 'S' for physical store and 'V' for virtual store.	N, -1 for null
	Fulfill Loc ID	Number(10)		This column contains the fulfillment location of the customer order. It can only be either a physical store or a virtual store.	N, -1 for null
	Posting Store	Number(10)		This column contains the store at which the item sale/return should be accounted for in case of cross-store sales happening at co-located stores. It is expected that this field will be populated only for items that are checked out at a different store from the one at which they are originally managed.	N, -1 for null

RDWF File

Table 6-25 RDWF File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Type Definition	Char(4)	RDWF	Identifies file as 'Retail Analytics Form of Payment (Tender) file'	Yes

Table 6-25 (Cont.) RDWF File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Detail	File Create Date	Number(14)	create date	Date file was written by external system. Format YYYYMMDDHH24MISS	Yes
	File Type Record Descriptor	Char(5)	FDETL	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	Business date	Number(8)		Format YYYYMMDD	Yes
	Transaction Date	Number(14)	transaction date	Date sale/return transaction was processed at the POS. Format YYYYMMDDHH24MISS	Yes
	Location	Number(10)	specified by external system	Store or warehouse identifier.	Yes
	Cashier Identifier	Char(10)		The cashier number. This will be the unique employee number.	Yes, -1 for null
	Register Identifier	Char(5)			Yes, -1 for null
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Transaction Sequence Number	Numeric(20)		Unique reference used within sales audit to represent the date/store/register/transaction number	Yes
	Revision number	Number(3)		Number used to identify the version of the transaction being sent.	Yes
	Transaction Type	Char(6)		Transaction type code.	Yes

Table 6-25 (Cont.) RDWF File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Tender type group	Char(6)			Yes
	Tender type id	Number(6)		Tender type code.	Yes
	Tender amount	Number(20)		Tender amount.	Yes
	Credit Card Entry Mode	Char(6)		Contains the method in which the transaction was entered at the POS. Possible entry modes could include: Terminal Used, Magnetic Strip Track One Read, Magnetic Strip Two Read, Magnetic Strip One Transmitted, or Magnetic Strip Two Transmitted. The code type for this field is 'CCEM'.	No
	Voucher Number	Char(25)			No
	Voucher Age	Numeric(5)		Age of the gift certificate. Redeemed date minus sold date.	Yes if Tender Type Group is 'VOUCH'.
	Escheat Date	Numeric(8)		Date on which this gift certificate escheats. Format is YYYYMMDD.	Yes if voucher can escheat.
	Coupon Number	Char(40)			Yes if Tender Type Group is 'COUPON'.
	Coupon Reference Number	Char(16)			No. Only if Tender Type Group is 'COUPON'.
	Transaction Status	Char(1)		Determines if the transaction is Present ('P') or Voided/Deleted ('R' - Reverse)	NO
	Reference Number 9	Char(30)			No
	Reference Number 10	Char(30)			No
	Reference Number 11	Char(30)			No

Table 6-25 (Cont.) RDWF File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Trailer	Reference Number 12	Char(30)			No
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Record Counter	Number(10)		Number of records/ transactions processed in current file (only records between head & tail)	Yes

Retail Analytics Form of Payment File after Translation by resa2dw

Table 6-26 Form of Payment File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Business date	Number(8)		Format YYYYMMDD	Yes
	Transaction Date	Number(14)	transaction date	Date sale/return transaction was processed at the POS. Format YYYYMMDDHH24MISS	Yes
	Location	Number(10)	specified by external system	Store or warehouse identifier.	Yes
	Cashier Identifier	Char(10)		The cashier number. This will be the unique employee number.	Yes, -1 for null
	Register Identifier	Char(5)			Yes, -1 for null
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes

Table 6-26 (Cont.) Form of Payment File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Transaction Sequence Number	Numeric(20)		Unique reference used within sales audit to represent the date/store/register/transaction number	Yes
	Revision number	Number(3)		Number used to identify the version of the transaction being sent.	Yes
	Transaction Type	Char(6)		Transaction type code.	Yes
	Tender type group	Char(6)			Yes
	Tender type id	Number(6)		Tender type code.	Yes
	Tender amount	Number(20)		Tender amount.	Yes
	Credit Card Entry Mode	Char(6)		Contains the method in which the transaction was entered at the POS. Possible entry modes could include: Terminal Used, Magnetic Strip Track One Read, Magnetic Strip Two Read, Magnetic Strip One Transmitted, or Magnetic Strip Two Transmitted. The code type for this field is 'CCEM'.	No
	Voucher Number	Char(25)			No
	Voucher Age	Number(5)		Age of the gift certificate. Redeemed date minus sold date.	Yes if Tender Type Group is 'VOUCH'.
	Escheat Date	Number(8)		Date on which this gift certificate escheats. Format is YYYYMMDD.	Yes if voucher can escheat.
	Coupon Number	Char(40)			Yes if Tender Type Group is 'COUPON'.

Table 6-26 (Cont.) Form of Payment File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Coupon Reference Number	Char(16)			No. Only if Tender Type Group is 'COUPON'.
	Transaction Status	Char(1)		Determines if the transaction is Present ('P') or Voided/Deleted ('R' - Reverse)	No
	Reference Number 9	Char(30)			No
	Reference Number 10	Char(30)			No
	Reference Number 11	Char(30)			No
	Reference Number 12	Char(30)			No

RDWS File

Table 6-27 RDWS File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Type Definition	Char(4)	RDWS	Identifies file as 'Retail Analytics Store Totals file'	Yes
	File Create Date	Numeric(14)	create date	Date file was written by external system. Format YYYYMMDDHH24MISS	Yes
File Detail	File Type Record Descriptor	Char(5)	FDETL	Identifies transaction record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	Business date	Number(8)		Format YYYYMMDD	Yes

Table 6-27 (Cont.) RDWS File

Record Name	Field Name	Field Type	Default Value	Description	Required
	Location	Number(10)	specified by external system	Store or warehouse identifier	Yes
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Total ID	Char(10)		Category identifier used to determine the type of total.	Yes
	Reference Number 1	Char(30)			No
	Reference Number 2	Char(30)			No
	Reference Number 3	Char(30)			No
	Total Sign	Char(1)	'P'- positive 'N' – negative		Yes
	Total Amount	Number(20)		Total over/short amount with 4 implied decimal places.	Yes
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Record Counter	Number(10)		Number of records/ transactions processed in current file (only records between head & tail)	Yes

Store Totals Information after Translation by resa2dw

Table 6-28 Store Totals Information

Record Name	Field Name	Field Type	Default Value	Description	Required
	Business date	Number(8)		Format YYYYMMDD	Yes

Table 6-28 (Cont.) Store Totals Information

Record Name	Field Name	Field Type	Default Value	Description	Required
	Location	Number(10)	specified by external system	Store or warehouse identifier	Yes
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Total ID	Char(10)		Category identifier used to determine the type of total.	Yes
	Reference Number 1	Char(30)			No
	Reference Number 2	Char(30)			No
	Reference Number 3	Char(30)			No
	Total Sign	Char(1)	'P'- positive 'N' – negative		Yes
	Total Amount	Number(20)		Total over/short amount with 4 implied decimal places.	Yes

RDWC File

Table 6-29 RDWC File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Type Definition	Char(4)	RDWC	Identifies file as 'Retail Analytics Cashier/Register Totals file'	Yes
	File Create Date	Number(14)	create date	Date file was written by external system. Format YYYYMMDDHH24MISS	Yes

Table 6-29 (Cont.) RDWC File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Detail	File Type Record Descriptor	Char(5)	FDETL	Identifies transaction record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	Business date	Number(8)		Format YYYYMMDD	Yes
	Location	Number(10)	specified by external system	Store or warehouse identifier	Yes
	Cashier Identifier	Char(10)		The cashier number	If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL. Yes, -1 for null
	Register ID	Char(5)		The register identifier	If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL. Yes, -1 for null
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Total ID	Char(10)		Category identifier used to determine the type of total.	Yes
	Reference Number 1	Char(30)			No
	Reference Number 2	Char(30)			No
	Reference Number 3	Char(30)			No
	Total Sign	Char(1)	'P'- positive 'N' – negative		Yes

Table 6-29 (Cont.) RDWC File

Record Name	Field Name	Field Type	Default Value	Description	Required
File Trailer	Total Amount	Number(20)		Total over/short amount with 4 implied decimal places.	Yes
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type	
	File Line Identifier	Number(10)	specified by external system	ID of current line being processed by input file.	Yes
	File Record Counter	Number(10)		Number of records/ transactions processed in current file (only records between head & tail)	Yes

Cashier/ Register Totals Information after Translation by resa2dw

Table 6-30 Cashier/Register Totals Information

Record Name	Field Name	Field Type	Default Value	Description	Required
	Business date	Number(8)		Format YYYYMMDD	Yes
	Location	Number(10)	specified by external system	Store or warehouse identifier	Yes
	Cashier Identifier	Char(10)		The cashier number	If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL. Yes, -1 for null
	Register ID	Char(5)		The register identifier	If Cashier_id is NULL then Register_id has value. If Cashier_id has value then Register_id is NULL. Yes, -1 for null

Table 6-30 (Cont.) Cashier/Register Totals Information

Record Name	Field Name	Field Type	Default Value	Description	Required
	Sales Sign	Char(1)	'P'- positive 'N' – negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative.	Yes
	Total ID	Char(10)		Category identifier used to determine the type of total.	Yes
	Reference Number 1	Char(30)			No
	Reference Number 2	Char(30)			No
	Reference Number 3	Char(30)			No
	Total Sign	Char(1)	'P'- positive 'N' – negative		Yes
	Total Amount	Number(20)		Total over/short amount with 4 implied decimal places.	Yes

Design Assumptions

N/A

Export Inventory Reservation/Release for In Store Customer Order & Layaway Transactions (saordinvexp)

Module Name	saordinvexp.pc
Description	Export Inventory Reservation/Release for In Store Customer Order & Layaway Transactions from Sales Audit
Functional Area	Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA12
Wrapper Script	rmswrap_multi_dnld_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program will generate a flat file to reserve or un-reserve the inventory for items on in-store customer order or layaway transactions. Inventory will be reserved for items on customer order/layaway initiate and un-reserved for customer order/layaway cancel or complete transactions.

Customer orders can be categorized into two categories: In-Store Customer Orders and External Customer Orders. The In-Store Customer Orders are defined as orders that are serviced at the store and inventory reservation is done in Oracle Retail Store Inventory Management (SIM). While the External Customer orders are serviced by an external order management system, no inventory reservation will be made at the store in SIM.

This batch should only process records where the sales type is not equal to External Customer Sales, as it handles only the in-store type orders.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination.

Records are fetched, updated, and inserted in batches of `pl_commit_max_ctr`. Only two commits are done, one to establish the store/day lock and another at the end, to release the lock after a store/day is completely processed. The ORIN formatted output file is created with a temporary name and renamed just before the end of store/day commit.

In case of failure, all work done is rolled back to the point right after the call to `get_lock()` and the lock is released. Thus, the rollback segment should be large enough to hold all inserts into `sa_exported` for one store/day.

I/O Specification

Integration Type	Inventory Export from Sales Audit to Merchandising
File Name	ORIN_<store>_<tran_date>_<sysdate>
Integration Contract	IntCon000049

Output File Layout

Table 6-31 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type.
	File Line Id	Char(10)	0000000001	Sequential file line number.
	File type Definition	Char(4)	ORIN	Identifies the file type.
	File Create Date	Char(14)	N/A	File Create Date in YYYYMMDDHHMMSS format.
	Location	Number(10)	N/A	Store location number.
THEAD	Record descriptor	Char(5)	THEAD	Identifies the file record type.
	File Line Id	Char(10)		Sequential file line number.

Table 6-31 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
TDETL	Transaction Date & Time	Char(14)	Transaction Date	Date and time of the order processed.
	Transaction Type	Char(6)	SALE	Transaction type code specifies whether the transaction is sale or return.
	Record descriptor	Char(5)	TDETL	Identifies the file record type.
	File Line Id	Char(10)	N/A	Sequential file line number.
	Item Type	Char(3)	REF or ITM	Can be REF or ITM.
	Item	Char(25)	N/A	ID number of the ITM or REF.
	Item Status	Char(6)	LIN - Layaway Initiate LCA - Layaway Cancel LCO - Layaway Complete PVLCO - Post void of Layaway complete ORI - Pickup/delivery Initiate ORC - Pickup/delivery Cancel ORD - Pickup/delivery Complete PVORD - Post void of Pick-up/delivery complete	Type of transaction.
	Dept	Number(4)	N/A	Department of item sold or returned.
	Class	Number(4)	N/A	Class of item sold or returned
	Sub class	Number(4)	N/A	Subclass of item sold or returned.
	Pack Ind	Char(1)	N/A	Pack indicator of item sold or returned.
	Quantity Sign	Char(1)	P or N	Sign of the quantity.
	Quantity	Number(12)	N/A	Quantity * 10000 (4 implied decimal places), number of units for the given order (item) status.
	Selling UOM	Char(4)	N/A	UOM at which this item was sold.
	Catchweight Ind	Char(1)	N/A	Indicates if the item is a catchweight item. Valid values are Y or NULL.
	Customer Order number	Char(48)	N/A	Customer Order number.

Table 6-31 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Posting Store	Number(10)		Contains the store at which the item reservation/ reservation cancellation should occur in case of cross-store transactions happening at co-located stores. It is expected that this field will be populated only for items that are to be reserved (or have the reservation canceled) at a different store from the one at which the checkout happened.
TTAIL	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type.
	File Line Identifier	Number(10)	Specified by Sales Audit	ID of current line being processed by input file.
	Transaction count	Number(6)	Specified by Sales Audit	Number of TDETL records in this transaction set.
FTAIL	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type.
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.
	File Record Counter	Number(10)		Number of records/transactions processed in the current file (only records between FHEAD and FTAIL).

Design Assumptions

N/A

Export of POS transactions from Sales Audit to Merchandising (saexprms)

Module Name	saexprms.pc
Description	Export of POS transactions from Sales Audit to Merchandising
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA01
Wrapper Script	rmswrap_multi_dnld_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to fetch all sale and return transactions that do not have Merchandising errors from the Sales Audit database tables for transmission to the Merchandising system. Transaction data is rolled up to the item/store/day/price point/sales type level for SALES transaction type and item/store/day/price point/sales type/no inventory return indicator/return disposition/return warehouse level for RETURN transaction types.

If unit of work system parameter is defined as 'S', then the whole store/day is skipped if any Merchandising error is found. If this value is 'T', then only transactions with Merchandising errors are skipped.

If the consignment rate review process is used, then consignment transactions eligible for review will not be exported from Sales Audit to Merchandising until the transaction has either been marked as reviewed or the review period specified has elapsed. Once the review period has elapsed, the batch will mark the transactions still pending review as expired and export them to Merchandising.

If the transaction has a status of Deleted and it has previously been transmitted, a reversal of the transaction will be sent.

A file is generated for each store/day.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, updated and inserted in batches of pl_commit_max_ctr. Only two commits will be done, one to establish the store/day lock and another at the end, to release the lock after a store/day has been completely processed. The POSU formatted output file will be created with a temporary name and renamed just before the end of store/day commit.

In case of failure, all work done will be rolled back to the point right after the call to get_lock() and releases the lock. Thus, the rollback segment should be large enough to hold all inserts into sa_exported for one store/day.

I/O Specification

Integration Type	Download from ReSA
File Name	"POSU_" appended with store number, business date and system date
Integration Contract	IntCon000044

Table 6-32 File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type
	File Line Id	Char(10)	0000000001	Sequential file line number
	File type definition	Char(4)	POSU	Identifies the file type
	File Create Date	Char(14)		File Create Date in YYYYMMDDHHMMSS format
	Store	Number(10)		Store location

Table 6-32 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
THEAD	Vat include indicator	Char(1)		Determines whether or not the store stores values including vat. Not required but populated by Oracle Retail sales audit
	Vat region	Number(4)		Vat region the given location is in. Not required but populated by Oracle Retail sales audit
	Currency code	Char(3)		Currency of the given location. Not required but populated by Oracle Retail sales audit
	Currency retail decimals	Number(1)		Number of decimals supported by given currency for retails. Not required but populated by Oracle Retail sales audit
	Record descriptor	Char(5)	THEAD	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Transaction date	Char(14)		Transaction date in YYYYMMDDHHMMSS format. Corresponds to the date that the sale/return transaction was processed at the POS
	Item Type	Char(3)	REF or ITM	Can be REF or ITM
	Item	Char(25)		Id number of the ITM or REF
	Dept	Number(4)		Department of item sold or returned.
	Class	Number(4)		Class of item sold or returned.
	Sub Class	Number(4)		Subclass of item sold or returned.
	Pack Ind	Char(1)		Pack indicator of item sold or returned.
	Item Level	Number(1)		Item level of item sold or returned.
	Tran level	Number(1)		Transaction level of item sold or returned.
	Wastage Type	Char(6)		Wastage type of item sold or returned
	Wastage pct	Number(12)		Waste pct (4 implied decimal places)
	Tran type	Char(1)		Transaction type code to specify whether transaction is a sale or a return
	Drop Shipment indicator	Char(1)		Indicates whether the transaction is a drop shipment or not.
	Total sales qty	Number(12)		Total sales quantity (4 implied decimal places)
	Selling UOM	Char(4)		Selling Unit of Measure for the item

Table 6-32 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Sales sign	Char(1)		Determines if the Total Sales Quantity and Total Sales Value are positive or negative.
	Total Sales Value	Number(20)		Total sales value of goods sold/ returned (4 implied decimal places)
	Last Date time modified	Char(14)		Date and time of last modification in YYYYMMDDHHMMSS format. For VBO future use
	Catchweight indicator	Char(1)		Indicates if item is a catchweight item.
	Total weight	Number(12)		The actual weight of the item, only populated if catchweight_ind = 'Y'
	Sub Tran type indicator	Char(1)		Tran type for ReSA Valid values are 'A', 'D', NULL
	Total IG TAX Value	Number(20)		This will indicate total of all IG TAX amount for the item.
	Sales Type	Char(1)		This column indicates whether the line item is a Regular Sale, a customer order serviced by OMS (External CO) or a customer order serviced by a store (In Store CO).
	No Inventory Return Indicator	Char(1)		This column contains an indicator that identifies a return without inventory. This is generally a non-required column, but in case of Returns, this is required.
	Return Disposition	Char(10)		This column will contain the disposition code published by RWMS as part of the Returns upload to OMS.
	Return Warehouse	Char(10)		This column contains the physical warehouse ID for the warehouse identifier where the item was returned.
	Customer Order No	Char(48)		This column contains the customer order number ID.
	Fulfillment Order No	Char(48)		This column contains the fulfillment order number ID.
	Fulfillment Loc Type	Char(2)		This column contains the fulfillment location type. Code for the fulfillment loc type from code_detail where code_type = 'FLTP'
	Fulfillment Loc	Number(10)		This column contains the fulfillment loc ID.
	Orig Store	Number(10)		This column contains the original store value for a Return transaction.

Table 6-32 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	POS Tran Id	Number(20)		This column contains the unique identifier for a sale transaction. This is an Optional field.
	Posting Store	Number(10)		This column contains the store at which the item sale/return should be accounted for in case of cross-store sales happening at co-located stores. It is expected that this field will be populated only for items that are checked out at a different store from the one at which they are originally managed.
	Consignment Rate	Number(12)		This column contains the consignment rate that should be applied while posting the sales/returns to Merchandising. (4 implied decimal places)
	Consignment Unit Cost	Number(20)		This column contains the consignment unit cost that should be applied while posting the sales/returns to Merchandising. (4 implied decimal places)
	Inventory Identifier Type	Char(6)		This column contains the inventory identifier type passed in Sales/Return transactions. Valid values are found under the Inventory Identifier Types (IIDT) code type, for e.g Lot (L), Expiry Date (E), Import Document (D).
	Inventory Id	Char(120)		This column contains the inventory id value being passed in sales/return transactions.
TTAX	Record descriptor	Char(5)	TTAX	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Tax Code	Char(6)		The Tax Code of the item
	Tax Rate	Number(20)		The tax rate of the item (10 implied decimal places)
	Total Tax Amount	Number(20)		The item level tax or prorated transaction level tax of the item (4 implied decimal places)
TDETL	Record descriptor	Char(5)	TDETL	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Promo Tran Type	Char(6)		Code for the promotional type from code_detail where code_type = 'PRMT'
	Promotion Number	Number(10)		Promotion number from Merchandising

Table 6-32 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Sales quantity	Number(12)		Sales quantity sold for this promotion type (4 implied decimal places)
	Sales value	Number(20)		Sales value for this promotion type (4 implied decimal places)
	Discount value	Number(20)		Discount value for this promotion type (4 implied decimal places)
	Promotion component	Number(10)		Links the promotion to additional pricing attributes. This contains the OFFER_ID from Pricing.
TTAIL	Record descriptor	Char(5)	TTAIL	Identifies the file record type
	File Line Id	Char(10)		Sequential file line number
	Tran Record Counter	Number(6)		Number of TDETL records in this transaction set
FTAIL	Record descriptor	Char(5)	FTAIL	Identifies the file record type
	File Line Id	Number(10)		Sequential file line number
	File Record counter	Number(10)		Number of records/transactions processed in current file (only records between head & tail)

Fields expected in POSU format based on changes adopted:

	V16	V16 with Customer Order Changes	V19	V19.3	V24	V25
Fulfillment Order No	No	Yes	Yes	Yes	Yes	Yes
Fulfillment Loc Type	No	Yes	Yes	Yes	Yes	Yes
Fulfillment Loc	No	Yes	Yes	Yes	Yes	Yes
Orig Store	No	Yes	Yes	Yes	Yes	Yes
POS Tran Id	No	No	Yes	Yes	Yes	Yes
Posting Store	No	No	No	Yes	Yes	Yes
Consignment Rate/ Unit Cost	No	No	No	No	Yes	Yes
Inventory Identifier Type	No	No	No	No	No	Yes
Inventory Id	No	No	No	No	No	Yes

Design Assumptions

1. Tax can be sent either in TTAX or IGTAX regardless of default `tax_type` of SVAT, GTAX, SALES or GTS. Prorated tax in TTAX will only be sent to Merchandising in all configuration.
2. POS can send either transactional level tax details in TTAX lines or item-level tax details in IGTAX lines through the RTLOG file to ReSA. These tax details will be passed on to Merchandising in the TTAX lines of the POSU file. Even though POS can pass multiple

IGTAX/TTAX lines to ReSA and from Sales Audit to Merchandising, Merchandising only supports one tax code per item. If multiple taxes for an item are sent from POS to Sales Audit, they will be summed to a single tax in Merchandising sales upload process and assigned one of the applicable tax codes when writing tran_data 88.

3.

Export of POS Transactions from Sales Audit to Merchandising Based on Direct Table Load (saexprocsales)

Module Name	saexprocsales.ksh
Description	Export of POS transactions from Sales Audit to Merchandising based on direct table load
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ksh
Catalog ID	RSA01
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch module fetches all sale and return transactions that do not have Merchandising errors from the Sales Audit database tables for transmission to the Merchandising system. Transaction data is rolled up to the store/day/transaction/item/price point/sales type level for the SALES transaction type and store/day/transaction/item/price point/sales type/no inventory return indicator/return disposition/return warehouse level for the RETURN transaction type.

If the unit of work system parameter is defined as S, then the whole store/day is skipped if any Merchandising error is found. If this value is T, then only transactions with Merchandising errors are skipped.

If the consignment rate review process is used, then consignment transactions eligible for review will not be exported from Sales Audit to Merchandising until the transaction has either been marked as reviewed, or the specified review period has elapsed. Once the review period has elapsed, the batch will mark the transactions still pending review as expired and export them to Merchandising.

If the transaction has a status of Deleted and it has previously been transmitted, a reversal of the transaction will be sent.

Transactions are exported to the Merchandising Sales process global temporary tables and the sales postings are processed.

The execution of this batch is controlled through a Sales Audit System option that indicates whether a table-based or file-based export is to be used.

Design Assumptions

For exports from Sales Audit to Merchandising, either saexprocsales or saexprms can be used to export Sales Audit transactions, depending on the configuration of Sales Audit.

For Merchandising, importing sales transactions is supported through both the saexprcsales table-based integration or POSU-file-based import modes.

POS can send either transaction-level tax details in TTAX lines or item-level tax details in IGTAX lines through the RTLOG file to Sales Audit. These tax details are passed on to Merchandising in relevant sales processing, global, temporary table records.

For a non-GTS environment, even though POS can pass multiple IGTAX/TTAX lines to Sales Audit and from Sales Audit to Merchandising, Merchandising only supports one tax code per item. If multiple taxes for an item are sent from POS to Sales Audit, they will be summed to a single tax in the Merchandising sales upload process and assigned one of the applicable tax codes when writing tran_data 88.

When GTS is set as the Default Tax type in the system options, the sales process batch should be capable of both processing multiple records in sales processing global temporary tables, and posting multiple lines for transaction code 88, with one line for each tax detail record.

When GTS is set to the Default Tax type in the system options, the sales process batch can process multiple TTAX lines in the POSU file and post multiple lines for transaction code 88, one for each TTAX line.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. The rollback segment should be large enough to hold all inserts into sa_exported for one store/day.

The number of threads running in parallel is based on the RMS_PLSQL_BATCH_CONFIG.MAX_CONCURRENT_THREADS column value with the SA_EXPRC_SALES_SQL program name.

Key Tables Affected

Table	Select	Insert	Update	Delete
SA_STORE_DAY	Yes	No	No	No
SA_EXPORT_LOG	Yes	No	Yes	No
MV_SA_RESTART_EXPORT_RMS	Yes	No	No	No
STORE	Yes	No	No	No
CURRENCIES	Yes	No	No	No
SA_TRAN_HEAD	Yes	No	No	No
SA_ERROR	Yes	No	No	No
SA_ERROR_IMPACT	Yes	No	No	No
SA_EXPORTED	Yes	Yes	No	No
SA_TRAN_SEQ_TEMP	No	Yes	No	Yes
SA_TRAN_HEAD_REV	Yes	No	No	No
SA_EXPORTED_REV	Yes	No	No	No
SA_SYSTEM_OPTIONS	Yes	No	No	No

Table	Select	Insert	Update	Delete
SA_TRAN_ITEM_REV	Yes	No	No	No
ITEM_MASTER	Yes	No	No	No
SA_TRAN_DISC_REV	Yes	No	No	No
SA_TRAN_DISC	Yes	No	No	No
SA_TRAN_ITEM	Yes	No	No	No
SA_TRAN_SEQ_T EMP	Yes	Yes	No	Yes
SA_STORE_DAY_READ_LOCK	No	Yes	No	Yes
SA_TRAN_IGTAX_REV	Yes	No	No	No
SA_TRAN_IGTAX	Yes	No	No	No
SA_TRAN_TAX	Yes	No	No	No
SA_TRAN_TAX_R EV	Yes	No	No	No
VAT_ITEM	Yes	No	No	No
VAT_HISTORY	No	Yes	Yes	No
DAILY_SALES_DISCOUNT	No	Yes	Yes	No
JOB_AUDIT_LOG	No	Yes	No	No
SA_EXPROC_SALES_ERROR	No	Yes	No	No
SVC_TRAN_HEAD_GTT	No	Yes	No	No
SVC_POSUPLD_F HEAD_GTT	No	Yes	No	No
SVC_POSUPLD_T HEAD_GTT	No	Yes	Yes	Yes
SVC_POSUPLD_T TAX_GTT	No	Yes	No	No
SVC_POSUPLD_T DETL_GTT	No	Yes	No	No
DEAL_HEAD	Yes	No	No	No
DEAL_COMP_PROM	Yes	No	No	No
DEAL_ACTUALS_FORECAST	Yes	No	No	No
ITEM_LOC	Yes	No	No	No
ITEM_LOC_SOH	Yes	No	Yes	No
VAT_ITEM	Yes	No	No	No
ITEM_SUPP_COUNTRY	Yes	No	No	No
ITEM_SUPPLIER	Yes	No	No	No

Table	Select	Insert	Update	Delete
SUPS	Yes	No	No	No
TERMS	Yes	No	No	No
PRICE_HIST	Yes	No	No	No
TEMP_TRAN_DATA	No	Yes	No	No
ITEM_LOC_HIST	Yes	Yes	Yes	No
ITEM_LOC_HIST_MTH	Yes	Yes	Yes	No
EDI_DAILY_SALES	Yes	Yes	Yes	No
ORDHEAD	Yes	Yes	No	No
INVC_HEAD	Yes	Yes	No	No
INVC_MERCH_VANT	Yes	Yes	Yes	No
INVC_XREF	No	Yes	No	No
INVC_DETAIL_TEMP2	No	Yes	No	No
INVC_DETAIL	Yes	No	No	No
CODE_DETAIL	Yes	No	No	No
UOM_CLASS	Yes	Yes	No	No
ITEM_XFORM_HEADER	Yes	No	No	No
ITEM_XFORM_DETAIL	Yes	No	No	No
ITEM_SUPP_COUNTRY_LOC	Yes	No	No	No
TRAN_DATA	No	Yes	No	No
INVC_DETAIL_TEMP	No	Yes	No	No
INVC_HEAD_TEMP	No	Yes	No	No
CONCESSION_DATA	No	Yes	No	No
DEAL_ACTUALS_ITEM_LOC	Yes	Yes	Yes	No
V_PACKSKU_QTY	Yes	No	No	No
RTV_HEAD	Yes	No	No	No

Export of Revised Sale/Return Transactions from ReSA to SIM/SIOCS (saexpsim)

Module Name	Saexpsim.pc
Description	Export of Revised Sale/Return Transactions from Sales Audit to SIM
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC

Catalog ID	RSA14
Wrapper Script	rmswrap_multi_dnld_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to fetch all revised sale and return transactions that do not have SIM errors from the Sales Audit database tables for transmission to SIM. It retrieves all quantity revision transaction data for SALES, RETURN, EEXCH, VOID, and SPLORD transaction types.

If sa_system_options.unit_of_work is S, the whole store/day is skipped if any SIM error is found. If this value is T, then only transactions with SIM errors are skipped.

The batch will only export transactions whose quantity has been revised. The batch will write these revised transactions to the output file along with a reversal of the quantity.

A file of type SIMT is generated for each store/day.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, updated, and inserted in batches of pl_commit_max_ctr. Only two commits will be done, one to establish the store/day lock and another at the end, to release the lock after a store/day has been completely processed. The SIMT formatted output file will be created with a temporary name and renamed just before the end of store/day commit.

In case of failure, all work done will be rolled back to the point right after the call to get_lock() and the lock released. Thus, the rollback segment should be large enough to hold all inserts into SA_EXPORTED for one store/day.

I/O Specification

Integration Type	Download from Sales Audit
File Name	SIMT_ appended by store number, business date, and system date
Integration Contract	IntCon000045

Output File Layout

Table 6-33 Output File

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type.
	File Line Id	Char(10)	0000000001	Sequential file line number.
	File type definition	Char(4)	SIMT	Identifies the file type.
	Store	Number(10)	N/A	Store location.

Table 6-33 (Cont.) Output File

Record Name	Field Name	Field Type	Default Value	Description
THEAD	Business Date	Char(8)	N/A	Business Date in YYYYMMDD format.
	File Create Date	Char(14)	N/A	File Create Date in YYYYMMDDHHMMSS format.
	Record descriptor	Char(5)	THEAD	Identifies the file record type.
	File Line Id	Char(10)	N/A	Sequential file line number.
	Transaction Number	Number(10)	N/A	Transaction Identifier.
	Revision Number	Number(3)	N/A	Revision Number of the transaction.
	Transaction date	Char(14)	N/A	Transaction date in YYYYMMDDHHMMSS format. Corresponds to the date that the transaction occurred.
	Transaction Type	Char(6)	N/A	Transaction Type.
TDETL	POS Transaction Indicator	Char(1)	N/A	Indicates if the transaction was received from POS or manually created. Valid values: Y - POS N - Manual
	Record descriptor	Char(5)	TDETL	Identifies the file record type.
	File Line Id	Char(10)	N/A	Sequential file line number.
	Item Sequence Number	Number(4)	N/A	Item sequence number.
	Item	Char(25)	N/A	Identifies the merchandise item.
	Item number type	Char(6)	N/A	Identifies the type of item number if the item type is ITEM or REF.
	Item Status	Char(6)	N/A	Status of the item within the transaction, V for item void, S for sold item, R for returned item. ORI - Order Initiate ORC - Order Cancel ORD - Order Complete LIN - Layaway Initiate LCA - Layaway Cancel LCO - Layaway Complete
	Serial Number	Char(128)	N/A	Unique ID.
	Pack Indicator	Char(1)	N/A	Pack Indicator.
	Catchweight Indicator	Char(1)	N/A	Catchweight Indicator.
	Quantity Sign	Char(1)	N/A	Sign of the quantity.

Table 6-33 (Cont.) Output File

Record Name	Field Name	Field Type	Default Value	Description
	Quantity Value	Number(12)	N/A	Number of items, with 4 implied decimal places.
	Standard Unit of Measure	Char(4)	N/A	Standard Unit of Measure of the item.
	Selling Unit of Measure	Char(4)	N/A	Unit of Measure of the quantity value.
	Waste Type	Char(6)	N/A	Waste Type.
	Waste Percent	Number(12)	N/A	Waste Percent.
	Drop Ship Indicator	Char(1)	N/A	Indicates whether the item is part of a drop shipment.
	Actual Weight	Number(12)	N/A	Contains the weight of the item sold, with 4 implied decimal places.
	Actual Weight Sign	Char(1)	N/A	Sign of the actual weight.
	Reason Code	Char(6)	N/A	Reason entered by the cashier for some transaction types.
	Sales Value	Number(20)	N/A	Transaction value, with 4 implied decimal places
	Sales Value Sign	Char(1)	N/A	Transaction value sign.
	Unit Retail	Number(20)	N/A	Unit retail, with 4 implied decimal places.
	Sales Type	Char(1)	N/A	Indicates if the transaction is an In Store Customer Order, External Customer Order, or Regular Sale.
	Customer Order Number	Char(48)	N/A	Contains the customer order ID.
	Customer Order Type	Char(6)	N/A	Customer order type.
	Fulfillment Order Number	Char(48)	N/A	Contains the order ID of the fulfillment order.
	Customer Order Line Number	Number(6)		Contains customer order line number.
TTAIL	Record descriptor	Char(5)	TTAIL	Identifies the file record type.
	File Line Id	Char(10)	N/A	Sequential file line number.
	Tran Record Counter	Number(6)	N/A	Number of TDETL records in this transaction set.
FTAIL	Record descriptor	Char(5)	FTAIL	Identifies the file record type.
	File Line Id	Number(10)	N/A	Sequential file line number.
	File Record counter	Number(10)	N/A	Number of records/transactions processed in the current file (only records between head and tail).

Design Assumptions

N/A

Export of Revised Sale/Return Transactions from Sales Audit to Store Inventory Management (saexport_sales_to_sim)

Module Name	saexport_sales_to_sim.ksh
Description	Export of Revised Sale/Return Transactions from ReSA to SIOCS in a Direct Database integration.
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	KSH
Catalog ID	
Wrapper Script	

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to fetch all revised sale and return transactions that do not have SIOCS errors from the Sales Audit database tables for transmission to Store Inventory Operations System (SIOCS). It retrieves all quantity revision transaction data for SALES, RETURN, EEXCH, VOID, and SPLORD transaction types.

If `sa_system_options.unit_of_work` is S, then the whole store/day is skipped if any SIOCS error is found. If this value is T, then only transactions with SIOCS errors are skipped.

The batch will only export transactions whose quantity has been revised. The batch will write these revised transactions to the `SA_EXPSIM_REVISED_SALES_HEAD` and `SA_EXPSIM_REVISED_SALES_DETAIL` interface tables along with a reversal of the quantity. It will also populate the `ICL_SA_REVISED_SALES` table to signal the Store Inventory Operations system to process the data.

Note

This batch can be run in two modes – trickle mode and batch mode. If Y is passed as a parameter while running the batch, then the batch would run in trickle mode else if N or no parameter is passed, it would then run in normal batch mode.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Only two commits will be done: one to establish the store/day lock, and another at the end to release the lock after a store/day has been completely processed. In case of failure, all work done will be rolled back to the point right after the call to `get_lock()` and releases the lock. Thus, the rollback segment should be large enough to store all inserts into `SA_EXPORTED` for one store/day.

Tables Affected

Table	Select	Insert	Update	Delete
ICL_SA_REVISED_SALES	No	Yes	No	No
SA_EXPSIM_REVISED_SALES_HEAD	No	Yes	No	No
SA_EXPSIM_REVISED_SALES_DETAIL	No	Yes	No	No
SA_STORE_DAY	Yes	No	No	No
SA_EXPORT_LOG	Yes	No	Yes	No
SA_EXPORTED	Yes	Yes	Yes	No
SA_EXPORTED_REV	Yes	No	No	No
SA_TRAN_HEAD	Yes	No	No	No
SA_TRAN_HEAD_REV	Yes	No	No	No
SA_TRAN_ITEM	Yes	No	No	No
SA_TRAN_ITEM_REV	Yes	No	No	No
SA_ERROR	Yes	No	No	No
SA_ERROR_IMPACT	Yes	No	No	No

Design Assumptions

N/A

Export to Universal Account Reconciliation System from Sales Audit (saexpuar)

Module Name	saexpuar.pc
Description	Export to Universal Account Reconciliation System from Sales Audit
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA06
Wrapper Script	N/A

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The SAEXPUAR program is used to select the lottery, bank deposit, money order, and credit card totals and write them to output files for export to an external account clearing house application. For each store day, saexpuar posts specified totals to their appropriate output files.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, updated, and inserted in batches of commit_max_ctr. Only two commits will be

done. One to establish the store/day lock (this will be done by the package) and the other is done at the end, after a store/day has been completely processed.

I/O Specification

Integration Type	Download from Sales Audit
File Name	UAR usage type appended with system date.
Integration Contract	IntCon000046

Output File Layout

The output file will contain one line for each store/day detail record in a comma-delimited format. The fields are surrounded by double quotes. For example, a record for store 1000 on May 20, 2001 with an amount of 19.99 will look something like this:

```
"1", "1000", "1999", "20010520", "2", "", "1", "", "", "", "", "", "", "", "", "MN", "RET"
```

Table 6-34 Output File

Field Name	Field Type	Description
Detail Flag	Char	"1" for detail record.
Store	Number	Store number.
Amount	Number	Total Value * 100 (with 2 implied decimal places).
TranDate	Char	Transaction Date in YYYYMMDD format.
UAR TranCode	Char	Transaction Code. "1" for negative amount, "2" for positive amount.
User Defined Value 1	Char	Ref Number 1 on SA_TOTAL.
User Defined Value 2	Char	Total Seq Number on SA_TOTAL.
User Defined Value 3	Char	Ref Number 2 on SA_TOTAL.
User Defined Value 4	Char	Ref Number 3 on SA_TOTAL.
User Defined Value 5	Char	Not used.
User Defined Value 6	Char	Not used.
User Defined Value 7	Char	Not used.
User Defined Value 8	Char	Not used.
User Defined Value 9	Char	Not used.
User Defined Value 10	Char	Not used.
State	Char	State.
Account	Char	Total Identification on SA_TOTAL.

Design Assumptions

N/A

Extract of POS Transactions by Store/Date from Sales Audit for Web Search (ang_saplgen)

Module Name ang_saplgen.pc

Description	Extract of POS Transactions by Store/Date from Sales Audit for Web Search
Functional Area	Sales Audit
Module Type	Integration
Module Technology	ProC
Integration Catalog ID	RMS162
Wrapper Script	N/A

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to fetch all corrected sale and return transactions that do not have Merchandising errors from the Sales Audit database tables for transmission to an external web search engine. If the transaction has a status of Deleted or Post Voided and has previously been transmitted, a reversal of the transaction will be sent. A file of type POSLOG is generated for each store/day.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, in batches of pl_commit_max_ctr. The POSLOG formatted output file will be created with a completion of store/day looping.

I/O Specification

Integration Type	Download from Sales Audit
File Name	POSLOG_<store>_<business date>_<system date>.xml
Integration Contract	IntCon000018

Output File Layout

Table 6-35 Output File Layout

Field Name	Field Type	Description
BatchID	CHAR(18)	A concatenation of store number and business date for a store.
RetailStoreID	CHAR(10)	The store number for which the POSLog file has to be extracted.
WorkStationID	CHAR(5)	RegistryID for the store.
TillID	CHAR(5)	RegistryID for the store.
SequenceNumber	CHAR(10)	Point of Sale system defined transaction number associated with a transaction.
BeginDate	CHAR(8)	Starting date time of the transaction.
EndDate	CHAR(8)	End date time of the transaction.
CurrencyCode	CHAR(3)	Code of the currency used during the transaction.

Table 6-35 (Cont.) Output File Layout

Field Name	Field Type	Description
VoidFlag	CHAR(5)	Indicates if the item in the transaction is voided or not. Valid values are TRUE and FALSE.
Item_Status	CHAR(40)	Status of the item is required for voided, exchanged, or returned item.
MerchandisingHierarchy	CHAR(4)	Department number to which the item belongs
Description	CHAR(250)	Item description that has been sold.
Item	CHAR(25)	Item number.
TaxIncludedInPrice	CHAR(5)	Indicates if the item is being taxed or not. Valid values are TRUE and FALSE.
RegularSalesUnitPrice	CHAR(20)	Field holds the unit retail in the standard unit of retail for the item/location combination.
ActualSalesUnitPrice	CHAR(20)	Retail price for the item.
ExtendedAmount	CHAR(20)	Total sales for the item in the detail level.
Qty	CHAR(21)	Unit sold of the item.

Design Assumptions

N/A

Post User Defined Totals from Sales Audit to General Ledger (saexpagl)

Module Name	saexpagl.pc
Description	Post User Defined Totals from Sales Audit to General Ledger
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Integration Catalog ID	RSA09
Wrapper Script	rmswrap.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this module is to post all the properly configured user-defined Sales Audit totals to a general ledger application (Oracle or PeopleSoft). Totals without errors will be posted to the appropriate accounting ledger, as defined in the Sales Audit GL cross-reference module. Depending on the unit of work system parameter, the data will be sent at either the store/day or individual total level. Newly revised totals, that have already been posted to the ledger, will have their previous revision reversed, and the new total posted to the appropriate accounts.

When this module encounters a total that is not mapped to the General Ledger (GL), it will write the same into the IF_ERRORS table and raise a notification that unmapped total/store combinations exist. The IF_ERRORS table is available through the Data Access Schema,

which will enable you to query for the error and create the missing mappings. It will also look for records written into IF_ERRORS on previous runs and attempt to reprocess the posting if GL mappings have been created.

'Late posted totals' that are received within a duration specified in the Close Month After Days system option after the end of the fiscal period will be processed by the module and posted to the intended month. All late posted totals received after this specified number of days has elapsed will be recorded against the first day of the subsequent month.

Restart/Recovery

The logical unit of work for this module is defined as a unique store/day combination. Records will be fetched, updated, and inserted in batches the size of commit max counter. Only one commit will be performed after a store/day has been completely processed. A call to the release lock functions performs a commit.

I/O Specification

Integration Type	Download from Sales Audit
File Name	N/A
Integration Contract	IntCon000019 TG_FIF_GL_DATA

Design Assumptions

N/A

Inbound Scheduled Integration

This section provides a summary of integrations that are scheduled either to be run once per day or periodically throughout the day to retrieve data from another solution to Merchandising or Sales Audit. It includes both file-based and BDI-based integrations.

Item, Cost, and Price

Merchandising subscribes to data related to items, costs, and competitive prices from external sources, such as suppliers, PIM solutions, and so on.

The following scheduled inbound integrations are included in this functional area:

- [Upload Competitor's Prices \(cmpupld\)](#)
- [Upload Items and Cost Changes \(iindbatch.ksh\)](#)

Upload Competitor's Prices (cmpupld)

Module Name	cmpupld.pc
Description	Upload Competitor's Prices
Functional Area	Competitive Pricing
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS61

Wrapper Script rmswrap_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program is used to upload and process competitor item prices from an external source. The flat file being uploaded can contain pricing data for a completed shopping list or data for a new list of items to be shopped. The module processes data for both features.

Restart/Recovery

This is a file based upload, and file based restart/recovery logic is applied. The commit_max_ctr field should be set to prevent excessive rollback space usage and to reduce the overhead of file I/O. The recommended commit counter setting is 10000 records (subject to change based on experimentation).

I/O Specification

Integration Type Upload to Merchandising
File Name Determined by runtime parameter
Integration Contract IntCon000007

Input File Layout

Table 6-36 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	CHAR (5)	FHEAD	Value that identifies the record type.
	File Line Identifier	NUMBER (10)	0000000001	Sequential file line number.
	File Type Definition	CHAR(4)	CMPU	Value that identifies the file as that for this program.
	File Create Date	CHAR (14)	N/A	Date when the file was written by the external system. It should be in the YYYYMMDDHH24MISS format.
File Detail	File Type Record Descriptor	CHAR (5)	FDETL	Value that identifies the record type.
	File Line Identifier	NUMBER (10)		Sequential file line number.

Table 6-36 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Shopper ID	NUMBER (4)		Numeric value that uniquely identifies the shopper to which the competitive shopping list is assigned.
	Shop Date	CHAR (14)		Date when the competitive shop was performed. It should be in the YYYYMMDDHH24MISS format.
	Item	CHAR (25)		Alphanumeric value that uniquely identifies the transaction level or below transaction level item that was competitively shopped.
	Competitor ID	NUMBER(10)		Numeric value that uniquely identifies a competitor.
	Competitor Store ID	NUMBER(10)		Numeric value that uniquely identifies a competitor's store.
	Recorded Date	CHAR (14)		Date when the item's retail price was recorded at the competitor's store. It should be in the YYYYMMDD24M ISS format.
	Competitive Retail Price	NUMBER(20,4)		Numeric value that represents the retail price at the competitor's store. Format for this value should include four implied decimal places.

Table 6-36 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Competitive Retail Type	CHAR(6)	R, P, C	Value that represents the retail type ('R' is for regular; 'P', promotional; and 'C', clearance) that was recorded.
	Promotion Start Date	CHAR (14)		Effective start date of the competitor's price. It should be in the YYYYMMDDHH24MISS format.
	Promotion End Date	CHAR (14)		Effective end date of the competitor's price. It should be in the YYYYMMDDHH24MISS format.
	Offer Type Code	CHAR(6)		Alphanumeric value that corresponds to a valid offer type (such as,. Coupon, Bonus Card, Pre-priced). Valid values are defined on CODE_DETAIL table with CODE_TYPE 'OFTP'.
	Multi-Units	NUMBER(12,4)		Numeric value that represents the number of units that must be purchased to qualify for a multi-unit price. An example of a multi-unit price would be 2 for \$3.00. There are four implied decimal places.

Table 6-36 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Multi-Units Retail	NUMBER(20,4)		Numeric value that represents the price for a multi-unit item that was competitively shopped. There should be four implied decimal places.
File Trailer	File Type Record Descriptor	CHAR(5)	FTAIL	Value that identifies the record type.
	File Line Identifier	NUMBER (10)	N/A	Sequential file line number.
	File Record Counter	NUMBER (10)	N/A	Numeric value that represents the number of FDETL records in the file.

Design Assumptions

- Items included in the file must be defined as transaction level items in Merchandising.

Upload Items and Cost Changes (iindbatch.ksh)

Module Name	iindbatch.ksh
Description	Upload items and cost changes from an external system
Functional Area	Item and Cost Maintenance
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS474
Wrapper Script	rmswrap_shell_multi_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program is used to bulk upload XML data files from template files to the Merchandising templates table. It supports two types of templates - those for items and those for cost changes. The templates used in this upload are the same as those used for spreadsheet upload of items and cost changes.

See also *Oracle Retail Merchandising Induction CSV to XML File Transformer Usage* on My Oracle Support (Doc ID 2730273.1), for more details on formatting XML files for this upload.

This batch will be responsible for validating the input parameters, below are the list of validations.

- The Input file should exist.
- The Input file's extension must be ".xml".
- The template name should be valid.
- Destination (Optional Parameter) determines whether data will be loaded into the main Merchandising tables (RMS) or staging tables for further enrichment (STG). If a destination is not included, then it will be defaulted to STG.

Once XML data is loaded into the staging table, the script will do the following:

- Initializes a row in the process tracker table for asynchronous processing.
- Call the main induction process that uploads data into the staging tables, validates and inserts data into the base Merchandising item or cost change tables.

Note

The base templates used by this batch are loaded through a script on provisioning (ITEM_MASTER_DATA and COST_CHANGE). Additional templates can be configured using the Data Loading Template Configuration in the Merchandising task list under Application Administration for type Item or Cost Change.

Restart/Recovery

N/A

Design Assumptions

N/A

Ordering and Inventory

Merchandising subscribes to purchasing and inventory data via scheduled integration from external sources, such as stores, warehouses, order management solutions, and import partners.

This section has been broken into the following sub-sections:

- [Purchasing](#)
- [Import Management](#)
- [Stock Counts](#)
- [Franchise](#)
- [Other Inventory](#)

Purchasing

Merchandising subscribes to data related to purchase orders from external sources, such as suppliers, planning solutions, and so on.

The following scheduled inbound integrations are included in this functional area:

- [Upload of Deals from 3rd Party Systems \(dealupld\)](#)
- [Upload Order Data \(poindbatch.ksh\)](#)
- [Upload OTB Budget from Planning Systems \(otbupld\)](#)
- [Upload Purchase Order and Purchase Order Change Acknowledgements from Suppliers to RMS \(ediupack\)](#)
- [Upload Replenishment Data \(replindbatch.ksh\)](#)

For more on purchase order processing, see *Merchandising Operations Guide - Volume 1*.

Upload of Deals from 3rd Party Systems (dealupld)

Module Name	dealupld.pc
Description	Upload of Deals from 3rd Party Systems
Functional Area	Deals
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS42
Wrapper Script	rmswrap_multi_in_rej.ksh

Design Overview

This process uploads deals from external systems into Merchandising. Generally, deals are uploaded from merchandise suppliers and other trading partners. This program uses a proprietary file format (not any EDI standard).

The deals that are uploaded through the batch are created in the worksheet (W) status by default but can be created in the submitted (S) or approved (A) status as well based on the Deal Upload Status configuration for the supplier. In case of any validation errors during the deal submission or approval, it will be created in the worksheet status and the user needs to manually rectify the error through the Deal UI to approve it.

Note

This functionality is limited to Supplier-based deals. Deals created for other Partners can only be created in the worksheet status.

Backward Compatibility

Added additional parameter to indicate version. If no parameter is given, then the version defaults to 1 (initial version).

Assumptions

1. The fields noted below can be omitted from the file format if you are not creating deals with a billing type of Clearance Consignment Rate (CCR), Promotional Consignment Rate (PCR) or Vendor Funded Promotion (VFP).
 - Consignment Rate in the TDETL record for Transaction Detail Record Type DCDTL.
 - The CPDTL section of the file, including THEAD, TDETL, and TTAIL.
 - a. Clearance Consignment Rate (CCR):

- The DCDTL TDETL Consignment Rate must be included in the upload.
 - The CPDTL section is not required for CCR type.
- b. Promotional Consignment Rate (PCR):
- The CPDTL section is required for PCR type.

Note

The CPDTL TDETL expects only File Type Record Descriptor, File Line Identifier, Prom ID, Promo Comp Id, Consignment Rate, Reference Line.

- c. Vendor Funded Promotion (VPF):
- The CPDTL section is required for VPF type.

Note

The CPDTL TDETL expects File Type Record Descriptor, File Line Identifier, Prom ID, Promo Comp Id, Consignment Rate (to be defaulted to 0 as not required for VFP), Reference Line, Vendor Contrib Type, Contrib Value, and Promo Source.

This is to support backward compatibility.

- The Promo Source field usage is based on the file version being used.
 - If the file version is 1
 - * Promo Source (in the CPDTL TDETL) is not required and will not be validated.
 - If the file version is 2
 - * The Promo Source (in the CPDTL TDETL) is a required field in the CPDTL TDETL
2. In File version number 3, a new billing type Allowance (ALW) has been introduced. Version number 3 need to be explicitly specified by user while calling the batch to create ALW deals.
3. If the input file contains billing type as BB and rebate indicator as Y, then the billing type will be converted to BBR while inserting into the DEAL_HEAD table.

Restart/Recovery

The program uses File based restart recovery process. The logical unit of work is a single deal head detail record and its associated component records in the input file.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000008

Input File Layout

Table 6-37 dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
FHEAD	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type (the beginning of the input file).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	File Type Definition	Char(5)	EDIDU	Identifies file as 'EDI Deals Upload'	
	File Create Date	Char(14)	Create date	Current date, formatted to 'YYYYMMDDHH24MISS'.	
THEAD	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type to upload a new deal header.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Detail Record Type	Char(5)	DHDTL	Identifies file record type Deal Header. This record MUST BE FOLLOWED BY ONE AND ONLY ONE REQUIRED TDETL RECORD that holds the deal head information.	
TDETL	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type to upload a new deal.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Partner Type	Char(6)	REQUIRED	Type of the partner the deal applies to. Valid values are 'S' for a supplier, 'S1' for supplier hierarchy level 1 (for example, the manufacturer), 'S2' for supplier hierarchy level 2 (for example, the distributor) and 'S3' for supplier hierarchy level 3 (that is, the wholesaler). Descriptions of these codes will be held on the codes table under a code_type of 'SUHL'. Information pertaining to a single deal has to belong to the same supplier, since a deal may have only one supplier hierarchy associated with it. Only items with the same supplier hierarchy can be on the same deal. Supplier hierarchy is stored at an item / supplier / country / location level.	
	Partner Id	Char(10)	Blank (space character string)	Level of supplier hierarchy (for example, manufacturer, distributor or wholesaler), set up as a partner in the PARTNER table, used for assigning rebates by a level other than supplier. Rebates at this level will include all eligible supplier/ item/country records assigned to this supplier hierarchy level. This field is required if the Partner Type field was set to 'S1', 'S2' or 'S3'. This field must be blank if the Partner Type field was set to 'S'.	
	Supplier	Number (10)	Blank (space character string)	Deal supplier's number. This supplier can be at any level of supplier hierarchy. This field is required if the Partner Type field was set to 'S'. This field must be blank if the Partner Type field was set to 'S1', 'S2' or 'S3'. Deals for items with an ownership of Consignment can only be set up for the primary supplier for the given item/ location combination.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Type	Char(6)	REQUIRED	Type of the deal. Valid values are A for annual deal, P for promotional deal, O for PO-specific deal or M for vendor-funded markdown. Deal types will be held on the codes table under a code type of 'DLHT'.	
	Currency Code	Char(3)	Blank (space character string)	Currency code of the deal's currency. All costs on the deal will be held in this currency. If Type is 'O', 'P' or 'A', then Currency Code may not be blank. Currency Code has to be blank if Type is 'M'.	
	Active Date	Char(14)	REQUIRED	Date the deal will become active. This date will determine when deal components begin to be factored into item costs. For a PO-specific deal, the active date will be the order's written date.	
	Close Date	Char(14)	Blank (space character string)	Date the deal will/did end. This date determines when deal components are no longer factored into item costs. It is optional for annual deals, required for promotional deals. It will be left NULL for PO-specific deals. Close Date must not be blank if Type is 'P' or 'M'. Close Date has to be blank if Type is 'O'.	
	External Reference Number	Char(30)	Blank (space character string)	Any given external reference number that is associated with the deal.	
	Order Number	Number (12)	Blank (space character string)	Order the deal applies to, if the deal is PO-specific.	
	Recalculate Approved Orders	Char(1)	REQUIRED	Indicates if approved orders should be recalculated based on this deal once the deal is approved. Valid values are Y for yes or N for no. Valid values are 'Y' and 'N'.	
	Comments	Char (2000)	Blank (space character string)	Free-form comments entered with the deal.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Billing Type	Char(6)	REQUIRED	Billing type of the deal component. Valid values are 'OI' for off-invoice, 'BB' for bill-back, 'BBR' for bill-back rebate, 'VFP' for vendor funded promotion, 'VFM' for vendor funded markdown, 'CCR' for clearance consignment rate and 'PCR' for promotional consignment rate and 'ALW' for allowance. Billing types are held in the codes table under a code type of 'DLBT'.	
	Bill Back Period	Char(6)	Blank (space character string)	Code that identifies the bill-back period for the deal component. This field will only be populated for billing types of 'BB' or 'BBR' or 'VFP' or 'VFM'. Valid bill back period codes are 'W', 'M', 'Q', 'H', 'A'. If Billing Type is 'BB' or 'BBR', then Bill Back Period must not be blank; if Billing Type is 'OI' (off invoice), 'CCR' (clearance consignment rate), 'PCR' (promotional consignment rate), then Bill back Period has to be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Deal Invoicing Basis	Char(6)	Blank (space character string)	Indicates when and how the deal component should be applied for invoicing for Purchase Based Bill Back deals. Valid values are 'O' for Approved POs, 'R' for Net Receipts, 'G' for Gross Receipts, and 'S' for Shipment. These values will be held on the codes tables under a code type of 'AALC'. In case of 'O' – Approved POs, the deal would apply based on when the PO was approved, in case of 'R' – Net Receipts and 'G' – Gross Receipts, the deal would apply based on when receipt was made and in case of 'S' – Shipment, the deal would apply based on when shipment was made. In case of Net Receipts, receipts are adjusted for Returns to Vendor (RTVs) and Receiver Unit and Cost Adjustments; while in case of Gross Receipts, gross value of receipts are considered. This attribute is not applicable to an M-type deal (vendor funded markdown) and must be NULL.	
	Threshold Limit Type	Char(6)	Blank (space character string)	Identifies whether thresholds will be set up as quantity values, currency amount values or percentages (growth rebates only). Valid values are 'Q' for quantity, 'A' for currency amount. Threshold limit types will be held on the codes table under a code type of 'DLLT'. It must be NULL for an M-type deal (vendor funded markdown) or if the threshold value type is 'Q' (buy/get deals) or a 'CCR'/PCR' deal. If Growth Rebate Indicator is 'Y', then the Threshold Limit Type has to be 'Q', 'A' or NULL.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Threshold Limit Unit of Measure	Char(4)	Blank (space character string)	Unit of measure of the threshold limits, if the limit type is quantity. Only Unit of Measures with a UOM class of 'VOL' (volume), 'MASS' or 'QTY' (quantity) can be used in this field. Valid Unit of Measures can be found on the UOM_CLASS table. If the Threshold Limit Type is 'A', then Threshold Limit Unit of Measure has to be blank. If the Threshold Limit Type is 'Q', Threshold Limit Unit of Measure must not be blank. If Threshold Limit Type is blank, Threshold Limit Unit of Measure must be blank.	
	Rebate Indicator	Char(1)	REQUIRED	Indicates if the deal component is a rebate. Deal components can only be rebates for bill-back billing types. Valid values are 'Y' for yes or 'N' for no. If Billing Type is 'BB', 'OI', 'CCR' or 'PCR', then Rebate Indicator must be 'N'.	
	Rebate Calculation Type	Char(6)	Blank (space character string)	Indicates if the rebate should be calculated using linear or scalar calculation methods. Valid values are 'L' for linear or 'S' for scalar. This field will be required if the rebate indicator is 'Y'. Rebate calculation types will be held on the codes table under a code type of 'DLCT'. If Rebate Indicator is 'Y', then Rebate Calculation Type must not be blank. Otherwise it has to be blank.	
	Growth Rebate Indicator	Char(1)	REQUIRED	Indicates if the rebate is a growth rebate, meaning it is calculated and applied based on an increase in purchases or sales over a specified period of time. Valid values are 'Y' for yes or 'N' for no. If Rebate Indicator is 'N', then Growth Rebate Indicator must be 'N'.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
TTAIL	Historical Comparison Start Date	Char(14)	Blank (space character string)	The first date of the historical period against which growth will be measured in this growth rebate. Note performance and the rebate amount are not calculated - this field is for informational/reporting purposes only. If Growth Rebate Indicator is 'Y', then Historical Comparison Start Date must not be blank. Otherwise it must be blank.	
	Historical Comparison End Date	Char(14)	Blank (space character string)	The last date of the historical period against which growth will be measured in this growth rebate. Note performance and the rebate amount are not calculated - this field is for informational/reporting purposes only. If Growth Rebate Indicator is 'Y', then Historical Comparison End Date must not be blank. Otherwise it must be blank.	
	Rebate Purchases or Sales Application Indicator	Char(6)	Blank (space character string)	Indicates if the rebate should be applied to purchases or sales. Valid values are 'P' for purchases, 'S' for sales, and 'I' for issues. It will be required if the rebate indicator is 'Y'. Rebate purchase/sales indicators will be held on the codes table under a code type of 'DLRP'. If the Rebate Indicator is 'Y', then the Rebate Purchases or Sales Application Indicator must not be blank. Otherwise it has to be blank.	
	Security Indicator	Char	Y	Security Indicator	
	File Line Identifier	Char(5)	TTAIL	Identifies file record type (the end of the transaction detail).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
THEAD	Transaction Record Counter	Numeric ID(6)	Sequential number Created by program.	Number of records/transactions in current transaction set (only records between thead and ttail). For DHDTL TDETL records this will always be 1!	
	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type to upload a new deal sub loop.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
TDETL	Transaction Detail Record Type	Char(5)	DCDTL	Identifies file record type of sub loop as Deal Component Detail.	
	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type to upload deal components.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Deal Component Type	Char(6)	REQUIRED	Type of the deal component, user-defined and stored on the DEAL_COMP_TYPE table.	
	Application Order	Number (10)	Blank (space character string)	Number indicating the order in which the deal component should be applied with respect to any other deal components applicable to the item within the deal. This number will be unique across all deal components within the deal. It must be NULL for an M-type deal (vendor funded markdown).	
	Collect Start Date	Char(14)	Blank (space character string)	Date that collection of the bill-back should begin. If Billing Type is 'BB' or 'BBR' then Collect Start Date must not be blank, otherwise it has to be blank.	
	Collect End Date	Char(14)	Blank (space character string)	Date that collection of the bill-back should end. If Billing Type is 'BB' or 'BBR' then Collect End Date must not be blank, otherwise it has to be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Cost Application Level Indicator	Char(6)	Blank (space character string)	Indicates what cost bucket the deal component should affect. Valid values are 'N' for net cost, 'NN' for net cost and 'DNN' for dead net cost. These values will be held on the codes tables under a code type of 'DLCA'. It must be NULL for an M-type deal (vendor funded markdown), 'CCR' or 'PCR' deals. It must be N for 'ALW' deal.	
	Pricing Cost Indicator	Char(1)	REQUIRED	Identifies deal components that should be included when calculating a pricing cost. Valid values are 'Y'es and 'N'o.	
	Deal Class	Char(6)	Blank (space character string)	Identifies the calculation class of the deal component. Valid values are 'CU' for cumulative (discounts are added together and taken off as one lump sum), 'CS' for cascade (discounts are taken one at a time with subsequent discounts taken off the result of the previous discount) and 'EX' for exclusive (overrides all other discounts). 'EX' type deal components are only valid for promotional deals. Deal classes will be held on the codes table under a code type of 'DLCL'. It must be NULL for an M-type deal (vendor funded markdown), 'CCR', 'PCR', and 'ALW' deals.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Threshold Value Type	Char(6)	Blank (space character string)	Identifies whether the discount values associated with the thresholds will be set up as qty values, currency amount values, percentages or fixed amounts. Valid values are 'Q' for qty, 'A' for currency amount, 'P' for percentage or 'F' for fixed amount. Qty threshold value (buy/get) deals are only allowed on off-invoice discounts. Deal threshold value types will be held on the codes table under a code type of 'DLL2'. It must be NULL for an M-type deal (vendor funded markdown), 'CCR' and 'PCR' deals. If Billing Type is 'BB' or 'BBR', then the Threshold Value Type must be 'A' or 'P'. If billing type is 'ALW' it can have values 'A' or 'P'	
	Buy Item	Char(25)	Blank (space character string)	Identifies the item that must be purchased for a quantity threshold-type discount. This value is required for quantity threshold value type discounts. Otherwise it has to be blank.	
	Get Type	Char(6)	Blank (space character string)	Identifies the type of the 'get' discount for a quantity threshold-type (buy/get) discount. Valid values include 'X' (free), 'P' (percent), 'A' (amount) and 'F' (fixed amount). They are held on the codes table under a code type of 'DQGT'. This value is required for quantity threshold value deals. Otherwise it has to be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Get Value	Number(20, 4)	All 0s.	Identifies the value of the 'get' discount for a quantity threshold-type (buy/get) discount that is not a 'free goods' deal. The Get Type above identifies the type of this value. This value is required for quantity threshold value type deals that are not a Get Type of free. Otherwise it has to be 0. If Get Type is 'P', 'A' or 'F', then Get Value must not be blank. If the Get Type is 'X' or blank, then Get Value has to be blank.	
	Buy Item Quantity	Number(12, 4)	All 0s.	Identifies the quantity of the threshold 'buy' item that must be ordered to qualify for the 'free' item. This value is required for quantity threshold value type discounts. Otherwise it has to be 0.	
	Recursive Indicator	Char(1)	REQUIRED	For 'buy/get free' discounts, indicates if the quantity threshold discount is only for the first 'buy amt.' purchased (such as, for the first 10 purchased, get 1 free), or if a free item will be given for every multiple of the 'buy amt' purchased on the order (such as, for each 10 purchased, get 1 free). Valid values are 'Y' for yes or 'N' for no. If the Get Type is blank, then Recursive Indicator has to be 'N'.	
	Buy Item Order Target Quantity	Number(12, 4)	All 0s.	Indicates the targeted purchase level for all locations on a purchase order. This is the target level that will be used for future calculation of net cost. This value is required for quantity threshold value type deals. Otherwise it has to be 0.	
	Average Buy Item Order Target Quantity Per Location	Number(12, 4)	All 0s.	Indicates the average targeted purchase level per location on the deal. This value will be used in future cost calculations. This value is required for quantity threshold value type deals. Otherwise it has to be 0.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Get Item	Char(25)	Blank (space character string)	Identifies the 'get' item for a quantity threshold-type (buy/get) discount. This value is required for quantity threshold value deals. Otherwise it has to be blank. If Get Type is 'P', 'A', 'F' or 'X', then Get Item must not be blank. If the Get Type is blank, then Get Item has to be blank.	
	Get Quantity	Number(12, 4)	All 0s.	Identifies the quantity of the identified 'get' item that will be given at the specified 'get' discount if the 'buy amt' of the buy item is purchased. This value is required for quantity threshold value type discounts. Otherwise it has to be 0. If Get Type is 'P', 'A', 'F' or 'X', then Get Quantity must not be 0. If the Get Type is blank, then Get Quantity has to be 0.	
	Free Item Unit Cost	Number(20, 4)	All 0s.	For 'buy/get free' discounts, identifies the unit cost of the threshold 'free' item that will be used in calculating the prorated qty. discount. It will default to the item/supplier cost, but can be modified based on the agreement with the supplier. It must be greater than zero as this is the cost that would normally be charged for the goods if no deal applied. If Get Type is 'P', 'A', 'F' or blank, then Free Item Unit Cost must be 0. If the Get Type is 'X', then Free Item Unit Cost must not be 0.	
	Transaction Level Discount Indicator	Char(1)	REQUIRED	Indicates if the discount is a transaction-level discount (for example, 10% across an entire PO). Valid Values are 'Y' or 'N'. If set to 'Y', Deal Class has to be 'CU' and Billing Type has to be 'OI'. For 'CCR', 'PCR', and 'ALW' deals, the valid value is 'N'. No DIDTL or PPDTL records may be present for a Transaction Level Discount DCDTL record.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Comments	Char(2000)	Blank (space character string)	Free-form comments entered with the deal component.	
	Get Free Discount	Number(12, 4)	All 0s.	This specifies how much percentage of the total discount should be apportioned from the get items unit cost for off invoice deals where buy item is not same as the get item and QTY_THRESH_GET_TYPE is X. The remaining will be apportioned from the buy item unit cost.	
	Consignment Rate	Number(12, 4)	All 0s.	Rate used to capture the deal consignment rate applicable for the set of item/location combinations that are included in the deal during the deal timeframe, instead of the regular consignment rate.	
TTAIL	File Line Identifier	Char(5)	TTAIL	Identifies file record type (the end of the transaction detail).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Record Counter	Numeric ID(6)	Sequential number Created by program.	Number of records/transactions in current transaction set (only records between thead and ttail).	
THEAD	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type to upload a new deal sub loop.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Detail Record Type	Char(5)	CPDTL	Identifies file record type of sub loop as Deal Component Prom.	
TDETL	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type to upload deal proof of performance details.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Prom ID	Number (10)	All 0s.	Promotion identification number.	
	Promo Comp Id	Number (10)	All 0s.	Promotion offer identification number.	
	Consignment Rate	Number(12, 4)	All 0s.	Rate used to capture the deal consignment rate applicable for the set of item/location combinations that are included in the deal during the deal timeframe, instead of the regular consignment rate.	
	Reference Line	Number (10)	REQUIRED	This value determines which line in the input file this deal component-promotional record belongs to.	
	Vendor Contrib Type	Varchar2(6)	Blank (space character string).	Vendor Contribution Type can be 'P' Percent or 'A' Amount, and is required for VFP Deal. This field is not required for PCR Deal	
	Contrib Value	Number(20, 4)	All 0s	Rate used to capture the deal Contribution Value applicable for the set of item/location combinations that are included in the deal during the deal timeframe. If Vendor Contrib Type is 'P', then the value should be between 0 and 100. For Vendor Contrib Type 'A', a positive value is required.	
	Promo Source	Varchar2(2)	Blank (space character string).	This contains the Promo Source. Valid values are 'MP' (Merch Pricing) or 'CE' (Customer Engagement). If the Promo Source is 'MP', the promo_id and promo_comp_id (offer_id) should be from Pricing. If the Promo Source is 'CE', the promo_id (promotion_id) and promo_comp_id (deal_id) should be from the Customer Engagement (CE).	2
TTAIL	File Line Identifier	Char(5)	TTAIL	Identifies file record type (the end of the transaction detail).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
THEAD	Transaction Record Counter	Numeric ID(6)	Sequential number Created by program.	Number of records/transactions in current transaction set (only records between thead and ttail).	
	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type to upload a new deal sub loop.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
TDETL	Transaction Detail Record Type	Char(5)	DIDTL	Identifies file record type of sub loop as Deal Component Item-location Detail.	
	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type to upload deal item-location details.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Merchandise Level	Char(6)	REQUIRED	Indicates what level of the merchandise hierarchy the record is at. Valid values include '1' for company-wide (all items), '2' for division, '3' for group, '4' for dept, '5' for class, '6' for subclass, '7' for line, '8' for line/differentiator 1, '9' for line/differentiator 2, '10' for line/differentiator 3, '11' for line/differentiator 4 and '12' for. These level types will be held on the codes table under a code type of 'DIML'.	
	Company Indicator	Char(1)	REQUIRED	Indicates if the deal component is applied company-wide (that is, whether all items in the system will be included in the discount or rebate). Valid values are 'Y' for yes and 'N' for no.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Division	Number (4)	Blank (space character string)	ID of the division included in or excluded from the deal component. Valid values are on the DIVISION table. If Group is not blank, then Division must not be blank. If Merchandise Level is 2, then Division must not be blank and Group, Department, Class and Subclass must be blank.	
	Group	Number (4)	Blank (space character string).	ID of the group included in or excluded from the deal component. Valid values are on the GROUPS table. If Department is not blank, then Group must not be blank. If Merchandise Level is 3, then Group must not be blank and Department, Class and Subclass must be blank.	
	Department	Number (4)	Blank (space character string).	ID of the department included in or excluded from the deal component. Valid values are on the DEPS table. If Class is not blank, then Department must not be blank. If Merchandise Level is 4, then Department must not be blank and Class and Subclass must be blank.	
	Class	Number (4)	Blank (space character string).	ID of the class included in or excluded from the deal component. Valid values are on the CLASS table. If Subclass is not blank, then Class must not be blank. If Merchandise Level is 5, then Class must not be blank and Subclass must be blank.	
	Subclass	Number (4)	Blank (space character string).	ID of the subclass included in or excluded from the deal component. Valid values are on the SUBCLASS table. If Merchandise Level is 6 or more than 6, then Subclass must not be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Item Parent	Char(25)	Blank (space character string)	Alphanumeric value that uniquely identifies the item/group at the level above the item. This value must exist as an item in another row on the ITEM_MASTER table. If Merchandise Level is 7, then Item Parent or Item Grandparent must not be blank (at least one of them has to be given).	
	Item Grandparent	Char(25)	Blank (space character string)	Alphanumeric value that uniquely identifies the item/group two levels above the item. This value must exist as both an item and an item parent in another row on the ITEM_MASTER table. If Merchandise Level is 7, then Item Parent or Item Grandparent must not be blank (at least one of them has to be given).	
	Differentiator 1	Char(10)	Blank (space character string)	Diff_group or diff_id that differentiates the current item from its item_parent. If Item Grandparent, Item Parent and Differentiator 2 are blank, then Differentiator 1 must be blank. If Merchandise Level is 8, then Differentiator 1 must not be blank.	
	Differentiator 2	Char(10)	Blank (space character string)	Diff_group or diff_id that differentiates the current item from its item_parent. If Item Grandparent, Item Parent and Differentiator 1 are blank, then Differentiator 2 must be blank. If Merchandise Level is 9, then Differentiator 2 must not be blank.	
	Differentiator 3	Char(10)	Blank (space character string)	Diff_group or diff_id that differentiates the current item from its item_parent. If Item Grandparent, Item Parent and Differentiator 1 and 2 are blank, then Differentiator 3 must be blank. If Merchandise Level is 10, then Differentiator 3 must not be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Differentiator 4	Char(10)	Blank (space character string)	Diff_group or diff_id that differentiates the current item from its item_parent. If Item Grandparent, Item Parent and Differentiator 1, 2 and 3 are blank, then Differentiator 4 must be blank. If Merchandise Level is 10, then Differentiator 4 must not be blank.	
	Organizational Level	Char(6)	Blank (space character string)	Indicates what level of the organizational hierarchy the record is at. Valid values include '1' for chain, '2' for area, '3' for region, '4' for district and '5' for location. These level types will be held on the codes table under a code type of 'DIOL'. If company indicator is N, this must not be blank. If location type is warehouse or location list, this must be 5.	
	Chain	Number (10)	Blank (space character string).	ID of the chain included in or excluded from the deal component. Valid values are on the CHAIN table. If org. level is 1, this field must not be blank.	
	Area	Number (10)	Blank (space character string).	ID of the area included in or excluded from the deal component. Valid values are on the AREA table. If org. level is 2, this field and chain must not be blank.	
	Region	Number (10)	Blank (space character string).	ID of the region included in or excluded from the deal component. Valid values are on the REGION table. If org. level is 3, this field, area, and chain must not be blank.	
	District	Number (10)	Blank (space character string).	ID of the district included in or excluded from the deal component. Valid values are on the DISTRICT table. If org. level is 4, then this field, region, area, and chain must not be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
TTAIL	Location	Number (10)	Blank (space character string).	ID of the location included in or excluded from the deal component. Valid values are on the STORE, WH, or LOC_LIST_HEAD table. If org. level is 5, this field must not be blank. Chain, area, region, and district should be blank if the loc_type is L or W. If the loc_type is S, then they all must not be blank. If Location Type is not blank, then Location must not be blank. Otherwise it has to be blank.	
	Origin Country Identifier	Char(3)	Blank (space character string)	Origin country of the item that the deal component should apply to.	
	Location Type	Char(1)	Blank (space character string)	Type of the location referenced in the location field. Valid values are 'S' and 'W'. Location types will be held on the codes table under the code type 'LOC3'. If location is blank then this field has to be blank also.	
	Item	Char(25)	Blank (space character string)	Unique alphanumeric value that identifies the item. If Merchandise Level is 10, then Item must not be blank.	
	Exclusion Indicator	Char(1)	REQUIRED	Indicates if the deal component item/location line is included in the deal component or excluded from it. Valid values are 'Y' for yes or 'N' for no.	
	Reference Line	Number (10)	REQUIRED	This value determines which line in the input file this item-loc record belongs to.	
	File Line Identifier	Char(5)	TTAIL	Identifies file record type (the end of the transaction detail).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Record Counter	Numeric ID(6)	Sequential number Created by program.	Number of records/transactions in current transaction set (only records between thead and ttail).	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
THEAD	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type to upload a new deal sub loop.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Detail Record Type	Char(5)	PPDTL	Identifies file record type of sub loop as Proof of Performance Detail.	
TDETL	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type to upload deal proof of performance details.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Deal Sub Item	Char(25)	No data	Specific transaction level (or below) item that's proof of performance is being measured. This can be populated when the deal itself is on a case UPC but the proof of performance is on an individual selling unit.	
	Proof of Performance Type	Char(6)	REQUIRED	Code that identifies the proof of performance type (that is, the term is that the item must be displayed on an end cap for 28 days - the pop_type is code 'ECD' for end cap display). Valid values for this field are stored in the code_type = 'PPT'. This field is required by the database.	
	Proof of Performance Value	Number (20,4)	All 0s.	Value that describes the term of the proof of performance type (that is, the term is that the item must be displayed on an end cap for 28 days - the pop_value is 28). This field is required by the database if the record has a pop_value_type. If Proof of Performance Value is not blank, then Proof of Performance Value Type must not be blank. If Proof of Performance Value is blank, then Proof of Performance Value Type must be blank.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
TTAIL	Proof of Performance Value Type	Char(6)	Blank (space character string)	Value that describes the type of the pop_value (that is, the term is that the item must be displayed on an end cap for 28 days - the pop_value_type is the code 'DAYS' for days). Valid values for this field are stored in the code_type = 'PPVT'. This field is required by the database if the record has a pop_value. If Proof of Performance Value is not blank, then Proof of Performance Value Type must not be blank. If Proof of Performance Value is blank, then Proof of Performance Value Type must be blank.	
	Vendor Recommended Start Date	Char(14)	Blank (space character string)	This column holds the date that the vendor recommends that the POP begin.	
	Vendor Recommended End Date	Char(14)	Blank (space character string)	This column holds the date that the vendor recommends that the POP end.	
	Planned Start Date	Char(14)	Blank (space character string)	This column holds the date that the merchandiser/category manager plans to begin the POP.	
	Planned End Date	Char(14)	Blank (space character string)	This column holds the date that the merchandiser/category manager plans to end the POP.	
	Comment	Char(255)	Blank (space character string)	Free-form comments.	
	Reference Line	Number (10)	REQUIRED	This value determines which line in the input file this Proof of Performance record belongs to.	
	File Line Identifier	Char(5)	TTAIL	Identifies file record type (the end of the transaction detail).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Record Counter	Numeric ID(6)	Sequential number Created by program.	Number of records/transactions in current transaction set (only records between thead and ttail).	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
THEAD	File Type Record Descriptor	Char(5)	THEAD	Identifies file record type to upload a new deal sub loop.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Detail Record Type	Char(5)	DTDTL	Identifies file record type of sub loop as Deal Component Threshold Detail.	
TDETL	File Type Record Descriptor	Char(5)	TDETL	Identifies file record type to upload deal threshold details.	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Lower Limit	Number (20,4)	REQUIRED	Lower limit of the deal component. This is the minimum value that must be met in order to get the specified discount. This value will be either a currency amount or quantity value, depending on the value in the deal_detail.threshold_limit_type field of this deal component (Threshold Value Type field of the DCDTL record that this DTDTL record belongs to as specified in the reference line field).	
	Upper Limit	Number (20,4)	REQUIRED	Upper limit of the deal component. This is the maximum value for which the specified discount will apply. This value will be either a currency amount or quantity value, depending on the value in the deal_detail.threshold_limit_type field of this deal component (Threshold Value Type field of the DCDTL record that this DTDTL record belongs to as specified in the reference line field).	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	Value	Number (20,4)	REQUIRED	Value of the discount that will be given for meeting the specified thresholds for this deal component. This value will be either a currency amount or quantity value, depending on the value in the deal_detail.threshold_value_type field of this deal component (Threshold Value Type field of the DCDTL record that this DTDTL record belongs to as specified in the reference line field).	
	Target Level Indicator	Char(1)	REQUIRED	Indicates if a threshold level is the targeted purchase or sales level for a deal component. This indicator will be used for cost calculations. Valid values are 'Y' for yes and 'N' for no.	
	Reference Line	Number (10)	REQUIRED	This value determines which line in the input file this Threshold record belongs to.	
	Allowance_comp_id	Char(10)	Blank (space character string)	This field will hold an allowance component that contains the details on how the allowance is to be calculated. This field is mandatory if the billing type is 'ALW'.	3
	Allowance_comp_rate	Number(20,4)	Blank (space character string)	The field will hold the rate for an allowance	3
TTAIL	File Line Identifier	Char(5)	TTAIL	Identifies file record type (the end of the transaction detail).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	
	Transaction Record Counter	Numeric ID(6)	Sequential number Created by program.	Number of records/transactions in current transaction set (only records between thead and ttail).	
FTAIL	File Line Identifier	Char(5)	FTAIL	Identifies file record type (the end of the input file).	
	File Line Identifier	Numeric ID(10)	Sequential number Created by program.	ID of current line being read from input file.	

Table 6-37 (Cont.) dealupld.pc - Input File Layout

Record Name	Field Name	Field Type	Default Value	Description/Constraints	Added in Version
	File Record Counter	Numeric ID(10)	Sequential number Created by program.	Number of records/transactions in current file (only records between head and tail).	

The input file structure should be as below:

```

FHEAD
{
THEAD of DHDTL   REQUIRED   for deal head record
  TDETL          REQUIRED   1 deal head record
  TTAIL          REQUIRED   end of deal head record
THEAD of DCDTL   REQUIRED   for deal component records
[
  TDETL          OPTIONAL  for deal component records
]
TTAIL           REQUIRED   end of deal component records
THEAD of CPDTL   OPTIONAL  for deal component promotion records
[
  TDETL          OPTIONAL  for deal component promotion records
]
TTAIL           OPTIONAL  end of deal component promotion records
THEAD of DIDTL   REQUIRED   for item-loc records
[
  TDETL          OPTIONAL  for item-loc records
]
TTAIL           REQUIRED   end of item-loc records
THEAD of PPDTL   REQUIRED   for proof of performance records
[
  TDETL          OPTIONAL  for proof of performance records
]
TTAIL           REQUIRED   end of proof of performance records
THEAD of DTDTL   REQUIRED   for threshold records
[
  TDETL          OPTIONAL  for threshold records
]
TTAIL           REQUIRED   end of threshold records
}
FTAIL
THEAD of DIDTL   REQUIRED   for item-loc records
[
  TDETL          OPTIONAL  for item-loc records
]
TTAIL           REQUIRED   end of item-loc records
THEAD of PPDTL   REQUIRED   for proof of performance records
[
  TDETL          OPTIONAL  for proof of performance records
]
TTAIL           REQUIRED   end of proof of performance records
THEAD of DTDTL   REQUIRED   for threshold records
[
  TDETL          OPTIONAL  for threshold records
]
TTAIL           REQUIRED   end of threshold records

```

}
FTAIL

Vendor Contrib Type	Varchar2(6)	Blank (space character string).	Vendor Contribution Type can be 'P' Percent or 'A' Amount, and is required for VFP Deal. This field is not required for PCR Deal
Contrib Value	Number(20,4)	All 0s	Rate used to capture the deal Contribution Value applicable for the set of item/ location combinations that are included in the deal during the deal timeframe. If Vendor Contrib Type is 'P', then the value should be between 0 and 100. For Vendor Contrib Type 'A', a positive value is required.

Promo Source	Varchar2(2)	Blank (space character string).	This contains the Promo Source. Valid values are 'MP' (Merch Pricing) or 'CE' (Customer Engagement). If the Promo Source is 'MP', the promo_id and promo_comp_id (offer_id) should be from Pricing. If the Promo Source is 'CE', the promo_id (promotion_id) and promo_comp_id (deal_id) should be from the Customer Engagement (CE).	2

Upload Order Data (poindbatch.ksh)

Module Name	poindbatch.ksh
Description	Upload Order Data
Functional Area	Purchase Order Maintenance
Module Type	Integration
Module Technology	Ksh
Catalog ID	RMS234
Wrapper Script	rmswrap_shell_multi_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program is used to Bulk upload xml file data from template files to S9T_FOLDER table (into content_xml column).

This batch will be responsible for validating the input parameters, below are the list of validations.

- The Input file should exist.
- The Input file's extension must be ".xml".
- The template_name should be valid. Function S9T_PKG.CHECK_TEMPLATE is called for validation.

- Destination (Optional Parameter) should be STG or Merchandising. If destination is not passed then default it to STG.

Once XML data is loaded into S9T_FOLDER table, the script will do post processing by calling the packages listed below:

- PO_INDUCT_SQL.INIT_PROCESS - This initialize a row in svc_process_tracker for asynchronous processing.
- PO_INDUCT_SQL.EXEC_ASYNC - This function calls the main induction process that uploads data into the staging tables, validates and inserts data into the base Merchandising purchase order tables.

Note

The base templates used by this batch are loaded through a script on provisioning (PURCHASE_ORDER_DATA). Additional templates can be configured using the Data Loading Template Configuration in the Merchandising task list under Application Administration for type Purchase Orders.

Restart/Recovery

N/A

Design Assumptions

N/A

Upload OTB Budget from Planning Systems (otbupld)

Module Name	otbupld.pc
Description	Upload OTB Budget from Planning Systems
Functional Area	Open To Buy
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS132
Wrapper Script	rmswrap_multi_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this batch module is to accept new and updated open to buy (OTB) budget data from an external planning system. Merchandising supports three types of OTB budgets – those associated with Non-Basic (N/B), Buyer Replenished Basic (BRB) and Auto-Replenished Basic (ARB) orders, as defined by the Order type on Merchandising purchase orders. OTB budgets are created by subclass/end of week date in Merchandising.

Restart/Recovery

Processing of each row is independent and thus if an erroneous record is found during processing; only that record needs to be corrected and reprocessed.

If a record fails validation, it will be written to a rejected record file. This file will facilitate easy reprocessing once the error is fixed by writing the record exactly as it was in the source file.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000033

Input File Layout

Table 6-38 otbupld - Input File

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File head descriptor	Char(5)	FHEAD	Describes file line type
	Line id	Number(10)	0000000001	Sequential file line number
	File Type Definition	Char(4)	'OTBI'	Identifies file as 'OTB Import'
	File Create Date	Char(14)	N/A	The date on which the file was written by external system. The Date is in YYYYMMDDHH 24MISS format
	Subclass	Number(4)		The ID number of a subclass within the class given
FDETL	Eow Date	Char(14)		The end of week date for the budgeted week in YYYYMMDDHH 24MISS format
	File record descriptor	Char(5)	FDETL	Describes file line type
	Line ID	Number(10)	N/A	Sequential file line number
	Transaction Set Control Number	Number(14)	N/A	Sequence number used to force unique transaction check

Table 6-38 (Cont.) otbupld - Input File

Record Name	Field Name	Field Type	Default Value	Description
	Order Type	Char(1)	N/A	Order type budgeted for: specified as A for ARB, B for BRB, and N for N/B
	Department	Number(4)	N/A	The ID number of a department
	Class	Number(4)	N/A	The ID number of a class within the department given
	Subclass	Number(4)	N/A	The ID number of a subclass within the class given
	Eow Date	Char(14)	N/A	The end of week date for the budgeted week in YYYYMMDDHH 24MISS format
	Budget Amount	Number(20)	N/A	Budgeted amount for the specified order type/week; value includes 4 implied decimal places
FTAIL	File record descriptor	Char(5)	N/A	Marks end of file
	Line ID	Number(10)	Line number in file	Sequential file line number
	Number of lines	Number(10)	Total detail lines	Number of lines in file not counting FHEAD and FTAIL

Design Assumptions

- POs with an Order Type of DSD and Customer Order do not impact open to buy.

Upload Purchase Order and Purchase Order Change Acknowledgements from Suppliers to RMS (ediupack)

Module Name	ediupack.pc
Description	Upload Purchase Order and Purchase Order Change Acknowledgements from Suppliers to Merchandising

Functional Area	Purchase Orders
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS48
Wrapper Script	rmswrap_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program has four functions:

1. to acknowledge vendor receipt of a buyer-generated order without changes (acknowledge type AK)
2. to acknowledge vendor receipt of a buyer-generated order with date, cost or quantity modifications (acknowledge type AK)
3. to notify buyer of a new or updated vendor-generated order (acknowledge type AP)
4. to acknowledge order cancellations (acknowledge type CA)

All acknowledgements update the ORDHEAD table with acknowledgement information.

When the supplier sends the acknowledgement of a buyer order with modifications, they can send the entire purchase order or only the changes. The file details are matched to the current order. If the Not Before Date, Not After Date, Quantity, Price, and item all match the current order, then no changes were submitted. If one of the variables is blank, for example the price, assume that no pricing changes were made. As soon as one of the variables does not match, the order has been changed. These changes will not be written directly to the order; they will be written to the revision tables. Revisions will be accepted in the on-line ordering screens and changed orders will be resubmitted via EDIDLORD.

Vendor generated orders will create new orders by inserting new records on the EDI temporary order tables that are picked up by a subsequent process (VRPLBLD). For revisions to a vendor-generated order, updates will be made to the order automatically without requiring user acceptance. If the update is to add a new item/location to the order, this will generate a new purchase order using the same vendor reference number.

For Customer Order POs created through an external Order Management System (OMS) and Franchise Order POs, the modifications to the dates, quantity and cost are applied automatically (and will not need to be accepted online). Also, changes to Franchise POs through this program will not affect their associated Franchise orders.

Restart/Recovery

The files will not have enough volume to warrant the implementation of restart recovery for commit/rollback considerations but minimal file-based restart/recovery capability will be added. The logical unit of work is a complete transaction represented by detail lines between the transaction header and transaction tail.

A savepoint will be issued before each transaction header record is successfully processed. If a non-fatal error occurs, a rollback to the last savepoint will be issued so that the rejected records are not posted to the database. If a fatal error occurs and restart is necessary, processing will restart at the last commit point.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000014

Input File Layout

Table 6-39 ediupack - Input File

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File head descriptor	Char(5)	FHEAD	Describes file line type
	Line id	Number(10)	0000000001	Sequential file line number
	File Type Definition	Char(4)	ORAK	Identifies file as 'Order Acknowledgment Import'
THEAD	File record descriptor	Char(5)	THEAD	Describes file line type
	Line id	Number(10)	Line number in file	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number
	Acknowledge type	Char(2)	N/A	AP-product replenishment (VMI orders and updates) AK- Acknowledge or change CA-cancel order (no detail)
	Order number	Char(15)	N/A	May be external order number (vendor order number) OR Oracle Retail order number
	Written_date	Char(8)	N/A	Written date in YYYYMMDD format
	Supplier number	Number(10)	N/A	Supplier number
	Not before date	Char(8)	N/A	Not_before_date YYYYMMDD
	Not after date	Char(8)	N/A	Not_after_date YYYYMMDD
	Purchase type	Char(6)	N/A	Specifies type of purchase – may be blank
	Pickup date	Char(8)	N/A	Pickup_date YYYYMMDD – may be blank
TITEM	File record descriptor	Char(5)	TITEM	Describes file line type
	Line id	Number(10)	Line number in file	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number
	ITEM	Char(25)	N/A	Item (either item or ref_item must be defined)
	Ref_item	Char(25)	N/A	Reference item (either item or ref_item must be defined)

Table 6-39 (Cont.) ediupack - Input File

Record Name	Field Name	Field Type	Default Value	Description
TSHIP	Vendor catalog number	Char(30)	N/A	VPN (Vendor Product Number)
	Unit cost value	Number(20)	N/A	Unit_cost * 10000 (4 implied decimal places)
	Loc_type	Char(2)	N/A	'ST' for store, 'WH' for warehouse
	Location	Number(10)	N/A	If NULL, apply to all locations for this item
	Pickup location	Char(250)	N/A	Location to pick up item – may be blank
	File record descriptor	Char(5)	TSHIP	Describes file line type
	Line id	Number(10)	Line number in file	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number
	Store/wh indicator	Char(2)	N/A	'ST' for store, 'WH' for warehouse
	Ship to location	Number(10)	N/A	Store or warehouse number
TTAIL	Quantity	Number(12)	N/A	Quantity ordered * 10000 (4 implied decimal places)
	File record descriptor	Char(5)	TTAIL	Describes file line type
	Line id	Number(10)	Line number in file	Sequential file line number
	Transaction number	Number(10)	N/A	Sequential transaction number
FTAIL	Lines in transaction	Number(6)	N/A	Total number of lines in this transaction
	File record descriptor	Char(5)	FTAIL	Marks end of file
	Line id	Number(10)	Line number in file	Sequential file line number
	Number of transactions	Number(10)	N/A	Number of lines between FHEAD and FTAIL

Design Assumptions

N/A

Upload Replenishment Data (replindbatch.ksh)

Module Name	replindbatch.ksh
Description	Upload replenishment schedule
Functional Area	Inventory Movement
Module Type	Integration

Module Technology	Ksh
Catalog ID	RMS475
Wrapper Script	rmswrap_shell_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program is used to Bulk upload xml file data from template files to a staging table (into the content XML column).

This batch will be responsible for validating the input parameters, below are the list of validations.

- The input file should exist.
- The input file's extension must be ".xml".
- The template_name should be valid. A package function will be called for validation.

Once xml data is loaded into the staging table, the script will do the following:

- Initialize a row in the process tracker table for asynchronous processing.
- Call the main induction process that uploads data into the staging tables, validates and inserts data into the base Merchandising replenishment schedule tables.

Note

The base templates used by this batch are loaded through a script on provisioning (REPLENISHMENT_DATA). Additional templates can be configured using the Data Loading Template Configuration in the Merchandising task list under Application Administration for type Replenishment.

Restart/Recovery

N/A

Design Assumptions

N/A

Import Management

When using the Import Management features in Merchandising, there are several inbound integration processes that are available for harmonized tariff schedules (HTS), transportation, and letter of credit functions. If you are using Simplified Import Management (based on your system options configurations), then only the HTS upload is supported.

For additional information about import management, including detailed flow diagrams, see the *RTM Overview* white paper in the Merchandising Documentation Library (Doc ID: 1585843.1).

The following integrations are included in this section:

- [Harmonized Tariff Schedule Upload \(htsupld\)](#)
- [Letter of Credit Confirmation Upload \(lcupld\)](#)
 - [SWIFT File Conversion - Letter of Credit Confirmation \(lcmt730\)](#)
- [Letter of Credit Drawdowns and Charges Upload \(lcup798\)](#)
 - [SWIFT File Conversion - Letter of Credit Drawdowns and Charges \(lcmt798\)](#)
- [Transportation Upload \(tranupld\)](#)

Harmonized Tariff Schedule Upload (htsupld)

Module Name	htsupld.pc
Description	Harmonized Tariff Schedule Upload
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS41
Wrapper Script	rmswrap_multi_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The harmonized tariff schedule batch module processes a file containing the most recent tariff schedule, by country of import, into Merchandising tables. The module uploads both the initial entry of the schedule and all the updates, as they become available. Once the HTS definitions have been updated in Merchandising, if the system settings indicate that updates of items and/or purchase orders should be performed, the program will update items and purchase orders if definitions of the HTS classifications that are associated with the items or orders were updated.

Restart/Recovery

Recommended commit counter is 2000. Input file names must end in a ".1" for the restart mechanism to properly parse the file name. Because there is only 1 input file to be uploaded, only 1 thread is used.

A reject file is used to hold records that have failed processing. You can fix the rejected records and process the reject file again.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000051

Input File Layout

Table 6-40 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record Descriptor	Char(5)	FHEAD	Describes file line type
	Line number	Number(10)	0000000001	Sequential file line number
	File ID	Char(5)	HTSUP	Describes file type
THEAD	Record Descriptor	Char(5)	THEAD	Describes file line type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction id	Number(14)	N/A	Unique transaction id

Table 6-40 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	HTS Line	Char(418)	N/A	The HTS Line is broken down into subsections, V1, V2, V3, and V4 or VL. These subsections are concatenated together. The V1 and V2 subsections are required, but the V3, V4 and VL subsections are optional. The data in the V4 and VL subsections is not used by Merchandising/Import Management, if provided, the data is ignored. For US imports, special programs are provided in the US HTS file in sections V3 and VD. The special program codes in both of these records should be concatenated together in the Oracle Retail HTS Upload Input File's THEAD HTS Line's V3 subsection. See HTS Line breakdown below for details
TDETL	Record Descriptor	Char(5)	TDETL	Describes file line type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction id	Number(14)	N/A	Unique transaction id

Table 6-40 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Tax/fee line	Char(95)	N/A	The Tax/Fee Line is broken down into subsections V5, V6, V7, V8, V9, VA, VB, and VC. Each subsection is optional and when provided, is expected to be on a separate TDETL line. If there is no data for a subsection, the entire TDETL record will not be present in the file, which means for some THEAD records there may not be any TDETL lines.
TTAIL	Record Descriptor	Char(5)	TTAIL	Describes file line type
	Line number	Number(10)	N/A	Sequential file line number
	Detail lines	Number(6)	N/A	Number of lines between THEAD and TTAIL
FTAIL	Record Descriptor	Char(5)	FTAIL	Describes file line type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction Lines	Number(10)	N/A	Number of lines between FHEAD and FTAIL

Input File THEAD – HTS Line Subsections

For each tariff (also referred to as an HTS classification), the HTS Line in THEAD is broken into concatenated subsections: V1, V2, V3, and V4 or VL. Subsections V1 and V2 are mandatory. Subsections V3, V4 and VL are optional, which means the entire subsection can be blank.

The inclusion of a V4 or VL subsection in the THEAD record will not cause any errors in the program. However, any data provided in subsections V4 or VL is not currently used in Merchandising/Import Management. Because neither is utilized by Merchandising/Import Management, they should be omitted from the input file completely.

If one or more special programs (tariff treatments) are associated with an HTS in the V3 subsection, and no corresponding V5-VC records are provided, the rates of the special program are assumed to be zero (that is, duty free).

Record Name	Field Name	Field Type	Default Value	Description
V1a	Control identifier	Char(1)	V	Identifies start of record
	Record type	Char(1)	1	Identifies record type
	Tariff number	Number(25)		Harmonized Tariff Schedule (HTS) code or tariff number used to classify merchandise for import. If this number is less than 25 positions, it is left justified.
	Transaction code	Char(1)	A, D, R	A code representing the type of transaction. Valid Transaction Codes are: A = Add D = Delete R = Replace
	Begin effective date	char(6)		A numeric date in MMDDYY (month, day, year) format representing the record begin effective date. This date indicates when the record becomes effective.
	End effective date	char(6)		A numeric date in MMDDYY (month, day, year) format representing the record end effective date. This date indicates the last date the record is effective.
	number of reporting units	number(1)	0,1,or 2 or 3	The number of reporting units required by the Bureau of the Census. In a few instances, units not required by Census may be required to compute duty. In these cases, the Census reporting units are always first, followed by any additional units required to compute the duty.

Record Name	Field Name	Field Type	Default Value	Description
h	1 st reporting unit of measure	char(4)		A code representing the first unit of measure. Valid unit of measure codes are found on the Units of Measure Class table (uom_class) in Merchandising. If this number is less than 4 positions, it is left justified.
i	2 nd reporting unit of measure	char(4)		A code representing the second unit of measure. Valid unit of measure codes are found on the Units of Measure Class table (uom_class) in Merchandising. If this number is less than 4 positions, it is left justified.
j	3 rd reporting unit of measure	char(4)		A code representing the third unit of measure. Valid unit of measure codes are found on the Units of Measure Class table (uom_class) in Merchandising. If this number is less than 4 positions, it is left justified.
k	duty computation code	char(1)		A code indicating the formula to be used to compute the duty. Valid Duty Computation Codes are found under the Duty Computation Codes (DCMP) code type.
l	commodity description	char(30)		A condensed version of the commodity description that appears in the HTS.
m	column 1 specific rate of duty	Number(12)		The specific rate of duty (monetary amount per unit of measure) applied for imports in general when no conditional tariff treatments are applicable. Within Merchandising this rate is stored against the Column 1 (C1) tariff treatment. Eight decimal places are implied.

Record Name	Field Name	Field Type	Default Value	Description
n	base rate indicator	char(1)	'B' or blank	A code indicating if the rate contains a base rate. If the base rate indicator is <i>B</i> , the duty rate is a base rate; otherwise, space fill. Not Used in RMS.
o	space fill	char(1)	blank	Space fill.
V2	Control identifier	char(1)	V	Identifies start of record
a				
b	Record type	char(1)	2	Identifies record type
c	tariff number	Number (25)		Harmonized Tariff Schedule (HTS) code or tariff number used to classify merchandise for import. If this number is less than 25 positions, it is left justified. This number is the same as that in Record Identifier V1.
d	general column 1 ad valorem percentage	Number (12)		The ad valorem rate of duty applied for imports in general when no conditional tariff treatments are applicable. Within Merchandising this rate is stored against the Column 1 (C1) tariff treatment. Eight decimal places are implied.
e	column 1 other	Number (12)		The other rate of duty applied for imports in general when no conditional tariff treatments are applicable. Within Merchandising this rate is stored against the Column 1 (C1) tariff treatment. Eight decimal places are implied.

Record Name	Field Name	Field Type	Default Value	Description
f	Column 2 specific rate	Num(12)		The specific rate of duty (monetary amount per unit of measure) applied for imports in general when the country of origin is not in good standing with the country of import and higher rates of duty are assessed to deter trade. Within Merchandising this rate is stored against the Column 2 (C2) tariff treatment. Eight decimal places are implied.
g	Column 2 ad valorem percentage	Num(12)		The ad valorem rate of duty applied for imports in general when the country of origin is not in good standing with the country of import and higher rates of duty are assessed to deter trade. Within Merchandising this rate is stored against the Column 2 (C2) tariff treatment. Eight decimal places are implied.
h	Column 2 other rate	Num(12)		The other rate of duty applied for imports in general when the country of origin is not in good standing with the country of import and higher rates of duty are assessed to deter trade. Within Merchandising this rate is stored against the Column 2 (C2) tariff treatment. Eight decimal places are implied.
i	countervailing duty flag	char(1)	blank or 1	A code of 1 indicating the tariff number is subject to countervailing duty; otherwise, space fill.
j	additional tariff indicator	char(1)	blank or 'R'	A code indicating if an additional HTS code or tariff number may be required to fully classify the item. This indicator is R when an additional tariff number may be required; otherwise, space fill.

Record Name	Field Name	Field Type	Default Value	Description
k	Miscellaneous Permit/ License Indicator	char(2)		A code indicating if a tariff number may be subject to a miscellaneous permit/ license number.
l	space fill	char(4)	blanks	Blank space fill.
V3	Control identifier	char(1)	V	identifies start of record
a				
b	Record type	char(1)	3	identifies record type
c	tariff number	Number(25)		Harmonized Tariff Schedule (HTS) code or tariff number used to classify merchandise for import. If this number is less than 25 positions, it is left justified. This number is the same as that in Record Identifier V1.
d	GSP excluded countries	char(20)		The International Organization for Standardization (ISO) country code that indicates countries not eligible for preferential treatment under GSP. Valid country codes are found on the country table (country). Up to 10 2-position country codes.
e	OGA codes	char(15)		Codes that indicate special requirements by Other Government Agencies (OGA) may apply. Up to five 3 position OGA codes can be provided.
f	anti-dumping flag	char(1)	1 or blank	A code of 1 indicating the tariff number is subject to an antidumping duty; otherwise, space fill.
g	quota indicator	char(1)	1 or blank	A code of 1 indicating the tariff number may be subject to quota. If the tariff number is not subject to quota, space fill.

Record Name	Field Name	Field Type	Default Value	Description
h	category number	char(6)		A code located in the HTS indicating the textile category assigned to the tariff number. If there is no textile category number, space fill.
I	special program indicators Special Program Indicator (SPI) / Tariff Treatment	char(60)		Special Program Indicator or Tariff Treatment codes that indicate if a tariff number is subject to a special program with preferential rates of duty. Up to thirty 2 position codes can be reported. Left justify, meaning if the SPI code is 1-position, space fill position 2. The tariff treatment codes are not reported in any sequence. For US imports, special programs are provided in the US HTS file in sections V3 and VD. The special program codes in both of these records should be concatenated together in the Oracle Retail HTS Upload Input File's THEAD HTS Line's V3 subsection.
V4 a	Control identifier	char(1)	V	Identifies start of record. The entire V4 record not used in Merchandising/Import Management. If provided, it will be ignored.
b	Record type	char(1)	4	Identifies record type
c	tariff number	Number (25)		Harmonized Tariff Schedule (HTS) code or tariff number used to classify merchandise for import. If this number is less than 25 positions, it is left justified. This number is the same as that in Record Identifier V1.
d	value edit code	char(3)		A code representing the value edit.

Record Name	Field Name	Field Type	Default Value	Description
e	value low bounds	Number (10)		A value representing the minimum value edit. Five decimal places are implied. If this record contains date edits this field will contain, space fill.
f	value high bounds	Number (10)		A value representing the maximum value edit. Five decimal places are implied. If this record contains date edits this field will contain, space fill.
g	entry date restriction	Number (1)	0,1, or 2	A code representing the first entry date restriction code.
h	beginning restriction date	char(4)		A numeric date in MMDD (month and day) format representing the first begin restriction date used in the edit. If this record contains a value edit this field will contain, space fill.
l	end restriction date	char(4)		A numeric date in MMDD (month and day) format representing the first end restriction date used in the edit. If this record contains a value edit this field will contain, space fill.
j	entry date restriction 2	number(1)	0,1, or 2	A code representing the second entry date restriction code.
k	beginning restriction date 2	char(4)		A numeric date in MMDD (month and day) format representing the second begin restriction date used in the edit. If this record contains a value edit this field will contain, space fill.
l	end restriction date 2	char(4)		A numeric date in MMDD (month and day) format representing the second end restriction date used in the edit. If this record contains a value edit this field will contain, space fill.

Record Name	Field Name	Field Type	Default Value	Description
m	country of origin	char(2)		A code representing the value edit.
n	space filler	char(2)	blanks	A value representing the minimum value edit. Five decimal places are implied. If this record contains date edits this field will contain, space fill.
o	quantity edit code	char(3)		A value representing the maximum value edit. Five decimal places are implied. If this record contains date edits this field will contain, space fill.
p	low quantity	Number (10)		A code representing the first entry date restriction code.
q	high quantity	Number (10)		A numeric date in MMDD (month and day) format representing the first begin restriction date used in the edit. If this record contains a value edit this field will contain, space fill.
VL a	Control identifier	char(1)	V	Identifies start of record. The entire VL record not used in Merchandising/Import Management. If provided, it will be ignored.
b	Record type	char(1)	L	Identifies the record type
c	tariff number	Number (25)		Harmonized Tariff Schedule (HTS) code or tariff number used to classify merchandise for import. If this number is less than 25 positions, it is left justified. This number is the same as that in Record Identifier V1.

Record Name	Field Name	Field Type	Default Value	Description
d	Participating Government Agency (PGA) Codes	char(60)		Codes that indicate special requirements by participating government agencies (PGAs) must or may apply. Up to 20 3-position PGA codes can be provided. PGAs are not supported in MFCS at this time. This data is ignored by the system.
e	space fill	char(8)	blanks	blank space fill

Input File TDETL – Tax/Fee Line Subsections

The Tax/Fee Line in TDETL contains tax/fee information and rates for special programs (tariff treatments), and all have the same structure. The Tax/Fee Line is broken into optional subsections V5-V9, VA, VB, and VC. Each of these subsections is in its own TDETL line.

Record Name	Field Name	Field Type	Default Value	Description
V5	Control identifier	char(1)	V	Identifies start of record
a				
b	Record type	char(1)	5,6,7,8,9,A,B,C	Identifies record type
c	tariff number	Number (25)		Harmonized Tariff Schedule (HTS) code or tariff number used to classify merchandise for import. If this number is less than 25 positions, it is left justified. This number is the same as that in Record Identifier V1.
d	Country code Special Program Indicator (SPI) / Tariff Treatment / Country Code	char(2)		A code representing a tariff treatment/special program indicator which may or may not be the same as a valid ISO country code. Single character tariff treatment codes should include a space after the code to fill the 2-character space per code.

Record Name	Field Name	Field Type	Default Value	Description
e	specific rate	Number (12)		The specific rate of duty listed in the Special column of the HTS. Eight decimal places are implied.
f	ad valorem rate	Number (12)		The ad valorem rate of duty listed in the Special column of the HTS. Eight decimal places are implied.
g	Other rate	Number (12)		The rate of duty listed in the Special column of the HTS that is not a specific or ad valorem rate. Eight decimal places are implied.
h	tax/fee class code	char(3)		A code representing the tax/fee class or type. The system assumes that any class/type code which is less than 23 (i.e. 016, 017, 018, 022, etc.) is a tax, and any value greater than or equal to 23 is a fee (i.e. 023, 024, 045, 056, etc.).
i	tax/fee computation code	char(1)		A code indicating the first tax/fee computation formula. Valid Computation formulas for taxes and fees are found under the Duty Computation Codes (DCMP) code type, but computation codes 0, J and K are only valid for duty, not for tax and fee computations.

Record Name	Field Name	Field Type	Default Value	Description
j	tax/fee flag	number(1)		A code indicating a tax/fee is required. Valid Tax/Fee Flag Codes are: 1 = Tax/fee required 2 = Tax/fee may be required. Not used in Merchandising/Import Management.
k	tax/fee specific rate	Number (12)	blank if no value	The specific rate of duty required to compute taxes and/or fees. Eight decimal places are implied. If this field is filled with 0's when no tax/fee class code, computation code, or flag are provided, HTS upload will ignore the rates. Meaning a rate filled with 0's is the same as leaving the field blank.
l	tax/fee ad valorem	Number (12)	blank if no value, or filled with 0's	The ad valorem rate of duty required to compute taxes and/or fees. Eight decimal places are implied. If this field is filled with 0's when no tax/fee class code, computation code, or flag are provided, HTS upload will ignore the rates. Meaning a rate filled with 0's is the same as leaving the field blank.
m	space fill	char(1)	blank	Space fill.
Note: V6, V7, V8, V9, VA, VB, and VC subsections have the same fields as the V5 subsection described above.				

Note

The HTS Line in THEAD and the Tax/Fee Line in TDETL have subsections that start with a two-character code such as V1, V3, VA, and so on, which correspond to records provided by US Customs and Border Protection in the electronic file publication of the US Harmonized Tariff Schedule. The VD record in the US HTS file is used as an overflow for the V3 record in the event that the HTS tariff has more than 14 special programs (or tariff treatments). The Oracle Retail HTS Upload input file format supports additional special program codes in the V3 subsection of the THEAD HTS Line that is long enough to accept codes from both of the US HTS file's V3 and VD records.

Design Assumptions

N/A

Letter of Credit Confirmation Upload (lcupld)

Module Name	lcupld.pc
Description	Letter of Credit Confirmation Upload
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS55
Wrapper Script	Rmswrap_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The LCUPLD program is used to upload LC (Letter of Credit) confirmations from bank partners.

After this program has processed a confirmation, the appropriate tables will be updated; a confirmation will update the LC to confirm status and it will write the appropriate records to the LC_ACTIVITY table.

Restart/Recovery

Restart/recovery for this program is set up at the individual FDETL record. Although there may be more than one FDETL record for a given LC, they will each be processed as a separate entity.

File based restart/recovery must be used. The commit_max_ctr field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 10000 records.

I/O Specification

Integration Type	Upload to Merchandising

File Name Determined by runtime parameter
Integratin Contract IntCon000054

Input File Layout

Table 6-41 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Sequence Number	Number(10)	0000000001	Line number of the current file
	File Type Definition	Char(4)	LCUP	Identifies file as 'Letter of Credit Upload'
	File Create Date	Char (14)	vdate	Date file was written by external system 'YYYYMMDDHH24MISS' format
File Detail	File Type Record Descriptor	Char(5)	FDETL	Identifies file record type
	File Line Sequence Number	Number(10)		Line number of the current file
	Sender's Reference	Char(16)	lc_head.bank_l c_id	The LC number that the bank assigns to a Letter of Credit
	Receiver's Reference	Number(8)	lc_activity.lc_ref _id	The LC number that Trade Management assigned to the Letter of Credit
	Date of Message Being Acknowledged	Char(14)	lc_activity.activit y_date	YYYYMMDDHH24MISS format
	Comments	Char(2000)	lc_activity.com ments	This field is a concatenation of the following SWIFT fields: 71B – Charges, 72 – Sender information
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence	Number(10)	N/A	Line number of the current file

Table 6-41 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Total number lines	Number(10)	N/A	Total number of lines in file not including FHEAD and FTAIL

Letter of Credit Drawdowns and Charges Upload (lcup798)

Module Name	lcup798.pc
Description	Letter of Credit Drawdowns and Charges
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS54
Wrapper Script	rmswrap_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program reads data from an input file containing letter of credit charges and drawings (in standard Oracle Retail format, modified from the SWIFT 798 format by the lcmt798 Perl script), validates it, and inserts it into the LC_ACTIVITY table. If a record fails validation, it will be written to a reject file. These rejected records can be reprocessed by lcup798 after errors have been corrected.

Restart/Recovery

This program will be restartable but not threadable.

Restart/recovery logic for file-based processing is used. Records will be committed to the database when commit_max_ctr defined in the RESTART_CONTROL table is reached.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000055

The input file for this batch program is the output from the lcmt798 Perl script.

Input File Layout

Table 6-42 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File head descriptor	Char(5)	FHEAD	Describes file line type
	Line id	Number (10)	0000000001	Sequential file line number
	File Type Definition	Char(4)	'LCCH'	Identifies as an LC 798 file-Letter of Credit Charges
	Current date	Date	N/A	File date in YYYYMMDDHH24MISS format
FDETL	File record descriptor	Char(5)	FDETL	Describes file line type
	Line id	Number (10)		Sequential file line number
	Bank letter of credit reference ID	Char (16)	SWIFT tag 20	Bank's LC ref ID
	Order number	Number(8)	SWIFT tag 21	Order number attached to LC.May be blank
	Invoice number	Number (15)	SWIFT tag 23	NOT a Merchandising invoice number, just a reference invoice number from the issuing bank. May be blank
	Transaction number	Number (10)	N/A	Amendment number or transaction number assigned by bank.May be null
	Transaction code	Char(6)	B or D	'B'ank charge or'D'rawdown
	Amount	Number(21)	SWIFT tag 33A,71A	(This is a 20-digit number with a leading – sign or blank and 4 implied decimal places.) Amount of charge or drawdown
	Currency code	Char(3)	SWIFT 33A,71A	Currency that the amount is in

Table 6-42 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	Activity date	Date	SWIFT 33A,32C,32D	Activity date(formatted as 'YYYYMMDD')
	Comments	Char(2000)	SWIFT tag 72	Any comments associated with activity.May be null
	File record descriptor	Char(5)	FTAIL	Marks end of file
	Line id	Char(10)	N/A	Sequential file line number
	Number of lines	Number(10)	N/A	Number of lines in file not counting FHEAD and FTAIL

SWIFT File Conversion - Letter of Credit Confirmation (lcmt730)

Module Name	lcmt730
Description	SWIFT File Conversion – Letter of Credit Confirmation
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	Perl
Catalog ID	RMS138
Wrapper Script	batch_lcmt730.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The lcmt730 Perl script converts letter of credit confirmations from a S.W.I.F.T. format (MT730) to a Merchandising flat file format. The output file from this script will be the input file for the lcupld.pc.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000054 (output) IntCon000139 (input)

Input File Layout

Table 6-43 Input File Layout

SWIFT I.D. and Description	Data Type	Description	How MT 730 fields are put into the Merchandising standard file format and what should be the size of Merchandising to be dealt with	Comments
20 - Sender's Reference	16x	LC number. The one assigned by the Sender (issuing bank)	FDETL - Sender's reference, Char(16)	This field maps to Trade Management's Bank LC Ref ID.
21 - Receiver's Reference	16x	LC number assigned by the Receiver (retailer)	FDETL - Receiver's reference, Number(8) (NOREF used if unknown)	This field maps to Trade Management's LC Ref ID. If this field has 'NOREF', the record must be rejected since this field is used to indicate the LC within Trade Management to which this record applies.
25 - Account Identification	35x	Identifies the number of the account, which has been used for the settlement of charges, on the books of the Sender.	N/A	Trade Management currently does not have fields that map directly to this. Current position - will be included in the input file. However, it will be ignored during the upload process.

Table 6-43 (Cont.) Input File Layout

SWIFT I.D. and Description	Data Type	Description	How MT 730 fields are put into the Merchandising standard file format and what should be the size of Merchandising to be dealt with	Comments
30 - Date of Message Being Acknowledged	6!n	When a message is acknowledging a MT700, this field specifies the date of issue. In all other cases, this field specifies the date on which the message being acknowledged was sent.	FDETL - Date of message Being Acknowledged, Date	This field maps to the LC activity date. As well, if this in confirming an LC application, it will be mapped to the LC's confirmation date. Year interpretation: If YY>79 then YYMMDD = 19YYMMDD Else YYMMDD = 20YYMMDD.
32a - Amount of Charges	Option B - 3!a15d Option D - 6!n3!a15d	Contains the currency code and total amount of charges claimed by the sender of the message. When charges have been debited, D is used (:32D) and when reimbursement for charges is needed, B is used (:32B).	FDETL -Upload_type = 'C'onfirmation	Current position - Because the 730 will only be used for confirmations, this field will not contain any values. The upload type should be set equal to 'C'onfirmation.

Table 6-43 (Cont.) Input File Layout

SWIFT I.D. and Description	Data Type	Description	How MT 730 fields are put into the Merchandising standard file format and what should be the size of Merchandising to be dealt with	Comments
57a - Account With Bank	Option A - [/1!a]/[34x] 4!a2!a2!c[3!c] Option D - [/1!a]/[34x] 4*35x	This field specifies the bank to which the amount of charges is to be remitted in favor of the Sender.	FDETL - Account With Bank, Char(10)	Current position - will be added to the input file however will be ignored in the upload process. Because Trade Management has no facilities to maintain BICs or party identifiers, option D will always be used for this field (that is, 57D) without [/1!a]/[34x] party identifier.
71B - Charges	6*35x	Specification of the charges claimed.	FDETL - Comments, Char(2000)	This field maps to Trade Management's activity comments field. Sender to Receiver information (72) will be concatenated to this.
72 - Sender to Receiver Information	6*35x	Text explanation if wanted.	FDETL - Comments, Char(2000)	This field maps to Trade Management's activity comments field. Charges (71B) will be concatenated to this.

Output File Layout

Table 6-44 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Sequence Number	Number(10)	specified by external system	Line number of the current file
	File Type Definition	Char(4)	LCUP	Identifies file as 'Letter of Credit Upload'
	File Create Date	Char (14)	vdate	date file was written by external system 'YYYYMMDD HH24MISS' format
File Detail	File Type Record Descriptor	Char(5)	FDETL	Identifies file record type
	File Line Sequence Number	Number(10)	specified by external system	Line number of the current file
	Sender's Reference	Char(16)	lc_head.bank_id_id	The LC number that the bank assigns to a Letter of Credit
	Receiver's Reference	Number(8)	lc_activity.lc_ref_id	The LC number that Merchandising assigned to the Letter of Credit
	Date of Message Being Acknowledged	Date (char 8)	lc_activity.activity_date	If the upload type is 'L' then this date will match the date MT 700 date of issue (which we have not resolved between being the vdate or the lc_head.application_date) 'YYYYMMDD' format

Table 6-44 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Comments	Char(2000)	lc_activity.com ments	Need to truncate? This field will probably be a concatenation of the following SWIFT fields: 71B – Charges, 72 – Sender information
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Sequence	Number(10)	Specified by external system	Line number of the current file
	Total number of lines	Number(10)	Specified by external system	Total number lines in file

Design Assumptions

N/A

SWIFT File Conversion - Letter of Credit Drawdowns and Charges (lcmt798)

Module Name	lcmt798
Description	SWIFT File Conversion – Letter of Credit Drawdowns and Charges
Functional Area	Retail Trade Management - Letter of Credit Interfaces
Module Type	Integration
Module Technology	Perl
Catalog ID	RMS139
Wrapper Script	batch_lcmt798.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This Perl script converts letter of credit (L/C) activity data for charges and drawdowns from a S.W.I.F.T. format input file to a Merchandising format file.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000139 (input)

Input File Layout

Table 6-45 Input File Layout

Swift Tag	Description	Regd?	Datatype	Merchandising Field
20 - Transaction Reference Number	The sender's unambiguous identification of the transaction. Its detailed form and content are at the discretion of the sender.	Yes	16x - Transaction Reference Number	Bank L/C ID Lc_head.bank_lc_id Varchar2(16)
12 - Type of Financial Instrument	This field classifies the financial instrument by a description or proprietary code.	Yes	Option A- :4!c/[8c]/30x :4!c - Qualifier / - Delimiter [8c] - Issuer Code / - Delimiter 30x - Type	This field will contain a constant identifier - '798'
77E - Proprietary Message	This field contains the proprietary message in a format agreed to by the Sender and the Receiver.	Yes	Option E- 73x [n*78x]	This field will contain the information below (fields 21, 23, 32C, 32D, 71A, 33A, 72) Carriage return, Line feed, Colon 'CrLf:' will be used to separate fields included in this 77E For example: :77E:'CrLf' :21:10004321:CrLf' :32C:990121USD1045 and so on. There may be multiple 77Es in one file

Table 6-45 (Cont.) Input File Layout

Swift Tag	Description	Regd?	Datatype	Merchandising Field
21 - Related Reference	This field specifies, in an unambiguous way, a message or transaction identifier which is normally included as part of the information supplied with the message or transaction itself, and can subsequently be used to distinguish the message or transaction identified from other messages or transactions.	No	16x	P/O Number Lc_activity.order_no Number(8)
23 - Further identification	This field specifies the type of transaction being confirmed, as well as the settlement method used.	No	16x	Invoice Number Lc_activity.invoice_no Varchar2(15)
32C - Date and Amount	This field specifies the currency code and amount in a transaction and a corresponding date.	No	Option A- :4!c/[8c]/30x :4!c - Qualifier / - Delimiter [8c] - Issuer Code / - Delimiter 30x - Type	Charges Credited (this is interpreted as a positive amount) Date will be in format YYMMDD The integer part of the Amount must contain at least one digit. A decimal comma ',' is mandatory and is included in the maximum length Lc_activity.amount Number(20,4) Lc_activity.currency_code Varchar2(3) Lc_activity.activity_date Date

Table 6-45 (Cont.) Input File Layout

Swift Tag	Description	Regd?	Datatype	Merchandising Field
32D - Date and Amount	This field specifies the currency code and amount in a transaction and a corresponding date.	No	Option D- 6!n3!a15d 6!n - Date 3!a - Currency 15d - Amount	Charges Debited (this is interpreted as a negative amount) Date will be in format YYMMDD The integer part of the Amount must contain at least one digit. A decimal comma ',' is mandatory and is included in the maximum length Lc_activity.amount Number(20,4) Lc_activity.currency_code Varchar2(3) Lc_activity.activity_date Date
33A - Date and Amount	This field specifies the currency code and amount in a transaction and a corresponding date.	No	Option A- 6!n3!a15d 6!n - Date 3!a - Currency 15d - Amoun	Date, currency, amount of drawing (this is interpreted as a positive amount) Date will be in format YYMMDD The integer part of the Amount must contain at least one digit. A decimal comma ',' is mandatory and is included in the maximum length Lc_activity.amount Number(20,4) Lc_activity.currency_code Varchar2(3) Lc_activity.activity_date Date

Table 6-45 (Cont.) Input File Layout

Swift Tag	Description	Regd?	Datatype	Merchandising Field
33C - Date and Amount	This field specifies the currency code and amount in a transaction and a corresponding date.	No	Option A- 6!n3!a15d 6!n - Date 3!a - Currency 15d - Amount	Date, currency, amount of drawing (this is interpreted as a negative amount) Date will be in format YYMMDD The integer part of the Amount must contain at least one digit. A decimal comma ',' is mandatory and is included in the maximum length. Lc_activity.amount Number(20,4) Lc_activity.currency_code Varchar2(3) Lc_activity.activity_date Date
72 - Sender to Receiver Information	This field specifies instructions or additional information for the Receiver, Intermediary, Account with Institution or Beneficiary Institution.	No	6*35x	Comments Lc_activity.comment Varchar2(2000)
18A - Number of Repetitive Parts	This field specifies the number of times the repetitive part(s)/ sequence(s) directly before or after this field appears in the message.	No	Option A- 5n - Number of Repetitive Parts.	Number of 77E's contained within the file.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000055 (input)

Output File Layout

Table 6-46 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Identifier	Number (10)	Line number in file	ID of current line being created for output file
	File Type Definition	Char(4)	LCCH	Identifies file as 'Letter of Credit Changes'
	File Create Date	Char(14)	Create date	Current date, formatted to 'YYYYMMDDHH24MISS'
File Detail	File Type Record Descriptor	Char(5)	FDETL	Identifies file record type
	File Line Sequence Number	Number (10)	Line number in file	ID of current line being created for output file
	Bank Letter of Credit Reference ID	Char(16)	SWIFT tag 20	Bank L/C ID
	Order Number	Number (8)	SWIFT tag 21	Contains the order number that is attached to the letter of credit
	Invoice Number	Char (15)	SWIFT tag 23	Identifies the Issuing Bank's invoice number to which the drawdown refers. This field does not correspond to a Merchandising invoice number
	Transaction Number	Char (10)	Null	Identifies the amendment number or actual transaction number assigned by the bank
	Transaction Code	Char (6)	If the transaction is a Bank Charge – 'B' If the transaction is a Drawdown – 'D'	Identifies the type of transaction that occurred The type is determined by what detail fields are received for the record. If the record contains a 33A this field will get a 'D'. If the record contains either a 32C or 32D this field will get a 'B'

Table 6-46 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Amount Sign	Char (1)	SWIFT 33A, 33C SWIFT 32C, 32D	If the record contains a 33A field leave a blank space in this field If the record contains a 33C field this field should contain a '-' If the record contains a 32C field leave a blank space in this field If the record contains a 32D field this field should contain a '-'
	Amount	Number (20)	SWIFT 33A, 33C SWIFT 32C, 32D	Holds the amount of the activity. This field will have 4 implied decimal places If SWIFT 32C or 32D (Bank Charge) contains a value, use the amount from this field If SWIFT 33A or 33C (Drawdown) contains a value, use the amount from this field
	Currency Code	Char (3)	SWIFT 33A, SWIFT 32C, 32D	Contains the activity's currency code If SWIFT 32C or 32D (Bank Charge) contains a value, use the currency from this field If SWIFT 33A (Drawdown) contains a value, use the currency from this field
	Activity Date	Char (8)	SWIFT 33A, SWIFT 32C, 32D	Holds the date that the activity took place. Formatted to 'YYYYMMDD' If SWIFT 32C or 32D (Bank Charge) contains a value, use the date from this field If SWIFT 33A (Drawdown) contains a value, use the date from this field
	Comments	Char (2000)	SWIFT tag 72	Holds any comments for the activity

Table 6-46 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Identifier	Number (10)	Sequential number Created by program.	ID of current line being created for output file
	File Record Counter	Number (10)	N/A	This will contain the number of FDETL lines processed

Transportation Upload (tranupld)

Module Name	tranupld.pc
Description	Transportation Upload
Functional Area	Oracle Retail Trade Management
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS140
Wrapper Script	rmswrap_multi_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This program uploads data from trading partners about the transportation of merchandise from the manufacturing site through customs clearance.

Restart/Recovery

The logical unit of work is a valid DTRAN record. The program reads each DTRAN record from the upload file, validates it and processes it. The recommended commit max counter value for this program is 1000 (this value depends on the implementation).

Backward Compatibility

Added additional parameter to indicate version. If no parameter is supplied, the version defaults to 1 (initial version).

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000177

Input File Layout Summary Across Versions

Version Number	Record Name	Length
1	File Header - FTRAN	33
	Transaction Header – DTRAN	257
	Transaction Detail – DPOIT	560
	Transaction Trailer – FTAIL	25
2	File Header - FTRAN	33
	Transaction Header – DTRAN	265
	Transaction Detail – DPOIT	560
	Transaction Trailer – FTAIL	25

Input File Layout

Table 6-47 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
FTRAN	Record descriptor	Char(5)	FTRAN	File head marker	
	Line id	Number(10)	0000000001	Unique line id	
	File type definition	Char(4)	TRUP	Identifies program as tranupld	
	File create date	Char(14)	Current date	YYYYMMDDHHMISS format	
DTRAN	Record descriptor	Char(5)	DTRAN	Vessel, Voyage, ETD, Container, BL, Invoice File head	
	Line id	Number(10)		Unique line id	
	Partner Type	Char(6)		Identifies the partner type	
	Partner ID	Char(10)		Identifies the partner id.	
	Vessel ID	Char(20)		Identifies the Vessel	
	Voyage ID	Char(10)		Identifies the Voyage or Flight ID	
	Estimated Depart Date	Char(8)		YYYYMMDD format	
	Shipment Number	Char (20)		Identifies an outside Shipment number	
	Actual Departure Date	Char(8)		YYYYMMDD format	2
	Actual Arrival Date	Char(8)		YYYYMMDD format	
	Trans Mode	Char(6)		Identifies the type of transportation being used. Valid values are found in the TRMO Code Type on the CODE_DETAIL table	

Table 6-47 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
DPOIT	Vessel SCAC Code	Char(6)		Customs defined ID for the Vessel. Validated against SCAC table.	
	Estimated Arrival Date	Char(8)		YYYYMMDD format	
	Lading Port	Char(5)		Identifies the Lading Port. Validated against OUTLOC with type = 'LP'	
	Discharge Port	Char(5)		Identifies the Discharge Port. Validated against OUTLOC with type = 'DP'	
	Service Contract Number	Char(15)		Identifies the outside Service Contract Number	
	Container id	Char(20)		Identifies the Container	
	Container SCAC code	Char(6)		Customs defined id for the container. Validated against SCAC table	
	Delivery Date	Char(8)		YYYYMMDD format	
	Seal id	Char(15)		Customs defined id for the container's seal	
	Freight Type	Char(6)		Code that identifies the container type. Validated against the FREIGHT_TYPE table.	
	Freight Size	Char(6)		Code that identifies the container size. Validated against the FREIGHT_SIZE table.	
	In Transit No.	Char(15)		External transit number	
	In Transit Date	Char(8)		YYYYMMDD format	
	BL/AWB id	Char(30)		Identifies the Bill of Lading or Air Way Bill	
	Candidate Ind	Char(1)	Defaulted to 'N'	Identifies a complete Transportation record. Valid values are 'Y' and 'N'	
	Record descriptor	Char(5)	DPOIT	Order/Item detail info	
DPOIT	Line id	Number(10)		Unique file line id	
	ACD_Code	Char(1)		Determines which process to perform 'A'dd, 'C'hange, 'D'elete.	
	Rush Ind	Char(1)	Defaulted to 'N'	Identifies whether or not the item should be on a 'Rush' delivery. Valid values are 'Y' and 'N'	

Table 6-47 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
	Order number	Number(12)		Merchandising order number	
	Item	Char(25)		Merchandising Item number	
	Invoice id	Char(30)		Identifies the Commercial Invoice	
	Invoice date	Char(8)		YYYYMMDD format	
	Currency Code	Char(3)		Currency that the Currency Amount is reported in. Validated against CURRENCIES table.	
	Exchange Rate	Char (20)		The exchange rate back to the primary currency (10 implied decimals)	
	Invoice amt	Char 20)		Invoice amt*10000 (with 4 implied decimal places), amount charged by supplier for the PO/Item	
	Origin Country id	Char(3)		Identifies where the PO/Item was made	
	Consolidation Country id	Char(3)		Identifies where the PO/Items were consolidated	
	Export Country id	Char(3)		Identifies where the PO/Items where shipped from	
	Status	Char(6)		Identifies the PO/Item status. Valid values are found in the TRCO Code Type on CODE_DETAIL	
	Receipt ID	Char(30)		Identifies the external receipt number	
	FCR id	Char(15)		Identifies the Freight Cargo Receipt id	
	FCR date	Char(8)		YYYYMMDD format	
	Packing Method	Char(6)		Identifies the Packing Type (Hanging or Flat). Valid values are 'HANG' or 'FLAT'	
	Lot Number	Char(15)		Identifies the Lot Number of the PO/Item	
	Item Qty	Number(12)		Item Qty*10000(with 4 implied decimals), qty of Items	
	Item QTY UOM	Char(4)		Identifies the UOM associated with the item quantity	

Table 6-47 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
FTAIL	Carton QTY	Number(12)		Carton QTY*10000 (with 4 implied decimals), qty of Cartons	
	Carton QTY UOM	Char(4)		Identifies the UOM associated with the carton quantity	
	Gross WT	Number(12)		Gross WT*10000 (with 4 implied decimals), Gross weight	
	Gross WT UOM	Char(4)		Identifies the UOM associated with the gross weight	
	Net WT	Number(12)		Net WT*10000 (with 4 implied decimals), Net Weight	
	Net WT UOM	Char(4)		Identifies the UOM associated with the net weight	
	Cubic	Number(12)		Cubic*10000 (with 4 implied decimals), cubic size	
	Cubic UOM	Char(4)		Identifies the UOM associated with the cubic size	
	Comments	Char(256)		User Comments	
	Record type	Char(5)	FTAIL		
	Line id	Number(10)		Unique file line id	
	No. of lines	Number(10)		Total number of transaction lines in file (not including FHEAD and FTAIL)	

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
FTRAN	Record descriptor	Char(5)	FTRAN	File head marker	
	Line id	Number(10)	0000000001	Unique line id	
	File type definition	Char(4)	TRUP	Identifies program as tranupld	
	File create date	Char(14)	Current date	YYYYMMDDH HMISS format	
DTRAN	Record descriptor	Char(5)	DTRAN	Vessel, Voyage, ETD, Container, BL, Invoice File head	

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
	Line id	Number(10)	N/A	Unique line id	
	Partner Type	Char(6)	N/A	Identifies the partner type	
	Partner ID	Char(10)	N/A	Identifies the partner id	
	Vessel ID	Char(20)	N/A	Identifies the Vessel	
	Voyage ID	Char(10)	N/A	Identifies the Voyage or Flight ID	
	Estimated Depart Date	Char(8)	N/A	YYYYMMDD format	
	Shipment Number	Char (20)	N/A	Identifies an outside Shipment number	
	Actual Arrival Date	Char(8)	N/A	YYYYMMDD format	
	Trans Mode	Char(6)	N/A	Identifies the type of transportation being used. Valid values are found in the TRMO Code Type on the CODE_DETAIL table	
	Vessel SCAC Code	Char(6)	N/A	Customs defined ID for the Vessel. Validated against SCAC table	
	Estimated Arrival Date	Char(8)	N/A	YYYYMMDD format	
	Lading Port	Char(5)	N/A	Identifies the Lading Port. Validated against OUTLOC with type = 'LP'	
	Discharge Port	Char(5)	N/A	Identifies the Discharge Port. Validated against OUTLOC with type = 'DP'	

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
DPOIT	Service Contract Number	Char(15)	N/A	Identifies the outside Service Contract Number	
	Container id	Char(20)	N/A	Identifies the Container	
	Container SCAC code	Char(6)	N/A	Customs defined id for the container. Validated against SCAC table	
	Delivery Date	Char(8)	N/A	YYYYMMDD format	
	Seal id	Char(15)	N/A	Customs defined id for the container's seal	
	Freight Type	Char(6)	N/A	Code that identifies the container type. Validated against the FREIGHT_TYPE table	
	Freight Size	Char(6)	N/A	Code that identifies the container size. Validated against the FREIGHT_SIZE table	
	In Transit No.	Char(15)	N/A	External transit number	
	In Transit Date	Char(8)	N/A	YYYYMMDD format	
	BL/AWB id	Char(30)	N/A	Identifies the Bill of Lading or Air Way Bill	
	Candidate Ind	Char(1)	Defaulted to 'N'	Identifies a complete Transportation record. Valid values are 'Y' and 'N'	
	Record descriptor	Char(5)	DPOIT	Order/Item detail info	
	Line id	Number(10)	N/A	Unique file line id	

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
	ACD_Code	Char(1)	N/A	Determines which process to perform 'A'dd, 'C'hange, 'D'elete.	
	Rush Ind	Char(1)	Defaulted to 'N'	Identifies whether or not the item should be on a 'Rush' delivery. Valid values are 'Y' and 'N'	
	Order number	Number(8)	N/A	Merchandising order number	
	Item	Char(25)	N/A	Merchandising Item number	
	Invoice id	Char(30)	N/A	Identifies the Commercial Invoice	
	Invoice date	Char(8)	N/A	YYYYMMDD format	
	Currency Code	Char(3)	N/A	Currency that the Currency Amount is reported in. Validated against CURRENCIES table.	
	Exchange Rate	Char (20)	N/A	The exchange rate back to the primary currency (10 implied decimals)	
	Invoice amt	Char (20)	N/A	Invoice amt*10000 (with 4 implied decimal places), amount charged by supplier for the PO/Item	
	Origin Country id	Char(3)	N/A	Identifies where the PO/Item was made	

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
	Consolidation Country id	Char(3)	N/A	Identifies where the PO/Items were consolidated	
	Export Country id	Char(3)	N/A	Identifies where the PO/Items where shipped from	
	Status	Char(6)	N/A	Identifies the PO/Item status. Valid values are found in the TRCO Code Type on CODE_DETAIL	
	Receipt ID	Char(30)	N/A	Identifies the external receipt number	
	FCR id	Char(15)	N/A	Identifies the Freight Cargo Receipt id	
	FCR date	Char(8)	N/A	YYYYMMDD format	
	Packing Method	Char(6)	N/A	Identifies the Packing Type (Hanging or Flat). Valid values are 'HANG' or 'FLAT'	
	Lot Number	Char(15)	N/A	Identifies the Lot Number of the PO/Item	
	Item Qty	Number(12)	N/A	Item Qty*10000(with 4 implied decimals), qty of Items	
	Item QTY UOM	Char(4)	N/A	Identifies the UOM associated with the item quantity	
	Carton QTY	Number(12)	N/A	Carton QTY*10000 (with 4 implied decimals), qty of Cartons	

Record Name	Field Name	Field Type	Default Value	Description	Added in Version
FTAIL	Carton QTY UOM	Char(4)	N/A	Identifies the UOM associated with the carton quantity	
	Gross WT	Number(12)	N/A	Gross WT*10000 (with 4 implied decimals), Gross weight	
	Gross WT UOM	Char(4)	N/A	Identifies the UOM associated with the gross weight	
	Net WT	Number(12)	N/A	Net WT*10000 (with 4 implied decimals), Net Weight	
	Net WT UOM	Char(4)	N/A	Identifies the UOM associated with the net weight	
	Cubic	Number(12)	N/A	Cubic*10000 (with 4 implied decimals), cubic size	
	Cubic UOM	Char(4)	N/A	Identifies the UOM associated with the cubic size	
	Comments	Char(256)	N/A	User Comments	
	Record type	Char(5)	FTAIL	N/A	
FTAIL	Line id	Number(10)	N/A	Unique file line id	
	No. of lines	Number(10)	N/A	Total number of transaction lines in file (not including FHEAD and FTAIL)	

Design Assumptions

N/A

Stock Counts

Merchandising subscribes to data related to stock counts from stores, warehouses, and third-party counters.

The following scheduled inbound integrations are included in this functional area:

- [Conversion of Warehouse Stock Count Results File \(lifstkup\)](#)
- [Upload Stock Count Results from Stores/Warehouses \(stockcountupload.ksh\)](#)

For more on stock count processing, see *Merchandising Operations Guide – Volume 1*.

Conversion of Warehouse Stock Count Results File (lifstkup)

Module Name	lifstkup.pc
Description	Conversion of WMS Stock Count Results File
Functional Area	Stock Counts
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS150
Wrapper Script	batch_lifstkup.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The Stock Upload Conversion batch is used when WMS sends count information to Merchandising. This batch converts the inventory balance upload file into the format supported by the Stock Count Upload process.

Restart/Recovery

Oracle Retail standard file-based restart/recovery is used. The commit max counter field should be set to prevent excessive rollback space usage, and to reduce the overhead of file I/O. The recommended commit counter setting is 1000 records (subject to change based on implementation).

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000172 (input from WMS) IntCon000102 (output for Merchandising stockcountupload)

Input File Layout

Table 6-48 Input File Layout

Field Name	Field Type	Description
DC_DEST_ID	11 – Number (10) + 1 for trailing space	Unique identifier for the warehouse
TRANSACTION_DATE	15 – Date (14) + 1 for trailing space	Date on which the transaction occurred
ITEM_ID	26 - Varchar2 (25) + 1 for trailing space	Uniquely identifies the item on the count
AVAILABLE_QTY	15 – Number (12) + 1 for leading sign and + 1 for decimal and + 1 for trailing space	Units available for distribution
DISTRIBUTED_QTY	14 – Number (12) + 1 for decimal and + 1 for trailing space	Units distributed include: Units distributed but not yet picked, units picked but not yet manifested, units manifested but not yet shipped
RECEIVED_QTY	15 - Number (12) + 1 for leading sign and + 1 for decimal and + 1 for trailing space	Units received but not put away
TOTAL_QTY	14 – Number (12,4) + 1 for decimal and + 1 for trailing space	Sum of all units that physically exist: container status of: I, D, M, R, T, X
AVAILABLE_WEIGHT	15 – Number (12,4) + 1 for leading sign + 1 for decimal + 1 for trailing space	Weight available for distribution of catch weight items
RECEIVED_WEIGHT	14 – Number (12,4) + 1 for decimal + 1 for trailing space	Weight received but not put away for catch weight items
DISTRIBUTED_WEIGHT	14 – Number (12,4) + 1 for decimal + 1 for trailing space	Weight distributed includes: weight distributed but not yet picked, weight picked but not yet manifested, weight manifested but not yet shipped (value only catch weight items)
TOTAL_WEIGHT	13 – Number (12,4) + 1 for decimal	Sum of all weight that physically exist: container status of: I, D, M, R, T, X. For catch weight items

Output File Layout

Table 6-49 Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	file type record descriptor	Char (5)	FHEAD	Describes the file line type
	file line identifier	Number (10)	0000000001	ID of current line being processed
	file type	Char (4)	'STKU'	Identifies the file type

Table 6-49 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FDETL	stocktake_date	Date (14)	N/A	The date on which the count occurred, formatted as YYYYMMDDHH24MISS
	file create date	Date (14)	N/A	Date on which the file was created, formatted as YYYYMMDDHH24MISS
	cycle count	Number (8)	N/A	stake_head.cycle_count
	Location type	Char (1)	'W'	Will always be 'W', as this process is only executed for warehouse locations
	location	Number(10)	N/A	Indicates the number of the physical warehouse where the count occurred
	file type record descriptor	Char(5)	FDETL	Identifies the file line type
	file line identifier	Number(10)	N/A	ID of current line being processed, internally incremented
	Item type	Char(3)	'ITM'	Indicates the type of item that was counted. This will always be 'ITM', indicating a transaction level item
	item value	Char(25)	N/A	The ID of the item that was counted
	inventory quantity	Number(12)	N/A	The total quantity or weight of product counted; includes four implied decimal places

Table 6-49 (Cont.) Output File Layout

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	location description	Char(150)	N/A	Used by Merchandising to determine the location where the item was counted. This program will always leave as NULL
	file type record descriptor	Char(5)	FTAIL	Identifies the file line type
	file line identifier	Number(10)	N/A	ID of current line being processed, internally incremented
	file record count	Number(10)	N/A	Indicates the number of detail records

Design Assumptions

N/A

Upload Stock Count Results from Stores/Warehouses (stockcountupload.ksh)

Module Name	stockcountupload.ksh
Description	Upload Stock Count Results from Stores/Warehouses
Functional Area	Stock Count
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS153
Wrapper Script	batch_stockcountupload.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this module is to upload the contents of the stock count file, which contains the results of a count that occurred in a store or warehouse, to staging tables for further processing.

Input/Out Specification

Integration Type	Upload in Merchandising
File Name	Determined by runtime parameter

Integratin Contract IntCon000102

Input File Layout

Table 6-50 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File head descriptor	Char(5)	FHEAD	Describes file line type
	File line identifier	Number(10)	0000000001	ID of current line being processed
	File Type	Char(4)	STKU	Identifies the file type
	File create date	Char(14)	N/A	Indicates the date the file was created in YYYYMMDDHH24MISS format
	Stock take date	Char(14)	N/A	Date on which stock count will take place in YYYYMMDDHH MISS format
	Cycle count	Number (8)	N/A	Unique number to identify the stock count
	Location Type	Char(1)	N/A	Indicates the type of location where the count occurred. Valid values are 'S','W','E'.
	Location	Number(10)	N/A	The location where the stock count occurred
Transaction Record	File record descriptor	Char(5)	FDETL	Describes file line type
	Line Number	Number(10)	N/A	Sequential file line number
	Item type	Char(3)	N/A	Indicates the type of item counted – either transaction level (ITM) or reference item (REF)
	Item value	Char(25)	N/A	Unique identifier for item that was counted

Table 6-50 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Inventory quantity	Number(12)	N/A	Total quantity counted for the item at the location formatted with 4 implied decimal places
	Location description	Char(150)	N/A	Description of inventory location (such as,. sales floor, backroom)
	Inventory Identifier Type	Char(6)		This field contains the inventory identifier type value being passed in physical stock count upload file.
	Inventory Id	Char(120)		This field contains the inventory ID value being passed in physical stock count upload file.
FTAIL	File record descriptor	Char(5)	FTAIL	Marks end of file
	File line identifier	Number(10)	N/A	ID of current line being processed, internally incremented
	File record count	Number(10)	N/A	Number of detail records

Design Assumptions

This program uses grep to search log files for errors. The GREP function should point to the /usr/xpg4/bin/ directory instead of /usr/bin directory to utilize the "-E" option. Otherwise, it will fail with an "illegal option" error message.

Franchise

Merchandising subscribes to data related to franchise customers, orders, and returns from order management solutions and other external franchise customer management solutions.

The following scheduled inbound integrations are included in this functional area:

- [Franchise Customer Upload \(fcustomerupload\)](#)
- [Franchise Order Upload \(wfordupld.ksh\)](#)

- [Franchise Return Upload \(wfretupld.ksh\)](#)
- [Upload Cost Buildup Template \(fcosttmplupld\)](#)
- [Upload of Franchise Sales \(wflsupld.ksh\)](#)

For more on franchise processing, see *Merchandising Operations Guide - Volume 1*.

Franchise Customer Upload (fcustomerupload)

Module Name	fcustomerupload.ksh
Description	Franchise Customers Upload
Functional Area	Franchise Management
Module Type	Integration
Module Technology	ksh
Integration Catalog ID	RMS126
Wrapper Script	rmswrap_shell_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This module uploads franchise customers and customer group details from an external system into Merchandising staging tables. It also performs both technical and business validation of the data sent in the file; for example, it validates that a customer cannot be deleted if a franchise store is associated with it.

Restart/Recovery

The restart recovery is different from the conventional Merchandising batch. There are three points on the batch upload process where you can evaluate the successful load of the data.

- **SQL load** - SQL load dumps invalid records that do not meet certain technical requirements (for example: data type inconsistencies, and so on.). The rejected record is written either to a bad file or to a discard file. The discard file contains records that do not satisfy conditions such as missing or invalid record types. Records with other technical issues are written to the bad file.

Note

A non-fatal code is returned by the program and a message will be written to the log file if reject files are created.

Action Required: When such conditions exist, you may update either the bad or discard file and attempt to reload using the same files.

- **File-Based Validations** - the data from the files are loaded into the staging tables for validation. PL/SQL functions will validate the data in the staging tables to determine if there are any issues with the FHEAD and FTAIL in the file. These kinds of errors are FATAL errors and the batch ends the file processing immediately with return code 255.

Action Required: When this condition exists, you can fix the data upload file and try to reload.

- **Business Validation Level - PL/SQL functions** determine if the transactions loaded are valid enough to modify the actual Merchandising tables. Records that do not meet certain technical or business validations are rejected and the information is updated back into the staging table with an appropriate error message and the batch issues a NON-FATAL return code 1.

Action Required: When this condition exists, you can fix the data upload file and try to reload.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000022

Input File Layout

Table 6-51 File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Record Descriptor	Char(5)	N/A	Identifies file record type. It should be FHEAD
	File Line ID	Number(10)	N/A	ID of current line being processed by input file
	File Type	Char(5)	FCUST	Identifies file as 'Franchise customer upload'
	File Create Date	Date	SYSDATE	Date file was written by external system
Transaction Header	File Record Descriptor	Char(5)	N/A	Identifies transaction record type. It should be THEAD
	File Line ID	Number(10)	N/A	ID of current line being processed by input file

Table 6-51 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
Transaction Detail	Message Type	Char(30)	N/A	Identifies the action that will be performed on the franchise customer transaction header record. It can be either create (fcustgrpre) or update (fcustgrpupd) or delete (fcustgrpdcl) a franchise customer group
	Franchise Customer group ID	Number(10)	N/A	Customer group ID
	Franchise Customer group Name	Char(120)	N/A	Customer group name. This field is optional for delete
	File Record Descriptor	Char(5)	N/A	Identifies transaction record type. It should be TDETL
	File Line ID	Number(10)	N/A	ID of current line being processed by input file
	Message Type	Char(30)	N/A	Identifies the action that will be performed on the franchise customer transaction detail record. It can be either create (fcustcre) or update (fcustupd) or delete (fcustdel) a franchise customer .
	Franchise Customer ID	Number(10)	N/A	Customer ID to be processed
	Franchise Customer Name	Char(120)	N/A	Customer Name

Table 6-51 (Cont.) File Layout

Record Name	Field Name	Field Type	Default Value	Description
Transaction Trailer	Credit Ind	Char(1)	N	This field will determine if the franchise customer has good credit. Valid values are Y and N
	Auto approve Ind	Char(1)	N	To auto approve the externally uploaded orders and returns. Valid values are Y and N
	File Record Descriptor	Char(5)	N/A	Identifies file record type. It should be TTAIL
	File Line ID	Number(10)	N/A	ID of current line being processed by input file
File Trailer	Transaction Record Count	Number(10)	N/A	Number of TDETL records in this transaction set.(total records between THEAD & TTAIL)
	File Record Descriptor	Char(5)	N/A	Identifies file record type. It should be FTAIL
	File Line ID	Number(10)	N/A	ID of current line being processed by input file.
	File Record Counter	Number(10)	N/A	Number of records/ transactions processed in current file (total records between FHEAD & FTAIL)

Design Assumptions

N/A

Franchise Order Upload (wfordupld.ksh)

Module Name	wfordupld.ksh
Description	Franchise Order Upload
Functional Area	Franchise Management
Module Type	Integration

Module Technology	ksh
Catalog ID	RMS60
Wrapper Script	batch_wfupload.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program is used to upload franchisee orders from an external source. These orders will be created with an order type of 'EDI' and will be created for the source type specified in the upload file. If source type is not specified, then the costing location for the item/franchise store will be used. Orders will be created in approved status if the customer is setup for auto approval, assuming that the customer has valid credit.

If the customer fails credit check or if available inventory at the source location is insufficient to fulfill the order, the order will be generated in input status. The check for inventory availability is subject to the setting of the "Validate Availability for External Franchise Orders" system option. If the option is set to Yes, then Franchise Order creation will be subject to checks for inventory availability. If set to No, Franchise Order creation would be carried out without checking for inventory availability.

Franchise orders from customers that are not identified for 'Auto Approval' are uploaded into Merchandising in input status. These orders will need to be manually approved in Merchandising in order to be considered active.

Restart/Recovery

The restart recovery is different from the conventional Merchandising batch. There are two points on the batch upload process where users can evaluate the successful load of the data.

- **SQL load** - At this point, SQL load dumps invalid records that do not meet certain technical requirements (for example: file layout issues, data type inconsistencies, and so on.). The rejected record is written to a bad file or to a discard file. The discard file contains records that do not satisfy conditions, such as missing or invalid record types. Records with other technical issues are written to the bad file.

Note

A non-fatal code is returned by the program and a message will be written to the log file if reject files are created.

Action Required: When such conditions exist, you may update either the bad or discard file and attempt to reload using the same files.

- **Business Validation** - At this point data from the file(s) are loaded into the staging table(s). PL/SQL functions determine if this loaded data is valid enough to be inserted into the actual Merchandising tables. For records that do not meet certain technical or business validations, the error message will be updated in staging table.

Action Required: When this condition exists, you can fix the data upload file and try to reload the file with valid data.

I/O Specification

Integration Type	Download from Merchandising
File Name	wford*.dat
Integration Contract	IntCon000108

SQL Loader Input File Layout

Table 6-52 SQL Loder Input File Layout

Record Name	Field Name	Field Type	Null allowed?	Default Value	Description
FHEAD	File head descriptor	Char(5)	No	FHEAD	Describes file line type.
	Line Number	Number(10)	No	N/A	Id of the current line being processed.
	Customer Id	Number(10)	No	N/A	Customer ID of the customer requesting the order.
	Customer Order Reference number	Char(20)	No	N/A	A reference field used by the customer for their tracking purposes.
	Currency Code	Char(3)	No	N/A	This is the currency on which the order was transacted.
	Default Billing location	Number(10)	No	N/A	A customer's location where the billing for the entire order is sent. If blank, each location is billed.
	Comments	Char(2000)	Yes	N/A	Any other miscellaneous information relating to the order.
FDETL	File record descriptor	Char(5)	No	FDETL	Describes file line type.
	Line Number	Number(10)	No	N/A	Id of the current line being processed.
	Item	Char(25)	No	N/A	The item on the franchise order.
	Customer Location	Number(10)	No	N/A	The franchise store requesting the order.

Table 6-52 (Cont.) SQL Loder Input File Layout

Record Name	Field Name	Field Type	Null allowed?	Default Value	Description
	Source Loc Type	Char(2)	Yes	N/A	Source location type for which the franchise order has been created. Valid values are ST - Store, WH - warehouse, or SU - Supplier
	Source Location	Number(10)	Yes	N/A	Source location for which the franchise order has been created. If the source location is warehouse then both physical and virtual warehouses are allowed.
	Requested Quantity	Number (12,4)	No	N/A	Number of item units being ordered, includes 4 implied decimal places
	Unit of Purchase	Char(3)	No	N/A	Unit of purchase can be the item's standard unit of measure, case, inners or pallets.
	Fixed Cost	Number (20,4)	Yes	N/A	This is cost which will be charged to the customer for the item on the franchise order; value includes 4 implied decimal places.
	Need Date	Char(11)	No	N/A	Date on which the item is needed in the franchise store, with the following format "DD-MON-YYYY" .

Table 6-52 (Cont.) SQL Loder Input File Layout

Record Name	Field Name	Field Type	Null allowed?	Default Value	Description
	Not After Date	Char(11)	No	N/A	Date after which the item may no longer be accepted for a franchise store, with the following format "DD-MON-YYYY".
FTAIL	File record descriptor	Char(5)		FTAIL	Marks end of file.
	Line Number	Number(10)		N/A	Id of current line being processed.
	File record count	Number(10)		N/A	Number of detail records.

Design Assumptions

N/A

Franchise Return Upload (wfretupld.ksh)

Module Name	wfretupld.ksh
Description	Franchise Return Upload
Functional Area	Franchise Management
Module Type	Integration
Module Technology	Ksh
Catalog ID	RMS154
Wrapper Script	batch_wfupload.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program is used for uploading franchise returns sent from an external source, such as an external order management application. When returns are uploaded in this manner, the data will be validated and the return will be created in Merchandising. Additionally, an associated franchise return transfer will also be created.

Restart/Recovery

The restart recovery is different from the conventional Merchandising batch. There are two points on the batch upload process where users can evaluate the successful load of the data.

- **SQL load** - At this point, SQL load dumps invalid records that do not meet certain technical requirements (for example: file layout issues, data type inconsistencies, and so on.). The rejected record is written either to a bad file or to a discard file. The discard file contains

records that do not satisfy conditions, such as missing or invalid record types. Records with other technical issues are written to the bad file.

Note

A non-fatal code is returned by the program and a message will be written to the log file if reject files are created. When such conditions exist, you may either update the bad or discard file and attempt to reload using the same files.

- **Business Validation** - At this point data from the file(s) are loaded into the staging table(s). PL/SQL functions determine if this loaded data is valid enough to be inserted into the actual Merchandising tables. For all records that do not meet certain technical or business validations, the error message will be updated in staging table. When this condition exists, you can fix the data upload file and try to reload the file with valid data.

I/O Specification

Integration Type	Upload to Merchandising
File Name	wfreturn*.dat
Integration Contract	Intcon000109

SQL Loader Input File Layout

The following is the file pattern for the upload file.

Note

The values are pipe "|" delimited and can optionally be enclosed by " ".

Table 6-53 SQL Loader Input File Layout

Record Name	Field Name	Field Type	Null Allowed?	Default Value	Description
FHEAD	File head descriptor	Char(5)	No	FHEAD	Describes file line type.
	Line Number	Number(10)	No		Id of the current line being processed.
	Customer ID	Number(10)	No		Franchise customer ID of the customer making the return.
	Customer Return Reference number	Char(20)	No		A reference field used by the franchise customer for their tracking purposes.
	Currency Code	Char(3)	No		This is the return currency.

Table 6-53 (Cont.) SQL Loader Input File Layout

Record Name	Field Name	Field Type	Null Allowed?	Default Value	Description
	Comments	Char(2000)	Yes		Any other miscellaneous information related to the return.
FDETL	File record descriptor	Char(5)	No	FDETL	Describes file line type.
	Line Number	Number(10)	No	N/A	Id of the current line being processed.
	Item	Char(25)	No	N/A	The item on the franchise return.
	Franchise Order Number	Number(10)	No	N/A	The franchise order number against which the return is made.
	Customer Location	Number(10)	No	N/A	The franchise location which is making the return.
	Return Loc Type	Char(1)	No	N/A	Return location type for the franchise return; valid values are S - store or W - warehouse.
	Return Location	Number(10)	No	N/A	Return location for the franchise return.
	Return Method	Char(1)	No	N/A	The type of return; valid values are: -R-Return to Store/Warehouse -D-Destroy at site
	Unit of measure	Char(3)	No	N/A	The unit measure of the return quantity. This is assumed to be the items standard UOM.
	Return qty	Number(12,4)	No	N/A	The quantity of item to be returned

Table 6-53 (Cont.) SQL Loader Input File Layout

Record Name	Field Name	Field Type	Null Allowed?	Default Value	Description
	Return Reason	Char(6)	No	N/A	Return reason code; valid values are found on the CODE_DETAIL table where CODE_TYPE is 'RTVR'.
	Return unit cost	Number(20,4)	Yes	N/A	The per unit cost for the return.
	Restock Type	Char(1)	No	N/A	Indicates how the restocking fee will be calculated per item; valid values are S-specific or V-value.
	Restock Fee	Number(20,4)	No	N/A	Unit restocking fee.
FTAIL	File record descriptor	Char(5)	No	FTAIL	Marks end of file.
	Line Number	Number(10)	No	N/A	Id of current line being processed.
	File record count	Number(10)	No	N/A	Number of detail records.

Design Assumptions

N/A

Upload Cost Buildup Template (fcosttmplupld)

Module Name	fcosttmplupld.ksh
Description	Upload Cost Buildup Template
Functional Area	Franchise Management
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS125
Wrapper Script	rmswrap_shell_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This module uploads cost buildup templates and franchise cost relationships used for franchise pricing from an external system into Merchandising staging tables. It also performs both

technical and business validation of the data sent in the file; for example, it validates that start and end dates are included for new and updated templates.

Note

No date format is specified in the input file, as any valid PL/SQL date format can be used.

Restart/Recovery

The restart recovery is different from the conventional Merchandising batch. There are three points on the batch upload process where users can evaluate the successful load of the data.

- SQL load - SQL load dumps invalid records that do not meet certain technical requirements (for example: file layout issues, data type inconsistencies, and so on.). The rejected record is written either to a bad file or to a discard file. The discard file contains records that do not satisfy conditions such as missing or invalid record types. Records with other technical issues are written to the bad file.

Note

A non-fatal code is returned by the program and a message will be written to the log file if reject files are created

Action Required: When such conditions exist, you may update either the bad or discard file and attempt to reload using the same files.

1. Business Validation Level - the data from the files are loaded into the staging tables for validation. PL/SQL functions determine if this loaded data is valid enough to be inserted into the actual Merchandising tables. Records that do not meet certain technical or business validations are rejected and the information is updated back into the staging table with an appropriate error message and the batch issues a NON-FATAL return code.

Action Required: When this condition exists, you can fix the data upload file and try to reload.

2. Chunking validated data - At this point the data from staging tables that have passed business validation are chunked based on the number of valid transactions (cost templates) and max_chunk_size from RMS_PLSQL_BATCH_CONFIG table. If there are no valid transactions to be chunked, batch issues a FATAL return code.

Action Required: When this condition exists, you can fix the data upload file and try to reload.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000021

SQL Loader Input File Layout

Table 6-54 SQL Loader Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	N/A	Identifies file record type. Valid value is FHEAD.
	File Line Identifier	Number(10)	N/A	Sequential file line number
	File Type Definition	Char(5)	CTMPL	Identifies file as 'Cost Template Upload'
	File Create Date	Date	SYSDATE	Date on which the file was created by external system
Transaction Header	File Record Descriptor	Char(5)	N/A	Identifies transaction header record type. Valid value is THEAD
	File Line Identifier	Number(10)	N/A	Sequential file line number
	Message Type	Char(30)	N/A	Identifies the action that will be performed on the franchise cost template header information that is provided as part of this record. It can be either create or update or delete a franchise cost template. Valid message types are: costtmpadd (for additions), costtmpmod (for updates), costtmpdel (for deletions)
	Template ID	Number(10)	N/A	Template ID
	Template Description	Char(120)	N/A	Template Description

Table 6-54 (Cont.) SQL Loader Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
Transaction Detail	Template Type	Char(1)	N/A	Indicates the type of the template. Valid values are M = Margin then Up-Charge, U = Up-charges, then Margin, R = % of Retail and C = Cost
	Percentage	Number(12,4)	N/A	Margin percent or % off Retail value; required if template type is M, U and R types of templates
	Cost	Number(20,4)	N/A	Indicates the franchise cost for an item when template type is 'C' This is mandatory and should only be populated if template type is 'C'
	Final Cost	Char(1)	N/A	Signifies if the cost is final or acquisition. Valid values are 'Y' or 'N'
	File Record Descriptor	Char(5)		Identifies transaction detail record type. Valid value is TDETL
	File Line Identifier	Number(10)		Sequential file line number

Table 6-54 (Cont.) SQL Loader Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Message Type	Char(30)		Identifies the action that will be performed on the franchise cost template relationship information that is provided as part of this record. It can be either create or update or delete a cost relationship. Valid values are: costtmpreladd (for additions), costtmprelmod (for updates), costtmpreldel (for deletions)
	Dept	Number(4)		Department associated with the cost template
	Class	Number(4)		Class associated with the cost template
	Subclass	Number(4)		Subclass associated with the cost template
	Item	Char(25)		Unique number that identifies a valid item associated with the template. Used for template types of 'C' only
	Location	Number(10)		Franchise Store Number associated with the template
	Start Date	Date		Date on which a cost template will be effective for the subclass/item and franchise store (required for update and delete of a cost relationship)

Table 6-54 (Cont.) SQL Loader Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	End Date	Date		Date on which a cost template will expire for a subclass/item and franchise store (required for update and delete of a cost relationship)
	New Start Date	Date		New Date on which a franchise cost relationship will be effective
	New End Date	Date		New Date on which a franchise cost relationship will expire
	Cost Component ID	Char(10)		Unique code which signifies the up-charge cost component when First_Applied is 'U' This should only be populated if First Applied is 'U'
Transaction Trailer	File Record Descriptor	Char(5)	N/A	Identifies transaction trailer record type. Valid value is TTAIL
	File Line Identifier	Number(10)	N/A	Sequential file line number
	Transaction Record Counter	Number(10)	N/A	Number of TDETL records in this transaction set
File Trailer	File Record Descriptor	Char(5)	N/A	Identifies file trailer record type. Valid value is TTAIL
	File Line Identifier	Number(10)	N/A	Sequential file line number
	File Record Counter	Number(10)	N/A	Number of records/ transactions processed in current file (only records between FHEAD & FTAIL)

Design Assumptions

N/A

Upload of Franchise Sales (wflsupld.ksh)

Module Name	wflsupld.ksh
Description	Upload of Franchise Sales to Merchandising
Functional Area	Franchise Management
Module Type	Integration
Module Technology	ksh
Catalog ID	RMS156
Wrapper Script	batch_wflsupld.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

Non-stockholding franchise stores in Merchandising are used for retailers who have franchise or other business customers for whom they supply inventory, but don't manage it for them. However, even though inventory information will not be available for these locations in Merchandising, sales information will be able to be uploaded to Merchandising via this process to allow retailers to have better visibility to future demand from these customers. In addition to uploading sales information, this same batch script also purges old non-stockholding franchise store sales records from Merchandising. The script runs in 4 modes:

- **Load** - this mode will load the data from the file into a staging table in Merchandising for processing; any errors encountered in validating the data on the upload are also written to the staging table (WFSLSUPLD_STAGING).
- **Process** - this mode will process the records in the staging table that did not have errors during load, which includes both writing the data to the WF_NONSTOCKHOLDING_SALES table, as well as purging the processed records from the staging table.
- **Reject** - this mode will process the records on the staging table that had errors on initial load. It will create a reject file for each location/report date with the data in error for that location/date. The records will then be deleted from the staging table.
- **Purge** - this mode is used to purge old sales records from the WF_NON_STOCKHOLDING_SALES table. Records are deleted based on the system parameter Non-stockholding Franchise Sales History days (WF_NON_STOCK_SALES_HIST_DAYS).

Restart/Recovery

The program can be restarted by running the wflsupld REJECT mode to create an input file of rejected records and wflsupld LOAD/PROCESS mode to reprocess the rejected records.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Input file name is a parameter during runtime
Integration Contract	IntCon000111

Input File Layout

Table 6-55 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type
	File Line Id	Char(10)	N/A	Sequential file line number
	File type definition	Char(4)	WFSU	Identifies the file type
	Customer Location	Number(10)	N/A	Store number identifier for the customer location
	Report Date	Char(14)	N/A	Report date of the file in YYYYMMDDHH MMSS format
	File Create Date	Char(14)	N/A	File Create Date in YYYYMMDDHH MMSS format
FDETL	Record descriptor	Char(5)	FDETL	Identifies the file record type
	File Line Id	Char(10)	N/A	Sequential file line number
	Item	Char(25)	N/A	Item number identifier
	Net Sales Quantity	Number(12)	N/A	Sales Quantity with 4 implied decimal places
	Net Sales Quantity UOM	Char(4)	N/A	Unit of Measure for the Net Sales Quantity
	Total Retail Amount	Number(20)	N/A	Total Retail Amount with 4 implied decimal places
FTAIL	Total Retail Amount Currency	Char(3)	N/A	Currency code for the Total Retail Amount
	Record descriptor	Char(5)	FTAIL	Identifies the file record type

Table 6-55 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	File Line Id	Number(10)	N/A	Sequential file line number
	File Record counter	Number(10)	N/A	Number of records/ transactions processed in current file (only records between head & tail)

Design Assumptions

N/A

Other Inventory

Other inventory related, scheduled inbound integrations include:

- [External Transaction Data Upload \(trandataload.ksh\)](#)
- [Upload and Process Inventory Reservations from Sales Audit \(ordinvupld\)](#)
- [Upload Item Availability for Type A & D Contracts from Suppliers \(ediupavl\)](#)

External Transaction Data Upload (trandataload.ksh)

Module Name	trandataload.ksh
Description	External Transaction Data Upload
Functional Area	Finance
Module Type	Integration
Module Technology	KSH
Catalog ID	RMS 376
Wrapper Script	batch_trandataload.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This process, along with trandataprocess.ksh, provides a mechanism to write records directly into the TRAN_DATA tables based on a file from an external system. The primary purpose of this functionality is to allow additional costs to be included in stock ledger valuation that cannot be included based on existing Merchandise functionality. Records written to the TRAN_DATA tables do not necessarily have a connection to any Merchandising transaction, and are based on a determination made outside of Merchandising. The records written through this mechanism function exactly the same as records written by normal Merchandising processes. For cost based transactions, the information must be passed at an item/location level. For retail-based transactions, it can be at either an item/location or subclass/location level.

Note

There is no support for recalculating or impacting unit inventory in Merchandising based on the transactions passed in, and only cost or retail value in the stock ledger is impacted - although the weighted average cost (WAC) may also be impacted if that method of accounting is used in Merchandising

The trandataload script loads the staging table STAGE_EXT_TRAN_DATA table from a flat file using SQL Loader and divides the data into chunks to be processed in parallel threads based on the commit_max_counter and num_threads value on RESTART_CONTROL table.

This script accepts the following input parameters:

- Database Connect string
- File load indicator – This indicator is passed as Y if a flat file has to be loaded into the table STAGE_EXT_TRAN_DATA else its N
- Input file – This is the path of the input file. This is mandatory when File load indicator is Y.

The SQL loading from a flat file is optional in the script. If File load indicator is Y the program validates if the input file exists and logs an error in case the input file does not exist. The SQL Load (sqlldr) process loads the input file using control file - trandataload.ctl into the STAGE_EXT_TRAN_DATA table.

- A fatal error from sqlldr will halt the process.
- Rejected records are a non-fatal error and loader will continue processing and create bad file and discard files in case the input file does not match the expected format.

If you chose not to load data into the staging table (File load indicator 'N') then the batch assumes that data has been loaded on the staging table from a different source. After the loading process is complete, the batch divides the data into chunks. If the staging table is empty or all the records are in 'P'rocessed status then the batch logs an appropriate error.

Chunking Logic

- Dense rank the staged records over Subclass, item and location.
- Divide the rank value by the commit max counter.
- Rounding the divided value gives the Chunk ID to which the particular value belongs to.
- Item can be NULL on the staging table, when NULL consider item to be '-999'.
- This will make sure the records with same subclass value and having item as NULL and NOT NULL are not grouped together in a chunk.

Since records with item have to be processed differently, (WAC recalculation and Variance postings) the batch makes sure that they fall in a different chunk to those records which do not have item value.

The Chunk data is inserted into STAGE_EXT_TRAN_DATA_CHUNK table.

Restart/Recovery

N/A

I/O Specification - Input File Specification

This batch uses SQL Loader to populate the staging table. The input file should be in pipe delimited format. Sample record structure would look like:

```
<item>|<dept>|<class>|<subclass>|<location>|<loc_type>|<tran_date>|<tran_code>|
<adj_code>|<units>|<total_cost>|<total_retail>|<ref_no_1>|<ref_no_2>|<GL_ref_no>|
<Old_unit_retail>|<New_unit_retail>|<Sales_type>|<VAT_rate>|<av_cost>|<ref_pack_no>|
<total_cost_excl_elc>|<WAC_reclculate_ind>|<status>|<create_timestamp>|
```

File Layout

The table below specifies the detail of each field in the record.

Table 6-56 File Layout

Field Name	Field Type	Default Value	Description / Constraints
Tran_id	NUMBER(10,0)	EXT_TRAN_DATA_SEQUENCE.NEXTVAL	This value is defaulted to a system generated value by the program.
Item	VARCHAR2(25)		Item is an optional field. Transactions can be uploaded at the Subclass level also.
Dept	NUMBER(4)		Mandatory Field
Class	NUMBER(4)		Mandatory Field
Subclass	NUMBER(4)		Mandatory Field
Loc_type	VARCHAR2(1)		Valid values - 'S', 'W', 'E'
Location	NUMBER(10)		Mandatory Field
Tran_date	DATE		Mandatory Field
Tran_code	NUMBER(4)		Mandatory Field
Adj_code	VARCHAR2(1)		Valid values – 'C', 'U', 'A'
Units	NUMBER(12, 4)		Mandatory Field
Total_cost	NUMBER(20, 4)		
Total_retail	NUMBER(20, 4)		
Ref_no_1	NUMBER(10)		
Ref_no_2	NUMBER(10)		
GL_ref_no	NUMBER(10)		
Old_unit_retail	NUMBER(20, 4)		
New_unit_retail	NUMBER(20, 4)		
Pgm_name	VARCHAR2(100)		
Sales_type	VARCHAR2(1)		Valid values – 'C', 'R', 'P'
Vat_rate	NUMBER(12, 4)		
Av_cost	NUMBER(20, 4)		
Ref_pack_no	VARCHAR2(25)		
Total_cost_excl_elc	NUMBER(20, 4)		

Table 6-56 (Cont.) File Layout

Field Name	Field Type	Default Value	Description / Constraints
Wac_recalculate_ind	VARCHAR2(1)		If Weighted Average Cost of the Item-Location should be recalculated after uploading this transaction then this value should be passed as 'Y'.
Status	VARCHAR2(1)	'N'	This value will be defaulted to 'N' by this program. It will be updated to 'P' once it has been processed else to 'E' in case of Error.
Err_msg	VARCHAR2(2500)		
Create_timestamp	DATE	Sysdate	
Last_updated_timestamp	DATE	Sysdate	

Design Assumptions

N/A

Upload and Process Inventory Reservations from Sales Audit (ordinvupld)

Module Name	ordinvupld.pc
Description	Upload and Process Inventory Reservations from Sales Audit
Functional Area	RMS
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS113
Wrapper Script	batch_ordinvupld.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This batch program processes the input file generated by the Sales Audit Inventory Export batch, which is generated to reserve and un-reserve inventory based on in-store customer orders and layaway. An in-store customer order is one where the customer is purchasing inventory present in the store, but will not take it home immediately. For example, with a large item like a sofa, the customer may pickup at a later time with a larger vehicle. Layaway is when a customer pays for an item over time and only takes the item home once it has been fully paid for. In processing this file, Merchandising updates the quantity of the item/location sent to either add or subtract from the quantity in the Customer Order inventory status type.

Restart/Recovery

The logical unit of work for this batch program is a valid item status transaction at a given store/location. The logical unit of work is defined as a group of these transaction records. The

Oracle Retail standard file-based restart/recovery logic is used. Records are committed to the database when the maximum commit counter is reached.

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000049

Input File Layout

Table 6-57 ordinvupld.pc - Input File

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Identifies the file record type
	File Line Id	Char(10)	0000000001	Sequential file line number
	File type definition	Char(4)	ORIN	Identifies the file type
	File Create Date	Char(14)	N/A	File Create Date in YYYYMMDDHH MMSS format
	Location	Number(10)	N/A	Store location number
THEAD	Record descriptor	Char(5)	THEAD	Identifies the file record type
	File Line Id	Char(10)	N/A	Sequential file line number
	Transaction Date & Time	Char(14)	Transaction Date	Date and time of the order processed
	Transaction Type	Char(6)	'SALE'	Transaction type code specifies whether the transaction is sale or Return
TDETL	Record descriptor	Char(5)	TDETL	Identifies the file record type
	File Line Id	Char(10)	N/A	Sequential file line number
	Item Type	Char(3)	REF or ITM	Can be REF or ITM
	Item	Char(25)	ITM	Id number of the ITM or REF

Table 6-57 (Cont.) ordinvupld.pc - Input File

Record Name	Field Name	Field Type	Default Value	Description
	Item Status	Char(6)	LIN - Layaway Initiate LCA - Layaway Cancel LCO - Layaway Complete PVLCO - Post void of Layaway complete ORI - Pickup/delivery Initiate ORC - Pickup/delivery Cancel ORD - Pickup/delivery Complete PVORD - Post void of Pick-up/delivery complete	Type of transaction
	Dept	Number(4)	N/A	Department of item sold or returned
	Class	Number(4)	N/A	Class of item sold or returned.
	Sub class	Number(4)	N/A	Subclass of item sold or returned
	Pack Ind	Char(1)	N/A	Pack indicator of item sold or returned
	Quantity Sign	Chanr(1)	'P' or 'N'	Sign of the quantity.
	Quantity	Number(12)	N/A	Quantity * 10000 (4 implied decimal places), number of units for the given order (item) status
	Selling UOM	Char(4)	N/A	UOM at which this item was sold
	Catchweight Ind	Char(1)	N/A	Indicates if the item is a catchweight item. Valid values are Y or NULL

Table 6-57 (Cont.) ordinuupld.pc - Input File

Record Name	Field Name	Field Type	Default Value	Description
	Customer Order number	Char(48)	N/A	Customer Order number
	Posting Store	Number(10)		Contains the store at which the item reservation/ reservation cancellation should occur in case of cross-store transactions happening at co-located stores. It is expected that this field will be populated only for items that are to be reserved (or have the reservation canceled) at a different store from the one at which the checkout happened.
TTAIL	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type
	File Line Identifier	Number(10)	Specified by Sales Audit	ID of current line being processed by input file.
	Transaction count	Number(6)	Specified by Sales Audit	Number of TDETL records in this transaction set
FTAIL	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file

Table 6-57 (Cont.) ordinuupld.pc - Input File

Record Name	Field Name	Field Type	Default Value	Description
	File Record Counter	Number(10)	N/A	Number of records/ transactions processed in current file (only records between FHEAD & FTAIL)

Design Assumptions

N/A

Upload Item Availability for Type A & D Contracts from Suppliers (ediupavl)

Module Name	ediupavl.pc
Description	Upload Item Availability for Type A & D Contracts from Suppliers
Functional Area	EDI - Contracts
Module Type	Integration
Module Technology	ProC
Catalog ID	RMS50
Wrapper Script	rmswrap_in_rej.ksh

Schedule

See Oracle Merchandising Batch Schedule.

Design Overview

This module runs to upload supplier availability information, which is a list of the items that a supplier has available. This information is used by Merchandising for type A and D contracts which require supplier availability information. The data uploaded is written to the SUP_AVAIL table.

Restart/Recovery

N/A

I/O Specification

Integration Type	Upload to Merchandising
File Name	Determined by runtime parameter
Integration Contract	IntCon000016

Input File Layout

Table 6-58 ediupavl.pc - File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor	Char(5)	FHEAD	Describes file line type
	Line number	Number(10)	0000000001	Sequential file line number
	File type	Char(4)	SPAV	N/A
	Create date	Char(14)	N/A	File create date in YYYYMMDDHH24 MISS format
FDETL	Record descriptor	Char(5)	FDETL	Describes file line type
	Line number	Number(10)	N/A	Sequential file line number
	Transaction number	Number(14)	N/A	Sequential transaction number
	Supplier	Number(10)	N/A	Indicates the supplier for whom the data applies
	Item type	Char(3)	N/A	Indicates the type of item contained in the file. Valid types are 'ITM', 'UPC', or 'VPN'
	Item id	Char(25)	N/A	Unique ID for the item
	Item supplement	Char(5)	N/A	UPC supplement
	Available quantity	Number(12)	N/A	Available quantity including 4 implied decimal places
FTAIL	Record descriptor	Char(5)	FTAIL	Number(10)
	Line number	Number(10)	N/A	Sequential file line number (total # lines in file)

Table 6-58 (Cont.) ediupavl.pc - File Layout

Record Name	Field Name	Field Type	Default Value	Description
	Number of detail records	Number(10)	N/A	Number of FDETL lines in file

Design Assumptions

This module will only be run if contracting is turned on in the system.

Fiscal Document Generation (FDG)

The Merchandising solution supports Fiscal Document Generation functionality. Fiscal document related data can be uploaded into FDG through scheduled inbound integration.

The following scheduled inbound integration is included in this section:

- [Fiscal Document Upload into FDG \(fdg_reim_job\)](#)

Fiscal Document Upload into FDG (fdg_reim_job)

Module Name	fdg_reim_job
Description	The script calls the FDG in order to migrate data between ReIM and FDG.
Functional Area	Rfm
Module Type	Admin – Ad hoc
Module Technology	Background Processing
Catalog ID	
Wrapper Script	fdg_reim_batch_sql.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this process is to migrate data from ReIM tables to FDG tables and then follow the existing process in the FDG module to finish the migration. ReIM will be the owner of the data to be stored in their staging tables, which will be consumed by batch, and will begin the migration process to FDG.

Restart/Recovery

N/A

Key Tables Affected

TABLE	SELECT	INSERT	UPDATE	DELETE
IM_FDG_DOC_HEAD	Yes	No	Yes	Yes
IM_FDG_DOC_ETT	Yes	No	No	Yes
IM_FDG_DOC_DTL	Yes	No	No	Yes
IM_FDG_DOC_DTL_PACK_COMP	Yes	No	No	Yes
IM_FDG_DOC_REF	Yes	No	No	Yes
IM_FDG_DOC_NON_MERCH	Yes	No	No	Yes
IM_FDG_DOC_TAX	Yes	No	No	Yes
IM_FDG_DOC_TEXT	Yes	No	No	Yes
IM_FDG_DOC_EXT	Yes	No	No	Yes
SVC_FDG_HDR	Yes	Yes	Yes	Yes
SVC_FDG_ETT	Yes	Yes	No	Yes
SVC_FDG_DTL	Yes	Yes	No	Yes
SVC_FDG_DTL_PACK	Yes	Yes	No	Yes
SVC_FDG_REF	Yes	Yes	No	Yes
SVC_FDG_NON_MERCH	Yes	Yes	No	Yes
SVC_FDG_TAX	Yes	Yes	No	Yes
SVC_FDG_TEXT	Yes	Yes	No	Yes
SVC_FDG_EXT	Yes	Yes	No	Yes
FDG_HDR	No	Yes	No	No
FDG_ETT	No	Yes	No	No
FDG_DTL	No	Yes	No	No
FDG_REF	No	Yes	No	No
FDG_NON_MERCH	No	Yes	No	No
FDG_TAX	No	Yes	No	No
FDG_DTL_PACK	No	Yes	No	No
FDG_TEXT	No	Yes	No	No
FDG_EXT	No	Yes	No	No
FDG_ERROR	No	Yes	No	No

Design Assumptions

N/A

Sales Processing

Merchandising and Sales Audit subscribe to data from point of sale (POS) and order management (OMS) solutions related to sales, returns, customer pick-ups, and so on. Generally, sales are first audited in Sales Audit and then sent to Merchandising for posting and inventory updates. However, customers can choose to bypass Sales Audit if using an external auditing solution or choosing not to audit sales data by sending data directly to Merchandising.

This section has been broken into the following sub-sections:

- [Sales Audit](#)
- [Sales Posting](#)

Sales Audit

The purpose of Sales Audit is to accept transaction data from point-of-sale (POS) and order management (OMS) solutions and move the data through a series of processes that culminate in "clean" data. Data that Sales Audit finds to be inaccurate is brought to the attention of the auditors who can use the features in Sales Audit to correct the exceptions.

For more information on Sales Audit processing see the *Merchandising Batch Operations Guide*.

This chapter contains details about the following integration processes used to import data to Sales Audit:

- [Customer Engagement Promotion Import \(CePromoBatch.ksh\)](#)
- [Import of Unaudited Transaction Data from POS to Sales Audit \(saimptlog/saimptlogi\)](#)
- [Import Total Value Adjustments From External Systems \(saimpadj\)](#)
- [Sales Audit Voucher Upload \(savouch\)](#)

Customer Engagement Promotion Import (CePromoBatch.ksh)

Module Name	CePromoBatch.ksh
Description	Invokes the Customer Engagement Promotion web service to fetch the promotions and saves it to the CE promo tables that will be used by the sagetref module.
Functional Area	Sales Audit
Module Type	Integration
Module Technology	ksh
Catalog ID	N/A
Wrapper Script	rmswrap_shell.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this script is to call the batch client that will execute the Oracle Retail Customer Engagement (ORCE) Promotion web service.

The values retrieved from the web service will populate the CE_PROMO and CE_PROMO_DEAL. These tables will be used by the Get Reference Data for Sales Audit Import Processing (sagetref) batch.

The list of valid promotions in ORCE will be extracted by using the retrieve promotions method under the Promotion Event Web Service offered by ORCE. This request will return promotion information including the promotion ID. Similarly the promotion component details in ORCE can be retrieved using the retrieve promotion deals method under the Promotion Event Web Service. The information returned through this request will include the deal ID. The data in the CE_PROMO and CE_PROMO_DEAL tables will be extracted into the promotion file format by

the sagetref batch and will be used to validate the RTLOG files being imported into Sales Audit.

Restart/Recovery

N/A

Key Tables Affected

Table	SELECT	INSERT	UPDATE	DELETE
CE_PROMO	No	Yes	No	Yes
CE_PROMO_DEAL	No	Yes	No	Yes

Design Assumptions

- The CE promotion web service URL will be different for each deployment. The URL will be set up during deployment and will become part of the batch environment variables.
- The ORCE web service is OAUTH enabled. The CePromoBatch.ksh will retrieve the necessary authentication through a REST API service call.

Import of Unaudited Transaction Data from POS to Sales Audit (saimptlog/saimptlogi)

Module Name	saimptlog.c saimptlogi.c
Description	Import of Unaudited Transaction data from POS to Sales Audit
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA11a RSA11b
Wrapper Script	batch_saimptlogi.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

Importing POS and Order Management System (OMS) data to Sales Audit is a five or six-step process depending on whether saimptlogi or saimptlog is used. Saimptlog produces SQL*Loader files while saimptlogi does inserts directly into the database. Saimptlogi is meant for use in a trickle feed environment.

To import POS and OMS data, perform the following:

1. SAGETREF must be run to generate the current reference files:
 - Items
 - Wastage
 - Sub-transaction level items

- Primary variant relationships
- Variable weight PLU
- Store business day
- Code types
- Error codes
- Store POS
- Tender type
- Merchant code types
- Partner vendor
- Supplier vendors
- Employee ids
- Banner ids
- Currency File
- Promotions File
- Warehouse File
- Inventory Status File

These files are all used as input to SAIMPTLOG and SAIMPTLOGI. Because SAIMPTLOG and SAIMPTLOGI can be threaded, this boosts performance by limiting interaction with the database.

2. Either SAIMPTLOG or SAIMPTLOGI must be run against each file. The files are the transaction log files in an Oracle Retail compatible format called RTLOG. The retailer is responsible for converting its transaction logs to RTLOGs. Both SAIMPTLOG and SAIMPTLOGI create a write lock, depending on the locking level specified in the Sales Audit System Options. It will create a write lock for a store/day combination on Sales Audit tables if the locking level indicated is Store Day. Otherwise, it will create a write lock for a transaction on Sales Audit tables if the locking level indicated is transaction. It will then set the data_status to loading until SAIMPTLOGFIN is executed. SAIMPTLOG generates distinct SQL*Loader files for that store/day for the sa_tran_head, sa_tran_head_attrib, sa_tran_item, sa_tran_item_attrib, sa_tran_disc, sa_tran_disc_attrib, sa_tran_igtax (item Level Tax not VAT), sa_tran_igtax_attrib (item Level Tax Attribute not VAT Attribute), sa_tran_payment (Payment details), sa_tran_tax, sa_tran_tax_attrib, sa_tran_tender, sa_tran_tender_attrib, sa_error, sa_customer, sa_cust_attrib, sa_tran_write_lock and sa_missing_tran tables, whereas SAIMPTLOGI inserts data to the database directly. Both produce an Oracle Retail formatted voucher file for processing.
3. SQL*Loader is executed to load the transaction tables from the files created by SAIMPTLOG. The store/day SQL*Loader files can be concatenated into a single file per table to optimize load times. Alternatively, multiple SQL*Loader files can be used as input to SQL*Loader. SQL*Loader may not be run in parallel with itself when loading a table. Header data (primary keys) must be loaded before ancillary data (foreign keys). This means that the sa_tran_head table must be loaded first, sa_tran_item before sa_tran_disc, and sa_customer before sa_cust_attrib. The main tables of each attribute table must be loaded first, before its attributes. This means that the sa_tran_item must be loaded first, before sa_tran_item_attrib; the sa_tran_disc must be loaded first, before sa_tran_disc_attrib; the sa_tran_igtax must be loaded first, before sa_tran_igtax_attrib; the sa_tran_tender must be loaded first, before sa_tran_tender_attrib; the sa_tran_tax must be loaded first, before sa_tran_tax_attrib. The remaining tables may be loaded in parallel.

4. SAVOUCH is executed to load each of the voucher files in Oracle Retail standard formatted. SAVOUCH may not be multi-threaded.
5. SAIMPTLOGFIN is executed to populate the sa_balance_group table, cancel post voided transactions and vouchers, validate missing transactions, and to mark the import as either partially or fully complete loaded. SAIMPTLOGFIN may not be multi-threaded.

Note

This design covers only Steps 2 and 3.

Restart and Recovery

N/A

File Upload Error Handling

For each RTLOG file, a record is written to the FILE_UPLOAD_STATUS table. In cases where a non-fatal error occurs after validating a record in the file, the error is written to the error file and a corresponding record is also inserted to the FILE_UPLOAD_ERRORS table.

I/O Specification

Integration Type	Upload to Sales Audit
File Name	Determined by runtime parameter
Integration Contract	Inputs from sagetref.pc: IntCon000113 (itemfile) IntCon000114 (wastefile) IntCon000115 (refitemfile) IntCon000116 (primvariantfile) IntCon000117 (varupcfile) IntCon000118 (storedayfile) IntCon000119 (promfile) IntCon000120 (codesfile) IntCon000121 (errorfile) IntCon000122 (storeposfile) IntCon000123 (tendertypefile) IntCon000124 (merchcodesfile) IntCon000125 (partnerfile) IntCon000126 (supplierfile) IntCon000127 (employeefile) IntCon000128 (bannerfile) IntCon000129 (promfile) IntCon000130 (whfile) IntCon000131 (invstatusfile) Inputs from POS: IntCon000048 (RTLOG)

Outputs (if using saimptlog SQL Loader Option note that saimptlogi inserts directly into Sales Audit tables and does not create these output files)

IntCon000160 (SAVO)
 IntCon000161 (satdisc.ctl)
 IntCon000162 (saigtax.ctl)
 IntCon000163 (sacust.ctl)
 IntCon000164 (sathead.ctl)
 IntCon000165 (satitem.ctl)
 IntCon000166 (sattend.ctl)
 IntCon000167 (satypmt.ctl)
 IntCon000168 (samisstr.ctl)
 IntCon000169 (sattax.ctl)
 IntCon000170 (sacustatt.ctl)
 IntCon000171 (saerror.ctl)
 IntCon000172 (sathatt.ctl)
 IntCon000173 (saitatt.ctl)
 IntCon000174 (saidatt.ctl)
 IntCon000175 (saixatt.ctl)
 IntCon000176 (satxatt.ctl)
 IntCon000177 (sattatt.ctl)
 (satwritelock.ctl)

The input files for this program are reference files generated by sagetref.pc and RTLOGs. Refer to the details for the sagetref.pc program for the input file specifications.

Output File Layout

Table 6-59 File Name: SAVO (Sales Audit Voucher File)

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File Type Record Descriptor	Char(5)	FHEAD	File type Record descriptor
	SA File Line No	Char(10)	N/A	Sales Audit File Line number
	Translator Id	Char(5)	SAVO	Identifies transaction type
	Sys Date	Char(14)	N/A	System date in YYYYMMDDHHMMSS format
	Is business date	Char(8)	N/A	Business date in YYYYMMDD format
FDETL	Record Descriptor	Char(5)	FDETL	File Type Record descriptor
	SA File Line No	Number(10)	N/A	Sales Audit File Line number
	Voucher seq Number	Number(20)	N/A	Unique identifier for an entry to sa_voucher table
	Voucher No	Char(25)	N/A	Voucher Number
	Voucher Type	Number(6)	N/A	Voucher Type
	Assigned Business Date	Char(8)	N/A	Business date in YYYYMMDD format

Table 6-59 (Cont.) File Name: SAVO (Sales Audit Voucher File)

Record Name	Field Name	Field Type	Default Value	Description
	Assigned Store	Number(10)	N/A	Store to which the voucher is assigned
	Issuing Date	Char(8)	N/A	Date this document was issued
	Issuing store	Number(10)	N/A	Store this document was issued from
	Issuing POS Register	Char(5)	N/A	Issuing Point Of Sale register
	Issuing Cashier	Char(10)	N/A	Issuing cashier
	Issued Tran Seq No.	Number(20)	N/A	Transaction sequence number
	Issued item seq number	Number(4)	N/A	Will hold the item sequence of the item when the voucher is sold as an item (gift voucher)
	Issued Tender Seq No.	Number(4)	N/A	Tender sequence number
	Issued Amount	Number(20)	N/A	Issued Amount * 10000 (4 implied digits)
	Issued Cust Name	Char(120)	N/A	Issued customer name
	Issued Customer Addr1	Char(240)	N/A	Issued customer addr1
	Issued Customer Addr2	Char(240)	N/A	Issued customer addr 2
	Issued Customer City	Char(120)	N/A	City of the customer, the voucher is issued
	Issued Customer State	Char(3)	N/A	State of the customer
	Issued Customer Postal Code	Char(30)	N/A	Postal address of the customer
	Issued Customer Country	Char(3)	N/A	Country of the customer the voucher was issued
	Recipient Name	Char(120)	N/A	Name of the intended recipient
	Recipient State	Char(3)	N/A	The state of the intended recipient
	Recipient Country	Char(3)	N/A	The country of the intended recipient
	Redemption Date	Char(8)	N/A	Date the voucher was redeemed
	Redemption Store	Number(10)	N/A	Store, the voucher was redeemed at
	Redemption Register	Char(5)	N/A	Register, the document was redeemed at

Table 6-59 (Cont.) File Name: SAVO (Sales Audit Voucher File)

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	Redemption cashier	Char(10)	N/A	Cashier redeeming the voucher
	Redemption tran seq number	Number(20)	N/A	Transaction number when the document was redeemed
	Redemption Tender seq number	Number(4)	N/A	This column will hold the tender sequence of the tender within the transaction when a voucher is redeemed as tender
	Redemption Amount	Number(20)	N/A	Amount the document was redeemed for*10000 (4 implied decimal places)
	Expiry Date	Char(8)	N/A	Expiry date
	Status	Char(1)	N/A	Indicator showing the document's status, issued or redeemed. Valid values = I - Issued, R - Redeemed
	Comments	Char(2000)	N/A	Comments
	Record Descriptor	Char(5)	FTAIL	File Type Record descriptor
	SA File Line No.	Number(10)	N/A	Sales Audit File Line Number
	#lines	Number(10)	N/A	Total number of transaction lines in file (not including FHEAD and FTAIL)

Control Files

Table 6-60 File Name: Satdisc.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_DISC	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	ITEM_SEQ_NO	Integer external	4	21:24	N/A
	DISCOUNT_SEQ_NO	Integer external	4	25:28	N/A
	RMS_PROMO_TYPE	Char	6	29:34	N/A
	PROMOTION	Integer external	10	35:44	N/A
	DISC_TYPE	Char	6	45:50	N/A
	COUPON_NO	Char	40	51:90	N/A
	COUPON_REF_NO	Char	16	91:106	N/A
	QTY	Decimal external	14	107:120	N/A

Table 6-60 (Cont.) File Name: Satdisc.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	UNIT_DISCOUNT_AMT	Decimal external	21	121:141	N/A
	STANDARD_QTY	Decimal external	14	142:155	N/A
	STANDARD_UNIT_DISC_AMT	Decimal external	21	156:176	N/A
	REF_NO13	Char	30	177:206	N/A
	REF_NO14	Char	30	207:236	N/A
	REF_NO15	Char	30	237:266	N/A
	REF_NO16	Char	30	267:296	N/A
	ERROR_IND	Char	1	297:297	N/A
	CATCHWEIGHT_IN D	Char	1	298:298	N/A
	UOM_QUANTITY	Integer external	12	299:310	N/A
	PROMO_COMP	Integer external	10	311:320	This field maps to the OFFER_ID field from Pricing
	STORE	Integer external	10	321:330	N/A
	DAY	Integer external	3	331:333	N/A

Table 6-61 File Name: Saigtax.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_IGT AX	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	ITEM_SEQ_NO	Integer external	4	21:24	N/A
	IGTAX_SEQ_NO	Integer external	4	25:28	N/A
	TAX_AUTHORITY	Char	10	29:38	N/A
	IGTAX_CODE	Char	6	39:44	N/A
	IGTAX_RATE	Decimal external	11	45:65	N/A
	TOTAL_IGTAX_AM T	Decimal external	22	66:87	N/A
	STANDARD_QTY	Decimal external	14	88:101	N/A
	STANDARD_UNIT_I GTAX_AMT	Decimal external	21	102:122	N/A
	ERROR_IND	Char	1	123:123	N/A

Table 6-61 (Cont.) File Name: Saigtax.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
	REF_NO_21	Char	30	124:153	N/A
	REF_NO_22	Char	30	154:183	N/A
	REF_NO_23	Char	30	184:213	N/A
	REF_NO_24	Char	30	214:243	N/A
	STORE	Integer external	10	244:253	N/A
	DAY	Integer external	3	254:256	N/A
	TAX_CALC_TYPE	Char	6	257:262	N/A

Table 6-62 File Name: Sacust.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_CUSTOME R	TRAN_SEQ_NO	Integer external Date	20	1 :20	N/A
	CUST_ID	Char	16	21 :36	N/A
	CUST_ID_TYPE	Char	6	37 :42	N/A
	NAME	Char	240	43 :162	N/A
	ADDR1	Char	240	163:402	N/A
	ADDR2	Char	240	403:642	N/A
	CITY	Char	240	643:762	N/A
	STATE	Char	3	763:765	N/A
	POSTAL_CODE	Char	30	766:795	N/A
	COUNTRY	Char	3	796:798	N/A
	HOME_PHONE	Char	20	799:818	N/A
	WORK_PHONE	Char	20	819:838	N/A
	E_MAIL	Char	100	839:938	N/A
	BIRTHDATE	Date	8	939:946	Format is YYYYMMDD
	STORE	Integer external	10	947:956	N/A
	DAY	Integer external	3	957:959	N/A

Table 6-63 File Name: Sathead.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_HE AD	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	REV_NO	Integer external	3	21:23	N/A

Table 6-63 (Cont.) File Name: Sathead.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	STORE_DAY_SEQ_NO	Integer external	20	24:43	N/A
	TRAN_DATETIME	Date	14	44:57	Format is YYYYMM DDHH24MI SS
	REGISTER	Char	5	58:62	N/A
	TRAN_NO	Integer external	10	63:72	N/A
	CASHIER	Char	10	73:82	N/A
	SALESPERSON	Char	10	83:92	N/A
	TRAN_TYPE	Char	6	93:98	N/A
	SUB_TRAN_TYPE	Char	6	99:104	N/A
	ORIG_TRAN_NO	Integer external	10	105:114	N/A
	ORIG_REG_NO	Char	5	115:119	N/A
	REF_NO1	Char	30	120:149	N/A
	REF_NO2	Char	30	150:179	N/A
	REF_NO3	Char	30	180:209	N/A
	REF_NO4	Char	30	210:239	N/A
	REASON_CODE	Char	6	240:245	N/A
	VENDOR_NO	Char	10	246:255	N/A
	VENDOR_INVC_NO	Char	30	256:285	N/A
	PAYMENT_REF_NO	Char	16	286:301	N/A
	PROOF_OF_DELIVERY_NO	Char	30	302:331	N/A
	STATUS	Char	6	332:337	N/A
	VALUE	Char	22	338:359	Includes an optional negative sign and a decimal point
	POS_TRAN_IND	Char	1	360:360	N/A
	UPDATE_ID	Char	30	361:390	N/A
	UPDATE_DATETIME	Date	14	391:404	Format is YYYYMM DDHH24MI SS
	ERROR_IND	Char	1	405:405	N/A
	BANNER_NO	Integer external	4	406:409	N/A

Table 6-63 (Cont.) File Name: Sathead.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	ROUND_AMT	Integer external	22	410:431	N/A
	ROUNDED_OFF_A MT	Integer external	22	432:453	N/A
	CREDIT_PROMOTI ON_ID	Integer external	10	454:463	N/A
	REF_NO25	Char	30	464:493	N/A
	REF_NO26	Char	30	494:523	N/A
	REF_NO27	Char	30	524:553	N/A
	STORE	Integer external	10	554:563	N/A
	DAY	Integer external	3	564:566	N/A
	RTLOG_ORIG_SYS	Char	3	567:569	N/A
	TRAN_PROCESS_ SYS	Char	3	570:572	N/A
	TRAN_DATE	Date	8	573:580	N/A
	REF_NO28	Char	30	581:610	N/A
	REF_NO29	Char	30	611:640	N/A
	REF_NO30	Char	30	641:670	N/A
	REF_NO31	Char	30	671:700	N/A

Table 6-64 File Name: Satitem.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_ITEM	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	ITEM_SEQ_NO	Integer external	4	21:24	N/A
	ITEM_STATUS	Char	6	25:30	N/A
	ITEM_TYPE	Char	6	31:36	N/A
	ITEM	Char	25	37:61	N/A
	REF_ITEM	Char	25	62:86	N/A
	NON_MERCH_ITEM	Char	25	87:111	N/A
	VOUCHER_NO	Char	25	112:136	N/A
	DEPT	Integer external	4	137:140	N/A
	CLASS	Integer external	4	141:144	N/A
	SUBCLASS	Integer external	4	145:148	N/A

Table 6-64 (Cont.) File Name: Satitem.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	QTY	Decimal external	14	149:162	Includes an optional negative sign and a decimal point
	UNIT_RETAIL	Decimal external	21	163:183	Includes a decimal point
	UNIT_RETAIL_VAT_INCL	Char	1	184:184	Indicates whether unit retail includes or excludes VAT
	SELLING UOM	Char	4	185:188	N/A
	OVERRIDE_REASON	Char	6	189:194	N/A
	ORIG_UNIT_RETAIL	Decimal external	21	195:215	Includes a decimal point
	STANDARD_ORIG_UNIT_RETAIL	Decimal external	21	216:236	N/A
	TAX_IND	Char	1	237:237	N/A
	ITEM_SWIPED_IN	Char	1	238:238	N/A
	ERROR_IND	Char	1	239:239	N/A
	DROP_SHIP_IND	Char	1	240:240	N/A
	WASTE_TYPE	Char	6	241:246	N/A
	WASTE_PCT	Decimal external	12	247:258	Includes a decimal point
	PUMP	Char	8	259:266	N/A
	RETURN_REASON_CODE	Char	6	267:272	N/A
	SALESPERSON	Char	10	273:282	N/A
	EXPIRATION_DATE	Date	8	283:290	Format is YYYYMMDD
	STANDARD_QTY	Decimal external	14	291:304	Includes an optional negative sign and a decimal point
	STANDARD_UNIT_RETAIL	Decimal external	21	305:325	Includes a decimal point
	STANDARD_UOM	Char	4	326:329	N/A
	REF_NO5	Char	30	330:359	N/A
	REF_NO6	Char	30	360:389	N/A
	REF_NO7	Char	30	390:419	N/A
	REF_NO8	Char	30	420:449	N/A

Table 6-64 (Cont.) File Name: Satitem.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
	CATCHWEIGHT_IN D	Char	1	450:450	N/A
	SELLING_ITEM	Char	25	451:475	N/A
	CUSTOMER_ORD ER_LINE_NO	Integer external	6	476:481	N/A
	MEDIA_ID	Integer external	10	482:491	N/A
	UOM_QUANTITY	Integer external	12	492:503	N/A
	TOTAL_IGTAX_AM T	Decimal external		504:524	N/A
	UNIQUE_ID	Char	25	525:652	N/A
	STORE	Integer external	10	653:662	N/A
	DAY	Integer external	3	663:665	N/A
	CUST_ORDER_NO	Char	48	666:713	N/A
	CUST_ORDER_DA TE	Date	14	714:727	Format is YYYYMMDDH H24MISS
	FULFILL_ORDER_ NO	Char	48	728:775	N/A
	NO_INV_RET_IND	Char	1	776:776	N/A
	SALES_TYPE	Char	1	777:777	N/A
	RETURN_WH	Integer external	10	778:787	N/A
	RETURN_DISPOSI TION	Char	10	788:797	N/A
	ORIG_STORE	Integer external	10	798:807	N/A
	ORIG_TRAN_NO	Integer external	10	808:817	N/A
	FULFILLMENT_LO C_TYPE	Char	2	818:820	N/A
	FULFILLMENT_LO C	Integer external	10	821:830	N/A
	POSTING_STORE	Integer external	10	830:839	N/A
	CONSIGNMENT_R ATE	DECIMAL EXTERNAL	14	840:853	INCLUDES A DECIMAL POINT.
	CONSIGNMENT_U NIT_COST	DECIMAL EXTERNAL	21	854:874	INCLUDES A DECIMAL POINT.

Table 6-64 (Cont.) File Name: Satitem.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	INVENTORY_IDENTIFIER_TYPE	CHAR	6	875:880	This field contains the inventory identifier type passed in Sales/Return transactions. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Serial Number (S), Import Document (D).
	INVENTORY_ID	CHAR	120	881:1000	This field contains the inventory ID value being passed in sales/return transactions.

Table 6-65 File Name: Sattend.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_TENDER	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	TENDER_SEQ_NO	Integer external	4	21:24	N/A
	TENDER_TYPE_GROUP	Char	6	25:30	N/A
	TENDER_TYPE_ID	Integer external	6	31:36	N/A
	TENDER_AMT	Decimal external	22	37:58	Includes an optional negative sign and a decimal point.
	CC_NO	Integer external	40	59:98	N/A
	CC_EXP_DATE	Date	8	99:106	FORMAT IS YYYYMMDD
	CC_AUTH_NO	Char	16	107:122	N/A
	CC_AUTH_SRC	Char	6	123:128	N/A

Table 6-65 (Cont.) File Name: Sattend.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	CC_ENTRY_MODE	Char	6	129:134	N/A
	CC_CARDHOLDER _VERF	Char	6	135:140	N/A
	CC_TERM_ID	Char	5	141:145	N/A
	CC_SPEC_COND	Char	6	146:151	N/A
	CC_TOKEN	Char	40	152:191	N/A
	VOUCHER_NO	Char	25	192:216	N/A
	COUPON_NO	Char	40	217:256	N/A
	COUPON_REF_NO	Char	16	257:272	N/A
	CHECK_ACCT_NO	Char	30	273:302	N/A
	CHECK_NO	Integer external	10	303:312	N/A
	IDENTI_METHOD	Char	6	313:318	N/A
	IDENTI_ID	Char	40	319:358	N/A
	ORIG_CURRENCY	Char	3	359:361	N/A
	ORIG_CURR_AMT	Decimal external	22	362:383	N/A
	REF_NO9	Char	30	384:413	N/A
	REF_NO10	Char	30	414:443	N/A
	REF_NO11	Char	30	444:473	N/A
	REF_NO12	Char	30	474:503	N/A
	ERROR_IND	Char	1	504:504	N/A
	STORE	Integer external	10	505:514	N/A
	DAY	Integer external	3	515:517	N/A

Table 6-66 File Name: Satpymt.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_PA YMENT	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	PAYMENT_SEQ_N O	Integer external	20	21:24	N/A
	PAYMENT_AMT	Decimal external	5	25:46	N/A
	ERROR_IND	Char	10	47:47	N/A
	STORE	Integer external	6	48:57	N/A
	DAY	Integer external	3	58:60	N/A

Table 6-67 File Name: Samisstr.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_MISSING_TRAN	MISS_TRAN_SEQ_NO	Integer external	20	1:20	N/A
	STORE_DAY_SEQ_NO	Integer external	20	21:40	N/A
	REGISTER	Char	5	41:45	N/A
	TRAN_NO	Integer external	10	46:55	N/A
	STATUS	Char	6	56:61	N/A
	RTLOG_ORIG_SYS	Char	3	62:64	N/A

Table 6-68 File Name: Sattax.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_TAX	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	TAX_CODE	Char	6	21:26	N/A
	TAX_SEQ_NO	Integer external	4	27:30	N/A
	TAX_AMT	Decimal external	22	31:52	Includes an optional negative sign and a decimal point
	ERROR_IND	Char	1	53:53	N/A
	REF_NO17	Char	30	54:83	N/A
	REF_NO18	Char	30	84:113	N/A
	REF_NO19	Char	30	114:143	N/A
	REF_NO20	Char	30	144:173	N/A
	STORE	Integer external	10	174:183	N/A
	DAY	Integer external	3	184:186	N/A

Table 6-69 File Name: Sacustatt.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_CUST_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	N/A
	ATTRIB_SEQSO	Char	4	21:24	N/A
	ATTRIB_TYPE	Char	6	25:30	N/A
	ATTRIB_VALUE	Char	6	31:36	N/A
	STORE	Integer external	10	37:46	N/A

Table 6-69 (Cont.) File Name: Sacustatt.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
	DAY	Integer external	3	47:49	N/A

Table 6-70 File Name: Saerror.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_ERROR	ERROR_SEQ_NO	Integer external	20	1:20	N/A
	STORE_DAY_SEQ_NO	Integer external	20	21:40	N/A
	BAL_GROUP_SEQ_NO	Integer external	20	41:60	N/A
	TOTAL_SEQ_NO	Integer external	20	61:80	N/A
	TRAN_SEQ_NO	Integer external	20	81:100	N/A
	ERROR_CODE	Char	25	101:125	N/A
	KEY_VALUE_1	Integer external	4	126:129	N/A
	KEY_VALUE_2	Integer external	4	130:133	N/A
	REC_TYPE	Char	6	134:139	N/A
	STORE_OVERRIDE_IND	Char	1	140:140	N/A
	HQ_OVERRIDE_IND	Char	1	141:141	N/A
	UPDATE_ID	Char	30	142:171	N/A
	UPDATE_DATE TIME	Date	14	172:185	Format is YYYYMMDDH H24MISS
	ORIG_VALUE	Char	70	186:255	N/A
	STORE	Integer external	10	256:265	N/A
	DAY	Integer external	3	266:268	N/A
	KEY_VALUE_3	Integer external	4	269:272	N/A

Table 6-71 File Name: Sathatt.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_HEAD_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	

Table 6-71 (Cont.) File Name: Sathatt.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	ATTRIB_SEQ_NO	Integer external	4	21:24	
	ATTRIB_TYPE	Char	6	25:30	
	ATTRIB_VALUE	Char	120	31:150	
	STORE	Integer external	10	151:160	
	DAY	Integer external	3	161:163	
	ERROR_IND	Char	1	164:164	

Table 6-72 File Name: Saitatt.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_ITEM_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	
	ITEM_SEQ_NO	Integer external	4	21:24	
	ATTRIB_SEQ_NO	Integer external	4	25:28	
	ATTRIB_TYPE	Char	6	29:34	
	ATTRIB_VALUE	Char	120	35:154	
	STORE	Integer external	10	155:164	
	DAY	Integer external	3	165:167	
	ERROR_IND	Char	1	168:168	

Table 6-73 File Name: Saidatt.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_DISC_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	
	ITEM_SEQ_NO	Integer external	4	21:24	
	DISCOUNT_SEQ_NO	Integer external	4	25:28	
	ATTRIB_SEQ_NO	Integer external	4	29:32	
	ATTRIB_TYPE	Char	6	33:38	
	ATTRIB_VALUE	Char	120	39:158	
	STORE	Integer external	10	159:168	
	DAY	Integer External	3	169:171	

Table 6-73 (Cont.) File Name: Saidatt.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
	ERROR_IND	Char	1	172:172	

Table 6-74 File Name: Saixatt.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_IGT AX_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	
	ITEM_SEQ_NO	Integer external	4	21:24	
	IGTAX_SEQ_NO	Integer external	4	25:28	
	ATTRIB_SEQ_NO	Integer external	4	29:32	
	ATTRIB_TYPE	Char	6	33:38	
	ATTRIB_VALUE	Char	120	39:158	
	STORE	Integer external	10	159:168	
	DAY	Integer external	3	169:171	
	ERROR_IND	Char	1	172:172	

Table 6-75 File Name: Satxatt.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_TA X_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	
	TAX_SEQ_NO	Integer external	4	21:24	
	ATTRIB_SEQ_NO	Integer external	4	25:28	
	ATTRIB_TYPE	Char	6	29:34	
	ATTRIB_VALUE	Char	120	35:154	
	STORE	Integer external	10	155:164	
	DAY	Integer external	3	165:167	
	ERROR_IND	Char	1	168:168	

Table 6-76 File Name: Sattatt.ctf

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_TE NDER_ATTRIB	TRAN_SEQ_NO	Integer external	20	1:20	

Table 6-76 (Cont.) File Name: Sattatt.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
	TENDER_SEQ_NO	Integer external	4	21:24	
	ATTRIB_SEQ_NO	Integer external	4	25:28	
	ATTRIB_TYPE	Char	6	29:34	
	ATTRIB_VALUE	Char	120	35:154	
	STORE	Integer external	10	155:164	
	DAY	Integer external	3	165:167	
	ERROR_IND	Char	1	168:168	

Table 6-77 File Name: Satwritelock.ctl

Table Name	Column Name	Field Type	Field Width	Position	Description
SA_TRAN_WR ITE_LOCK	STORE_DAY_SEQ _NO	Integer external Date	20	1:20	N/A
	TRAN_SEQ_NO	Integer external Date	20	21:40	N/A

Sales Audit Interface File Layout [rtlog]

The following illustrates the file layout format of the Oracle Retail TLOG. The content of each Oracle Retail TLOG file is per store per day. The filename convention is RTLOG_STORE_DATETIME.DAT (for example, RTLOG_1234_01221989010000.DAT).

Involves round off fields, credit promotion id, tax (vat) at item level and payment amount of customer orders.

Document has been modified regarding tender types, logic of handling both VAT-TAX in the system has been added.

Retailers must ensure that credit card numbers are masked when sent through RTLOGs. Similarly, when the tender type is check, checking account numbers must be masked when sent through RTLOGs. When Sales Audit encounters an RTLOG with a non-masked credit card or checking account number, the entire file will be rejected and will not be processed.

```

FHEAD      (Only 1 per file, required)
THEAD      (Multiple expected, one per transaction, required for each transaction)
THATT      (Attribute record specific to the THEAD record - Multiple allowed, optional)
TCUST      (Only 1 per THEAD record allowed, optional for some transaction types, see
table below)
CATT      (Attribute record specific to the TCUST record - Multiple allowed, only valid
if TCUST exists)
TITEM      (Multiple allowed per transaction, optional for some transaction types, see
table below)
ITATT      (Attribute record specific to the TITEM record - Multiple allowed, optional
and only valid if TITEM exists)

```

IDISC (Discount record specific to the TITEM record - Multiple allowed per item, optional see table below)

IDATT (Attribute record specific to the IDISC record - Multiple allowed, optional and only valid if IDISC exists)

IGTAX (VAT/Tax record specific to the TITEM record - Multiple allowed per item, optional. Either TTAX or IGTAX should appear in a given RTLOG, if originating system is POS, but not both, see table below). If originating system is OMS, both IGTAX and TTAX can appear but only the one matching the store's tax type will be processed, the other record will be ignored.

IXATT (Attribute record specific to the IGTAX record - Multiple allowed, optional and only valid if IGTAX exists)

TTAX (Vat/Tax record specific to the THEAD record - Multiple allowed per transaction, optional. Either TTAX or IGTAX should appear in a given RTLOG, if originating system is POS, but not both, see table below). If originating system is OMS, both IGTAX and TTAX can appear but only the one matching the store's tax type will be processed, the other record will be ignored.

TXATT (Attribute record specific to the TTAX record - Multiple allowed, optional and only valid if TTAX exists)

TPYMT (Multiple allowed per transaction, will have the deposit amount for pickup/delivery/layaway orders, optional see table below)

TTEND (Multiple allowed per transaction, optional for some transaction types, see table below)

TTATT (Attribute record specific to the TTEND record - Multiple allowed, optional and only valid if TTEND exists)

TTAIL (1 per THEAD, required)

FTAIL (1 per file, required)

The order of the records within the transaction layout above is important. It aids processing by ensuring that the information is present when it is needed.

Fields expected in RTLog format based on the changes adopted -

Version Number	1	2a	2b	3	4	5	6
Version Description	Initial Version (V16.0.x SaaS)	Initial Version Plus Fulfillment Type and Location	Initial Version Plus Reference Numbers 28-31	Initial Version Plus Fulfilment Type and Location and Reference Numbers 28-31	Version 3 Plus Posting Store	Version 4 Plus Consignment Rate/ Cost	Version 5 Plus Inventory Identifier Type/ Inventory Id
Reference No. 28	N	N	Y	Y	Y	Y	Y
Reference No. 29	N	N	Y	Y	Y	Y	Y
Reference No. 30	N	N	Y	Y	Y	Y	Y
Reference No. 31	N	N	Y	Y	Y	Y	Y
Fulfillment Location type	N	Y	N	Y	Y	Y	Y
Fulfillment Location	N	Y	N	Y	Y	Y	Y
Posting Store	N	N	N	N	Y	Y	Y

Version Number	1	2a	2b	3	4	5	6
Version Description	Initial Version (V16.0.x SaaS)	Initial Version Plus Fulfillment Type and Location	Initial Version Plus Reference Numbers 28-31	Initial Version Plus Fulfillment Type and Location and Reference Numbers 28-31	Version 3 Plus Posting Store	Version 4 Plus Consignment Rate/ Cost	Version 5 Plus Inventory Identifier Type/ Inventory Id
Consignment Rate/Cost	N	N	N	N	N	Y	Y
Inventory Identifier Type	N	N	N	N	N	N	Y
Inventory Id	N	N	N	N	N	N	Y

Table 6-78 File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	File Type Definition	Char(4)	RTLGL	Identifies file as Oracle Retail TLOG.	Y	Left/Blank
	File Create Date	Char(14)	Create date	Date and time file was written by external system (YYYYMMDD HHMMSS).	Y	Left/None
	Business Date	Char(8)	Business Date to process	Business date of transactions (YYYYMMDD)	Y	Left/None
	Location Number	Char(10)	Specified by external system	Store or warehouse identifier.	Y	Left/None

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
Transaction Header	Reference Number	Char(30)	Specified by external system	This may contain the Polling ID associated with the consolidated TLOG file or used for other purpose.	N	Left/Blank
	RTLOG Originating System	Char(3)	POS	Identifies the system the RTLOG file originated from. Valid values are OMS and POS.	Y	Left/None
	File Type Record Descriptor	Char(5)	Char(5) THEAD	Identifies file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Register	Char(5)	Transaction date	Till used at the store.	Y	Left/Blank
	Transaction Date	Char(14)	N/A	Date for the transactions that were processed at the POS (YYYYMMDD HHMMSS).	Y	Left/None
	Transaction Number	Number(10)	N/A	Transaction identifier. If sa_system_options, wkstation_tran_append_ind is Y, then the first 3 digits indicate the workstation ID and last 7 digits indicate the transaction number.	Y	Right/0
	Cashier	Char(10)	N/A	Cashier identifier.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Salesperson	Char(10)	N/A	Salesperson identifier.	N	Left/Blank
	Transaction Type	Char(6)	Refer to TRAT code_type for a list of valid types.	Transaction type.	Y	Left/Blank
	Sub-transaction type	Char(6)	Refer to TRAS code_type for a list of valid types.	Sub-transaction type. For sale, it can be employee, drive-off, and so on.	N	Left/Blank
	Orig_tran_no	Number(10)	N/A	Populated only for post-void transactions. Transaction number for the original transaction that will be cancelled.	N	Right/0
	Orig_reg_no	Char(5)	N/A	Populated only for post-void even exchange and return transactions. Register number from the original transaction	N	Left/Blank
	Reason Code	Char(6)	Refer to REAC code_type for a list of valid codes. If the transaction type is PAIDOU and the sub transaction type is MV or EV, than the valid codes come from the non_merch_code_head table.	Reason entered by the cashier for some transaction types. Required for Paid In and Paid out transaction types, but can also be used for voids, returns, and so on.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Vendor Number	Char(10)	N/A	Supplier ID for a merchandise vendor paid out transaction; partner ID for an expense vendor paid out transaction.	N	Left/Blank
	Vendor Invoice Number	Char(30)	N/A	Invoice number for a vendor paid out transaction.	N	Left/Blank
	Payment Reference Number	Char(16)	N/A	The reference number of the tender used for a vendor payout. This could be the money order number, check number, and so on.	N	Left/Blank
	Proof of Delivery Number	Char(30)	N/A	Proof of receipt number given by the vendor at the time of delivery. This field is populated for a vendor paid out transaction.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Reference Number 1	Char(30)	Na	Number associated with a particular transaction, for example, whether for a Store Conditions transaction. The SA_REFERENCE table defines what this field can contain for each transaction type.	N	Left/Blank
	Reference Number 2	Char(30)	N/A	Char(30)	N	Left/Blank
	Reference Number 3	Char(30)	N/A	Third generic reference number.	N	Left/Blank
	Reference Number 4	Char(30)	N/A	Fourth generic reference number.	N	Left/Blank
	Value Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of the value.	Y if Value is present.	Left/None
	Value	Number(20)	N/A	Value, with 4 implied decimal places. Populated by the retailer for TOTAL transaction, populated by Sales Audit for SALE and RETURN transactions.	Y if tran is a TOTAL	Right/0 when value is present. Blank when no value is sent.
	Banner id	Number(4)	N/A	Banner ID of the location.	Y	Right/0 when value is present. Blank when no value is sent

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Rounded Amount Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of rounded amount. Amount Sign is not used.	Y	Left/None
	Rounded Amount	Number(20)	N/A	Total rounded amount, with 4 implied decimal places. Rounded Amount is not used.	Y	Right/0 when RoundedAmount is present otherwise blank
	Rounded Off Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Rounded Off Sign is not used.	Y	Left/None
	Rounded Off Amount	Number(20)	N/A	Rounded off amount, with 4 implied decimal places. Rounded Off Amount is not used.	Y	Right/0 when RoundedAmount is present otherwise blank
	Credit Promotion Id	Char(10)	N/A	Credit Promotional ID.	Y	Left/None
	Reference Number 25	Char(30)	N/A	N/A	N	Left/Blank
	Reference Number 26	Char(30)	N/A	N/A	N	Left/Blank
	Reference Number 27	Char(30)	N/A	N/A	N	Left/Blank
	Transaction Processing System	Char(3)	Valid values are OMS and POS.	Contains the ID of the system that processed the transaction.	N	Left/None
	Reference Number 28	Char(30)		Generic Reference Number. It can be ignored if not needed based on the type of RTLOG being used.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Reference Number 29	Char(30)		Generic Reference Number. It can be ignored if not needed based on the type of RTLOG being used.	N	Left/Blank
	Reference Number 30	Char(30)		Generic Reference Number. It can be ignored if not needed based on the type of RTLOG being used.	N	Left/Blank
	Reference Number 31	Char(30)		Generic Reference Number. It can be ignored if not needed based on the type of RTLOG being used.	N	Left/Blank
Transaction Header Attribute	File Type Record Descriptor	Char(5)	THATT	Identifies file record type	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file.	Y	Right/0
	Attribute Type	Char(6)	Refer to 'SAHA' code_type for a list of valid types	Type of transaction header attribute	Y	Left/Blank
	Attribute Value	Char(120)		Value of transaction header attribute	Y	Left/Blank
Transaction Customer	File Type Record Descriptor	Char(5)	TCUST	Identifies the file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file	Y	Right/0

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
Customer Attribute	Customer ID	Char(16)	Customer identifier	The ID number of a customer.	Y	Left/Blank
	Customer Type ID	Char(6)	Refer to CIDT code_type for a list of valid types.	Customer ID type.	Y	Left/Blank
	Customer Name	Char(120)	N/A	Customer name.	N	Left/Blank
	Address 1	Char(240)	N/A	Customer address.	N	Left/Blank
	Address 2	Char(240)	N/A	Additional field for customer address.	N	Left/Blank
	City	Char(120)	N/A	City.	N	Left/Blank
	State	Char(12)	State identifier	State.	N	Left/Blank
	Zip Code	Char(30)	Zip identifier	Zip code.	N	Left/Blank
	Country	Char(3)	N/A	Country.	N	Left/Blank
	Home Phone	Char(20)	N/A	Telephone number at home.	N	Left/Blank
	Work Phone	Char(20)	N/A	Telephone number at work.	N	Left/Blank
	E-mail	Char(100)	N/A	E-mail address.	N	Left/Blank
	Birthdate	Char(8)	N/A	Date of birth. (YYYYMMDD)	N	Left/Blank
	File Type Record Descriptor	Char(5)	CATT	Identifies file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Attribute type	Char(6)	Refer to SACA code_type for a list of valid types.	Type of customer attribute	Y	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
Transaction Item	Attribute value	Char(6)	Refer to members of SACA code_type for a list of valid values.	Value of customer attribute.	Y	Left/Blank
	File Type Record Descriptor	Char(5)	TITEM	Identifies file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Item Status	Char(6)	Refer to SASI code_type for a list of valid codes.	Status of the item within the transaction. Valid values are: V - Void item S - Sold item R - Returned item ORI - Order Initiate ORC - Order Cancel ORD - Order Complete LIN - Layaway Initiate LCA - Layaway Cancel LCO - Layaway Complete ADJ - Appeasement/ Adjustment	Y	Left/Blank
	Item Type	Char(6)	Refer to SAIT code_type for a list of valid codes.	Identifies what type of item is transmitted.	Y	Left/Blank
	Item number type	Char(6)	Refer to UPCT code_type for a list of valid codes.	Identifies the type of item number if the item type is ITEM or REF	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Format ID	Char(1)	VPLU format ID	Used to interpret VPLU items.	N	Left/Blank
	Item	Char(25)	Item identifier	Identifies the merchandise item.	N	Left/Blank
	Reference Item	Char(25)	Item identifier	Identifies the sub-transaction level merchandise item.	N	Left/Blank
	Non-Merchandise Item	Char(25)	Item identifier	Item identifier Identifies a non-merchandise item.	N	Left/Blank
	Voucher	Char(25)	N/A	Gift certificate number.	N	Right/0
	Department	Number(4)	N/A	Identifies the department to which this item belongs. This is filled in by saimptlog.	N	Right/Blank
	Class	Number(4)	Class of the item	Class of item sold or returned. Not required from a retailer; populated by Sales Audit. This is filled in by saimptlog.	N	Right/Blank
	Subclass	Number(4)	Subclass of the item	Subclass of the item sold or returned. Not required from a retailer; populated by Sales Audit. This is filled in by saimptlog.	N	Right/Blank
	Quantity Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of the quantity	Y	Left/None

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Quantity	Number(12)	N/A	Number of items purchased, with 4 decimal places.	Y	Right/0
	Selling Unit of Measure	Char(4)	N/A	Unit of measure of the item's quantity.	Y	Left/None
	Unit Retail	Number(20)	N/A	Unit retail, with 4 implied decimal places.	Y	Right/0
	Override Reason	Char(6)	Refer to ORRC code_type for a list of valid codes.	This column is populated when an item's price has been overridden at the POS to define why it was overridden.	Y if unit retail was manually entered	Left/Blank
	Original Unit Retail	Number(20)	N/A	Value, with 4 implied decimal places. This column is populated when the item's price was overridden at the POS and the item's original unit retail is known.	Y if unit retail was manually entered	Right/0
	Taxable Indicator	Char(1)	Refer to YSNO code_type for a list of valid codes.	Indicates whether or not item is taxable.	Y	Left/None
	Pump	Char(8)	N/A	Fuel pump identifier.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Reference Number 5	Char(30)	N/A	Number associated with a particular item within a transaction, for example, special order number. The sa_reference table defines what this field can contain for each transaction type.	N	Left/Blank
	Reference Number 6	Char(30)	N/A	Second generic reference number at the item level.	N	Left/Blank
	Reference Number 7	Char(30)	N/A	Third generic reference number at the item level.	N	Left/Blank
	Reference Number 8	Char(30)	N/A	Fourth generic reference number at the item level.	N	Left/Blank
	Item_swiped_ind	Char(1)	Refer to YSNO code_type for a list of valid codes	Indicates if the item was automatically entered into the POS system or if it had to be manually keyed.	Y	Left/None
	Return Reason Code	Char(6)	Refer to SARR code_type for a list of valid codes.	The reason an item was returned.	N	Left/Blank
	Salesperson	Char(10)	N/A	The salesperson who sold the item.	N	Left/Blank
	Expiration_date	Char(8)	N/A	Gift certificate expiration date (YYYYMMDD)	N	

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Drop Ship Ind	Char(1)	Refer to YSNO code type for a list of valid codes.	Indicates whether the item is part of a drop shipment.	Y	Left/None
	Uom_qty	Number(12)	N/A	Quantity of items purchased in the given UOM, with 4 decimal places.	Y	Right/0
	Catchweight_ind	Char(1)	Valid values are Y and N.	Identifies if the item is a catchweight item.		Left/None
	Selling item	Char(25)	Item identifier	Identifies the selling item.	N	Left/Blank
	Customer order line no	Number(6)	N/A	Identifies the customer order number.	N	Left/Blank
	Media id	Number(10)	N/A	Identifies the customer media ID.	N	Left/Blank
	Total Igtax Amount	Number(21)	N/A	Contains the Igtax amount.	N	Right/0
	Unique ID	Char(128)	N/A		N	Left/Blank
	Customer Order Number	Char(48)	N/A	Contains the customer order ID.	N	Left/None
	Customer Order Date	Char(14)	N/A	Contains the customer order date. Format is: YYYYMMDDH HMMSS Customer orders and layaways require customer order date.	N	Left/Blank
	Fulfillment Order Number	Char(48)	N/A	Contains the order ID of the fulfillment order.	N	Left/None

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	No Inventory Return	Char(1)	N/A	Indicates if there is an associated inventory with the return transaction with an External Customer Order sales type.	N	Left/Blank
	Sales Type	Char(1)	N/A	Indicates if the transaction is an In Store Customer Order, External Customer Order, or Regular Sale	N	Left/Blank
	Return Warehouse	Char(10)	N/A	Contains the ID of the physical warehouse to which the inventory is returned.	N	Left/Blank
	Return Disposition	Char(10)	N/A	Contains the return disposition of the returned items.	N	Left/Blank
	Original Store	Char(10)		Contains the original store.	N	Left/Blank
	Original Transaction No	Char(10)		Contains the original transaction no.	N	Left/Blank
	Fulfillment Loc Type	Char(2)	Refer to 'FLTP' code type for a list of valid types.	Contains the fulfillment order location type. It is needed only if the file is for an OMS transaction.	N	Left/Blank
	Fulfillment Loc	Number(10)		Fulfillment Location ID. It is needed only if the file is for an OMS transaction.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Posting Store	Number(10)		Contains the store at which the item sale/ return should be accounted for in case of cross-store sales happening at co-located stores. It is expected that this field will be populated only for items that are checked out at a different store from the one at which they are originally managed.	N	Left/Blank
	Consignment Rate	Number(12)		Consignment Rate with 4 implied decimal places.	N	Right/0
	Consignment Unit Cost	Number(20)		Consignment Unit Cost with 4 implied decimal places.	N	Right/0
	Inventory Identifier Type	Char(6)		Contains the inventory identifier type passed in Sales/Return transactions.	N	Left/Blank
	Inventory Id	Char(120)		Contains the inventory id passed in Sales/Return transactions.	N	Left/Blank
Transaction Item Attribute	File Type Record Descriptor	Char(5)	ITATT	Identifies file record type	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file.	Y	Right/0

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
Item Discount	Attribute Type	Char(6)	Refer to 'SAIA' code_type for a list of valid types	Type of item attribute	Y	Left/Blank
	Attribute Value	Char(120)		Value of item attribute	Y	Left/Blank
	File Type Record Descriptor	Char(5)	IDISC	Identifies the file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Merchandising Promotion Type	Char(6)	Refer to PRMT code_type for a list of valid types	The Merchandising promotion type.	Y	Left/Blank
	Discount Reference Number	Number(10)	N/A	Discount reference number associated with the discount type. For example, if the discount type is a promotion, this contains the promotion number.	N	Left/Blank
	Discount Type	Char(6)	Refer to SADT code_type for a list of valid types.	The type of discount within a promotion. This allows a retailer to further break down coupon discounts within the In-store promotion, for example.	N	Left/Blank
	Coupon Number	Char(40)	N/A	Number of a store coupon used as a discount.	Y if coupon	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Coupon Reference Number	Char(16)	N/A	Additional information about the coupon, usually contained in a second bar code on the coupon.	Y if coupon	Left/Blank
	Quantity Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of the quantity.	Y	Left/None
	Quantity	Number(12)	N/A	The quantity purchased for which the discount is applied, with 4 implied decimal places.	Y	Right/0
	Unit Discount Amount	Number(20)	N/A	Unit discount amount for this item, with 4 implied decimal places.	Y	Right/0
	Reference Number 13	Char(30)	N/A	Number associated with a particular transaction type at the discount level. The sa_reference table defines what this field can contain for each transaction type.	N	Left/Blank
	Reference Number 14	Char(30)	N/A	Second generic reference number at the discount level.	N	Left/Blank
	Reference Number 15	Char(30)	N/A	Third generic reference number at the discount level.	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Reference Number 16	Char(30)	N/A	Fourth generic reference number at the discount level.	N	Left/Blank
	Uom_qty	Number(12)	N/A	Quantity of items purchased in the given UOM with 4 decimal places.	Y	Right/0
	Catchweight_ind	Char(1)	Valid values are Y and N.	Identifies if the item is a catchweight item.		Left/None
	Promo component	Number(10)	N/A	If the discount is a promotion, this field contains the promotion component value associated with the promotion (discount reference number).	N	Left/Blank
Transaction Item Discount Attribute	File Type Record Descriptor	Char(5)	IDATT	Identifies file record type	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file.	Y	Right/0
	Attribute Type	Char(6)	Refer to 'SADA' code_type for a list of valid types	Type of transaction item discount attribute	Y	Left/Blank
	Attribute Value	Char(120)		Value of transaction item discount attribute	Y	Left/Blank
Item Tax	File Type Record Descriptor	Char(5)	IGTAX	Identifies the file record type	Y	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Tax Authority	Char(10)	N/A	A free-form text field. Any value can be used as a default string describing the authority levying the tax.	Y	Left/Blank
	Igtax Code	Char(6)	Refer to tax_code/ vat_code of tax_codes/ vat_codes tables.	IGtax code (tax code/VAT code) to represent whether it is a State, City, or some other tax code/VAT code.	Y	Left/Blank
	Igtax Rate	Number(20)	N/A	Igtax rate, with 4 implied decimal places.	Y	Right/0
	Igtax Amount Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of the Igtax amount.	Y	Left/None
	Igtax Amount	Number(21)	N/A	Total igtax amount for this item, with 5 implied decimal places.	Y	Right/0
	Reference Number 21	Char(30)	N/A	N/A	N	Left/None
	Reference Number 22	Char(30)	N/A	N/A	N	Left/None
	Reference Number 23	Char(30)	N/A	N/A	N	Left/None
	Reference Number 24	Char(30)	N/A	N/A	N	Left/None
	Tax Calculation Type	Char(6)	Refer to the 'GTTT' code type for the list of valid values.	Contains the tax calculation type.	N	Left/None

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
Transaction Item Tax Attribute	File Type Record Descriptor	Char(5)	IXATT	Identifies file record type	Y	Left/None
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file.	Y	Right/0
	Attribute Type	Char(6)	Refer to 'SAXA' code_type for a list of valid types	Type of transaction item tax attribute	Y	Left/None
	Attribute Value	Char(120)		Value of transaction item tax attribute	Y	Left/None
Transaction Tax	File Type Record Descriptor	Char(5)	TTAX	Identifies the file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Tax Code	Char(6)	Refer to TAXC code_type for as list of valid types.	Tax code (tax code/VAT code) to represent whether it is a State, City, or some other tax code/VAT code.	Y	Left/Blank
	Tax Sign	Char(1)	Refer to SIGN code_type for a list of valid codes	Sign of the tax amount.	Y	Left/None
	Tax Amount	Number(20)	N/A	Total Tax amount for this item, with 4 implied decimal places.	Y	Right/0
	Reference Number 17	Char(30)	N/A	N/A	N	Left/None
	Reference Number 18	Char(30)	N/A	N/A	N	Left/None
	Reference Number 19	Char(30)	N/A	N/A	N	Left/None

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Reference Number 20	Char(30)	N/A	N/A	N	Left/None
Transaction Tax Attribute	File Type Record Descriptor	Char(5)	TXATT	Identifies file record type	Y	Left/None
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file.	Y	Right/0
	Attribute Type	Char(6)	Refer to 'SAXA' code_type for a list of valid types	Type of transaction tax attribute	Y	Left/None
	Attribute Value	Char(120)		Value of transaction tax attribute	Y	Left/None
Transaction payment	File Type Record Descriptor	Char(5)	TPYMT	Identifies the file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Payment Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of the deposit amount.	Y	Left/None
	Payment Amount	Number(20)	N/A	Deposit amount paid, with 4 implied decimal places.	Y	Right/0
Transaction Tender	File Type Record Descriptor	Char(5)	TTEND	Identifies the file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Tender Type Group	Char(6)	Refer to TENT code_type for as list of valid types	High-level grouping of tender types.	Y	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Tender Type ID	Number(6)	Refer to the pos_tender_type_head table for as list of valid types.	Low-level grouping of tender types.	Y	Left/Blank
	Tender Sign	Char(1)	Refer to SIGN code_type for a list of valid codes.	Sign of the value.	Y	Left/None
	Tender Amount	Number(20)	N/A	Amount paid with this tender in the transaction, with 4 implied decimal places.	Y	Right/0

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Cc_no	Char(40)	N/A	Credit card number. Merchandise is not a PCI compliant system. Full credit card numbers are not allowed in Merchandising . The value sent in the RTLOG should be masked so that it contains no more than the first 6 digits and last 4 digits in clear text. The remaining digits should be masked using the character defined in sa_system_options.cc_no_mask_char. If more than the first 6 and last 4 characters exist in any records, the transaction file will be rejected. Alternatively, the credit card number field can be left fully blank.	N	Left/Blank
	Cc_auth_no	Char(16)	N/A	Authorization number for a credit card.	N	Left/Blank
	cc authorization source	Char(6)	Refer to CCAS code_type for as list of valid types.	N/A	N	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	cc cardholder verification	Char(6)	Refer to CCVF code_type for as list of valid types	N/A	N	Left/Blank
	cc expiration date	Char(8)	N/A	YYYYMMDD	N	Left/Blank
	cc entry mode	Char(6)	Refer to CCEM code_type for as list of valid types.	Indicates whether the credit card was swiped, thus automatically entered, or manually keyed.	N	Left/Blank
	cc terminal id	Char(5)	N/A	Terminal number from which the transaction was sent.	N	Left/Blank
	cc special condition	Char(6)	Refer to CCSC code_type for as list of valid types.	N/A	N	Left/Blank
	cc token	Char(40)	N/A	Holds unique token when the tender type used is credit, debit card, PayPal, Fonacot or Others.	N	Left/Blank
	Voucher_no	Char(25)	N/A	Gift certificate or credit voucher serial number. Voucher number needs to be included If a voucher is voided from a transaction.	Y if voucher	Right/0
	Coupon Number	Char(40)	N/A	Number of a manufacturer's coupon used as a tender.	Y if coupon	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Coupon Reference Number	Char(16)	N/A	Additional information about the coupon, usually contained in a second bar code on the coupon.	Y if coupon	Left/Blank
	Cheque Account Number	Char(30)	N/A	Account number of the cheque. The value sent in the RTLOG is masked.	N	Left/Blank
	Cheque Number	Number(10)	N/A	Check number.	Required for the tender type CHECK	Right/0
	Identification Method	Char(6)	Refer to IDMH code_type for list of valid types.	Identification Method (such as a driver's license number or photo credit card).	N	Left/Blank
	Identification Id	Char(40)	N/A	Identification ID (license ID or photo card number).	N	Left/Blank
	Original Currency	Char(3)	Refer to the CURRENCIES table for valid currency codes.	The original currency with which the customer made the payment.	N	Left/Blank
	Original Currency Amount	Number(20)	N/A	Amount paid with this tender in the original currency, with 4 implied decimal places.	N	Right/0

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	Reference No 9	Char(30)	N/A	Number associated with a particular transaction type at the tender level. The sa_reference table defines what this field can contain for each transaction type.	N	Left/Blank
	Reference No 10	Char(30)	N/A	Second generic reference number at the tender level.	N	Left/Blank
	Reference No 11	Char(30)	N/A	Third generic reference number at the tender level.	N	Left/Blank
	Reference No 12	Char(30)	N/A	Fourth generic reference number at the tender level.	N	Left/Blank
Transaction Tender Attribute	File Type Record Descriptor	Char(5)	TTATT	Identifies file record type	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file.	Y	Right/0
	Attribute Type	Char(6)	Refer to 'SATA' code_type for a list of valid types	Type of transaction tender attribute	Y	Left/Blank
	Attribute Value	Char(120)		Value of transaction tender attribute	Y	Left/Blank
Transaction Trailer	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type.	Y	Left/Blank

Table 6-78 (Cont.) File Name: rtlog

Record Name	Field Name	Field Type	Default Value	Description	Required?	Justification/ Padding
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	Transaction Record Counter	Number(10)	N/A	Number of records processed in the current transaction (only those records between transaction head and tail).	N/A	N/A
File Trailer	File Type Record Descriptor	Char(5)	FTAIL	Identifies the file record type.	Y	Left/Blank
	File Line Identifier	Number(10)	Specified by external system	ID of the current line being processed by input file.	Y	Right/0
	File Record Counter	Number(10)	N/A	Number of transactions processed in the current file (only the records between the file head and tail).	Y	Right/0

The RTLOG file is imported into the Sales Audit tables after validation by the batch program saimptlog. This section describes the requirements and validations performed on the records.

Common Requirements/Validations

This section details the common requirements and validations performed on all transactions. The following sections describe the specific requirements of each type of transaction. If a transaction is not mentioned, it does not have specific requirements.

Table 6-79 Record Type Requirements

Transaction Type	Includes item records?	Includes tender records?	Includes tax records? IG TAX?	Includes customer records?
OPEN	No	No	No	No
NOSALE	No	Optional	No	No
VOID	Optional	Optional	Optional	Optional
PVOID	No	No	No	No

Table 6-79 (Cont.) Record Type Requirements

Transaction Type	Includes item records?	Includes tender records?	Includes tax records? IG TAX?	Includes customer records?
SALE	Optional	Yes	Optional	Optional
RETURN	Yes	Yes	Optional	Optional
EEXCH	Yes	No	Optional	Optional
PAIDIN	No	Yes	No	No
PAIDOU	No	Yes	No	No
PULL	No	Yes	No	No
LOAN	No	Yes	No	No
COND	No	No	No	No
CLOSE	No	No	No	No
TOTAL	No	No	No	No
REFUND	This transaction is not sent through the RTLOG. It is entered at the HQ level. The TITEM and TCUST records are optional. The TTEND record is required. A TTAX record should not be included if IG TAX appears in a transaction if originating system is POS. IG TAX is an item-level tax and TTAX is a transaction-level tax. Either IG TAX or TTAX can be used if originating system is POS, but not both. If originating system is OMS, both IG TAX and TTAX can appear but only the one matching the store's tax type will be processed, the other record will be ignored.			
METER	Yes	No	No	No
PUMPT	Yes	No	No	No
TANKDP	Yes	No	No	No
TERM	TERM records are created by saimptlog and then loaded into the database. They do not come from the RTLOG file. They require one TITEM, one TTEND, one TTAX, one TCUST record, and one CATT record, IG TAX, and one TPYMT which is newly coming up.			
DCLOSE	No	No	No	No
SPLORD	Optional	Yes	Optional	Optional
REOPEN	No	No	No	No
PREPAY	No	Yes	No	Optional

Requirements per Record Type

Table 6-80 Requirements per Record Type

Record Type	Requirements
IDISC	IDISC records must immediately follow their associated TITEM record. IDISC records should be after the ITATT record if it exists.
IGTAX	IGTAX will immediately follow TITEM or ITATT record (if it is present), if discount records are not present. Otherwise it should follow the IDISC or IDATT record (if it is present). Even if IGTAX is coming prior to IDISC, it will be processed, but for maintaining proper format, Sales Audit expects it to come after IDISC.

Table 6-80 (Cont.) Requirements per Record Type

Record Type	Requirements
TTAX	Either this record or IGTX should appear in the transaction. IGTX and TTAX cannot be both used at the same time if originating system is POS. If originating system is OMS, both IGTX and TTAX can appear but only the one matching the store's tax type will be processed, the other record will be ignored.
TPYMT	This record should be right before the TTEND record. It contains the deposit amount for pickup/delivery/layaway orders.
CATT	CATT records must immediately follow their associated TCUST record.
THATT	THATT records must immediately follow their associated THEAD record.
ITATT	ITATT records must immediately follow their associated TITEM record.
IDATT	IDATT records must immediately follow their associated IDISC record.
IXATT	IXATT records must immediately follow their associated IGTX record.
TXATT	TXATT records must immediately follow their associated TTAX record.
TTATT	TTATT records must immediately follow their associated TTEND record.

Code Type Validations

Table 6-81 Code Type Validations

Record Name	Field Name	Code Type
Transaction Header	Transaction Type	TRAT
	Sub-transaction Type	TRAS
	Reason Code	REAC or values from non_merch_code_head if the transaction type is PAIDOU and the sub-transaction type is MV or EV.
	Value Sign	SIGN
	Vender No	If the transaction type is PAIDOU and the sub-transaction type is MV, this field is validated against the supplier table. If the transaction type is PAIDOU and the sub-transaction type is EV, this field is validated against the partner table.
	Transaction Processing System	TSYS
Transaction Header Attribute	Attribute Type	SAHA
Transaction Item	Item Type	SAIT
	Item Status	SASI
	Item Number Type	UPCT
	Quantity Sign	SIGN
	Taxable Indicator	YSNO
	Price Override Reason Code	ORRC
	Item Swiped Indicator	YSNO
	Sales Type	SASY

Table 6-81 (Cont.) Code Type Validations

Record Name	Field Name	Code Type
	Return Disposition	INV_STATUS_CODES table
	No Inventory Return	YSNO
	Return Reason Code	SARR
	Fulfillment Loc Type	FLTP
Transaction Item Attribute	Attribute Type	SAIA
Item Discount	RMS Promotion Type	PRMT
	Discount Type	SADT
	Quantity Sign	SIGN
Item Discount Attribute	Attribute Type	SADA
Transaction Customer	Customer ID Type	CIDT
Customer Attribute	Attribute Type	SACA
	Attribute value	Code types from the codes in SACA.
Item Tax	Tax Calculation Type	GTTT
	Tax Code	TAXC from the CODE_DETAIL table or VATC from the VAT_CODES table
Item Tax Attribute	Attribute Type	SAXA
Transaction Tax	Tax code	TAXC from the CODE_DETAIL table or VATC from the VAT_CODES table.
	Tax sign	SIGN
Transaction Tax Attribute	Attribute Type	SAXA
Transaction Payment	Payment (Deposit Amount) Sign	SIGN
Transaction Tender	Transaction Tender Tender Type Group	TENT
	Tender Sign	SIGN
	Tender Type ID	Pos_tender_type_head table
	CC Authorization Source	CCAS
	CC Cardholder Verification	CCVF
	CC Entry Mode	CCEM
	CC Special Condition	CCSC
	Attribute Type	SATA

The following dates are validated: Business Date, Transaction Date, and Expiration Date. Also, saimptlog accepts only business dates that are within the PERIOD.VDATE minus the SA_SYSTEM_OPTIONS.DAYS_POST_SALE value.

The store number is validated against the STORE table. Numeric fields are checked for non-numeric characters.

For transactions of type SALE, RETURN, and EEXCH, saimptlog checks whether a transaction is in balance. With the introduction of the Item level tax and Payment amount lines, the balancing logic has been changed as below. Also with introduction of handling VAT/TAX, the logic of balancing has been modified as below.

- When TAX is on in the system (system_options.default_tax_type equals SALES):
 Transaction Items (Unit Retail * Unit Retail Sign * Quantity) of items which are on Regular Sale, Return, or EEXCH
 + Item Discounts (Unit Discount Amount * Unit Discount Sign * Quantity) of items which are on Regular Sale, Return, or EEXCH
 + Item Level Tax (Total Igtax Amount) of items which are on Regular Sale, Return, or EEXCH
 + Transaction Tax (Tax Amount * Tax Sign)
 + Transaction payment (Payment Amount * Payment Sign)
 equals Transaction Tenders (Tender Amount * Tender Sign)
 saimptlog will populate the Value field (on THEAD) with the transaction's sales value (item value minus discount value plus tax value) from the preceding calculation if it was not provided in the RTLOG. The following change is made in the sale total balancing: Value field in THEAD will be: (item value - discount value + tax value) for items which are on Regular Sale, Return, or EEXCH + payment value.

Note

If this Value field is being used in creating some totals, then accordingly, these totals needs to be modified to accommodate the extra amount coming in.

- When VAT is on in the system (system_options.default_tax_type in GTAX, SVAT, GTS), look for the store level VAT indicator, which tells whether the unit retail is inclusive or exclusive of VAT. The logic of balancing will vary:
 Transaction Items (Unit Retail * Unit Retail Sign * Quantity) of items which are on Regular Sale, Return, or EEXCH
 + Item Discounts (Unit Discount Amount * Unit Discount Sign * Quantity) of items which are on Regular Sale, Return, or EEXCH
 + Item Level Tax (Total Igtax Amount) of items which are on Regular Sale, Return, or EEXCH (when VAT is off at the item level).
 + Transaction Tax (Tax Amount * Tax Sign)
 + Transaction Payment (Payment Amount * Payment Sign)
 equals Transaction Tenders (Tender Amount * Tender Sign)

Vouchers are treated as follows:

- If an item sold is as a gift certificate (Transaction Item, Voucher field has a value), the issued information is written to the SA_VOUCHER table.
- If the Transaction Type is RETURN, and the Transaction Tender Type Group is voucher (VOUCH), the issued information is written to the SA_VOUCHER table.
- If the Transaction Type is SALE and the Transaction Tender Type Group is a voucher (VOUCH), the redeemed information is written to the SA_VOUCHER table.

- When a gift certificate is sold, the customer information should always be included. A receiving customer name value should be populated in the ref_no5 field, receiving customer state value should be populated in the ref_no6 field, and receiving customer country should be populated in the ref_no7 field. These reference fields can be changed by updating the sa_reference table, but the code needs to be modified as well. The expiration date is put in the expiration_date field in the TITEM record.

Other validations and points to consider:

- The salesperson in the TITEM record takes precedence over the salesperson in the THEAD record.
- If an item sold is a sub-transaction (REF) item (Transaction Item, reference item field has a value and item does not), it will be converted to the corresponding transaction level item (ITEM).
- If an item sold is an ITEM (Transaction Item, item field has a value), it will be validated against the Merchandising item tables.
- The corresponding Department, Class, Subclass, and Taxable Indicator will be selected from the Merchandising tables and populated for an item.

The balancing level determines whether the register or the cashier fields are required:

- If the balancing level is R (register), the register field on the THEAD must be populated.
- If the balancing level is C (cashier), the cashier field on the THEAD must be populated.
- If the balancing level is S (store), neither field is required to be populated.
- The tax_ind and the item_swiped_ind fields can only accept Y or N values. If an invalid value is passed through the RTLOG, an error will be flagged and the value will be defaulted to Y.

Transaction of Type SALE

A transaction of type SALE is generated whenever an item is sold. If a sale is for an employee, the sub-transaction type is EMP. If it is a drive-off sale, when someone drives off with unpaid gas, the sub-transaction type is DRIVEO. A special type of sale is an odd exchange, sub-transaction type EXCH, where items are sold and returned in the same transaction. If the net value of the exchange is positive, then it is a sale. If the net value is negative, it is a return.

Requirements per record type (other than what is described in the preceding Layout section):

Table 6-82 Requirements per Record Type

Record Type	Requirements
THEAD	N/A

Table 6-82 (Cont.) Requirements per Record Type

Record Type	Requirements
TITEM	- Item Status is a required field; it determines whether the item is Sold (S), Returned (R), or Voided (V). If the item status is S, the quantity sign is expected to be P. If the item status is R, the quantity sign is expected to be N. Also, if the item status is ORI, LIN, ORD, or LCO, the quantity sign should be P. In the case of ORC or LCA, it should be N. Item status ADJ is to support appeasement transactions from OMS, where a customer is partially refunded for their original purchase based on an agreed-to price adjustment, or other reasons. - If the item status is V, the quantity sign is the reverse of the quantity sign of the voided item. That is, if an item with status S is voided, the quantity sign would be N. Furthermore, the sum of the quantities being voided cannot exceed the sum of the quantities that are Sold or Returned. Note: Neither of the two validations are performed by saimptlog, but an audit rule could be created to check this. - The following item statuses are used for handling items on customer order layaway: ORI - Order Initiate ORD - Order Complete ORC - Order Cancel LIN - Layaway Initiate LCA - Layaway Cancel LCO - Layaway Complete - In a typical sale, the items all have a status of S. In the case of an odd exchange, some items will have a status of R. - In a typical return, the items all have a status of R. In the case of an odd exchange, some items will have a status of S. - If an item has status R, then the Return Reason Code field may be populated. If it is, it will be validated against code type SARR. Also, it is better to capture the Return Reason Code in the case of items on ORC or LCA, but it is not mandatory. No validation is kept for these new item statuses for checking of SARR. - If the price of an item is overridden, the Override Reason and Original Unit Retail fields must be populated.
IDISC	- The Merchandising Promotion Type field must always be populated with values of code type PRMT. - The Promotion field is validated, when a value is passed, against the promhead table. - If the promotion is In Store (code 1004), the Discount Type field must be populated with values of code type SADT. - The Discount Reference Number is a promotion number which is of status A, E, or M. - If the Discount Type is SCoup for Store Coupon, the Coupon Number field must be populated. The Coupon Reference Number field is optional.

Table 6-82 (Cont.) Requirements per Record Type

Record Type	Requirements
IGTAX	- The IGTAX_CODE field must always be populated depending on the system's default tax type. For a default tax type of SALES, this field will be populated with values of code_type TAXC. For a default tax type of SVAT, GTAX, or GTS, this field will be populated with VATC (vat_code from vat_codes table). IGTAX is an Item-level tax. - The TAX_AUTHORITY field must always be populated. This is a free form text and any value can be used as a default string describing the authority levying the tax. - If Item Status is ADJ (appeasement transaction), the IGTAX Tax Code from OMS is TOTAX.
TTAX	The TAX_CODE field must always be populated depending on the system's default tax type. For a default tax type of SALES, this field will be populated with values of code_type TAXC. For a default tax type of GTAX, SVAT, or GTS, this field will be populated with VATC (vat_code from vat_codes table). TTAX is a Transaction-level tax.
TPYMT	Payment (Deposit amount) sign and Payment (Deposit) amount fields are necessary if this line is appearing. Basically, this is the accumulation of various items being considered in one transaction, which are on pick up/delivery/lay away.
TTEND	If the tender type group is COUPON, the Coupon Number field must be populated. The Coupon Reference Number field is optional.

Meaning of reference number fields:

Note

The meaning of these reference number fields may be changed through the sa_reference table. The transaction type SPLORD is the same as SALE, but the inventory will not be reserved for the orders at its line level.

Table 6-83 Meaning of Reference Number Fields

Transaction Type	Sub-transaction Type	Item Type	Tender Type Group	Reference Number Field	Meaning of Reference Field	Req?
SALE	N/A	N/A	N/A	1	Speed Sale Number	Y
SALE	N/A	GCN	N/A	5	Recipient Name	N
SALE	N/A	GCN	N/A	6	Recipient State	N
SALE	N/A	GCN	N/A	7	Recipient Country	N
SALE	N/A	N/A	CHECK	9	Check Number	N
SALE	N/A	N/A	CHECK	10	Driver's License Number	N

Table 6-83 (Cont.) Meaning of Reference Number Fields

Transaction Type	Sub-transaction Type	Item Type	Tender Type Group	Reference Number Field	Meaning of Reference Field	Req?
SALE	N/A	N/A	CHECK	11	Credit Card Number	N
SALE	DRIVEO	N/A	N/A	1	Incident Number	Y
SALE	EMP	N/A	N/A	3	Employee Number of the employee receiving the goods.	N

Table 6-84 Expected Values for Sign Fields

TRANSACTION TYPE	TITEM.Quantity Sign	TEND.Tender Sign	TTAX.Tax Sign	IDISC.Quantity Sign
SALE	P if item is sold; N if item is returned; reverse of original item if item is voided.	P	P	P if item is sold; N if item is returned; reverse of original item if item is voided.
SALE	P if item is on ORI, LIN, ORD, or LCO; N if item is on ORC or LCA.	P	P	P if item is on ORI, LIN, ORD, or LCO; N if item is on ORC or LCA.

Transaction of Type PVOID

This transaction is generated at the register when another transaction is being post voided. The orig_tran_no and orig_reg_no fields must be populated with the appropriate information for the transaction being post voided. The PVOID transaction must be associated with the same store day as the original transaction. If the PVOID needs to be generated after the store day is closed, the transaction needs to be created using the forms.

Transaction of type RETURN

This transaction is generated when a customer returns an item.

This type of transaction has similar record type requirements as a SALE transaction.

Meaning of reference number fields:

Note

The assumption is that new item statuses will not come under transaction type RETURN.

If a customer wants to return the items (ORI, LIN), these will come under SALE but with item statuses as ORC or LCA.

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-85 Meaning of Reference Number Fields

Transaction Type	Sub-transaction Type	Reference Number Field	Meaning of Reference Field	Req?
RETURN	N/A	1	Receipt Indicator (Y/N)	Y
RETURN	N/A	2	Refund Reference Number	N
RETURN	EMP	3	Employee Number of the employee returning the goods.	N

Table 6-86 Expected Values for Sign Fields

TRANSACTION TYPE	TITEM.Quantity Sign	TEND.Tender Sign	TTAX.Tax Sign	IDISC.Quantity Sign
RETURN	P if item is sold; N if item is returned; reverse of original item if item is voided.	N	N	P if item is sold; N if item is returned; reverse of original item if item is voided

Transaction of type SPLORD

This transaction is generated when a customer picks up an item, which is not in stock. The item status can be ORI, ORC, or ORD. (Order Initiate, Order Cancel, or Order Complete).

Transaction of type EEXCH

This transaction is generated when there is an even exchange.

This type of transaction has similar record type requirements as a SALE transaction.

It is expected that the number of items returned equals the number of items sold. However, this validation is not performed by saimptlog. An audit rule could be created for this. Saimptlog only expects that there would be at least two item records.

No tender changes hands in this transaction.

Meaning of reference number fields:

Note

The items, which are on customer order or layaway, should not be come under this transaction type.

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-87 Meaning of Reference Number Fields

Transaction Type	Sub-transaction Type	Reference Number Field	Meaning of Reference Field	Req?
EEXCH	N/A	1	Receipt Indicator (Y/N)	Y
EEXCH	EMP	3	Employee Number of the employee exchanging the goods.	N

Transaction of type PAIDIN

This type of transaction has only one TTEND record.

A reason code is required.

Meaning of reference number fields:

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-88 Meaning of Reference Number Fields

Reason Code	Reference Number Column	Meaning	Req?
NSF	1	NFS Check Credit Number	N
ACCT	1	Account Number	N

Transaction Type PAIDOU

This type of transaction has only one TTEND record.

A reason code is required (code type REAC). If the sub-transaction type is EV or MV, the reason code comes from the non_merch_codes_head table.

If the sub-transaction type is EV or MV, then at least one field among the vendor number, vendor invoice number, payment reference number, and proof of delivery number fields should be populated.

If the sub-transaction type is EV, the vendor number comes from the partner table. If the sub-transaction type is MV, the vendor number comes from the supplier table.

Meaning of reference number fields:

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-89 Meaning of Reference Number Fields

Sub Transaction Type	Reason Code	Reference Number Column	Meaning	Req?
EV	N/A	2	Personal ID Number	N
EV	N/A	3	Routing Number	N
EV	N/A	4	Account Number	N
NA	PAYRL	1	Money Order Number	N
NA	PAYRL	2	Employee Number	N
NA	INC	1	Incident Number	N

Transaction of Type PULL

This transaction is generated when cash is withdrawn from the register.

This type of transaction has only one TTEND record.

Expected values for sign fields

Table 6-90 Expected Values for Sign Fields

TRANSACTION TYPE	TITEM.Quantity Sign	TEND.Tender Sign	TTAX.Tax Sign	IDISC.Quantity Sign
PULL	N/A	N	N/A	N/A

Transaction of Type LOAN

This transaction is generated when cash is added to the register.

This type of transaction has only one TTEND record.

Expected values for sign fields:

Table 6-91 Expected Values for Sign Fields

TRANSACTION TYPE	TITEM.Quantity Sign	TEND.Tender Sign	TTAX.Tax Sign	IDISC.Quantity Sign
LOAN	N/A	P	N/A	N/A

Transaction Type Cond

This transaction records the condition at the store when it opens. There can be at most one COND record containing weather information and at most one COND record containing temperature information. Both of these pieces of information may be in the same COND

record. There may be any number of COND records containing traffic and construction information.

This type of transaction does not have TITEM, IDISC, IGTX, TTAX, TPYMT, or TTEND records

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-92 Meaning of Reference Number Fields

Reference Number Column	Meaning	Req?
1	Weather - code type WEAT	N
2	Temperature - a signed 3 digit number.	N
3	Traffic - code_type TRAF	N
4	Construction - code_type CONS	N

Transaction of Type TOTAL

This transaction records the totals that are reported by the POS and OMS. The value field must be populated. Some systems generate only one transaction number for all totals. In order to avoid duplicate errors being reported, only one total transaction can have a transaction number and the subsequent ones can have blank transaction numbers. In other words, a TOTAL transaction is not required to have a transaction number.

This type of transaction does not have TITEM, IDISC, IGTX, TTAX, TPYMT, TTEND records.

Transaction of Type METER

This transaction is generated when a meter reading of a fuel pump is taken.

This type of transaction has only TITEM records.

Meaning of reference number fields:

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-93 Meaning of Reference Number Fields

Reference Number Column	Meaning	Req?
1	Reading Type: (A for adjustment, S for shift change, P for price change, or C for store close)	Y
5	Opening Meter Readings	Y
6	Closing Meter Reading	Y

Table 6-93 (Cont.) Meaning of Reference Number Fields

Reference Number Column	Meaning	Req?
7	If the reading type is P for price change, the old unit retail should be placed here. Decimal places are required.	Y
8	Closing Meter Value	Y

Transaction of Type PUMPT

This transaction is generated when a pump test is performed. This type of transaction has only TITEM records.

Transactions of Type TANKDP

This transaction is generated when a tank dip measurement is taken.

This type of transaction has only TITEM records.

Meaning of reference number fields:

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Table 6-94 Meaning of Reference Number Fields

Reference Number Column	Meaning of Reference Field	Req?
1	Tank identifier	Y
5	Dip Type (FUEL, WATER, and so on)	Y
6	Dip Height Major (decimal places required)	Y
7	Dip Height Minor (decimal places required)	Y

Transaction of Type DCLOSE

This transaction is generated when the day closed. The transaction number for this type of transaction has to be blank.

Note

Vouchers are minimally handled by saimptlog. Voucher information is written to the savouch file which is passed to the program savouch.pc.

- A voucher will appear on the TITEM record only if it was sold. When saimptlog encounters a SALE transaction with a voucher, it writes the voucher to the savouch file as an I for Issued voucher.

- A voucher will be issued when it appears on the TTEND record of transactions of type RETURN and PAIDOU. In other words, saimptlog will write it to the savouch file with status I.
- A voucher will be redeemed when it appears on the TTEND record of transactions of type SALE and PAIDIN. In other words, saimptlog will write it to the savouch file with status R.

Vouchers may not be returned. However, a transaction of type PAIDOU may be generated when the customer exchanges a voucher for another form of tender.

Transaction of Type REOPEN

This transaction is generated when a store day which was closed needs to be reopened to process additional transactions. Transaction number for this type of transaction has to be blank.

Transaction of Type OTHER

This transaction is a generic transaction type to support Micros Xstore integration. This will identify all the other transaction types that are not currently supported. This type of transaction has only THEAD and TTAIL records.

Note

Sales Audit typically doesn't allow transactions to exist outside of a store day or to be uploaded against a closed store day. To allow for exception cases where the store might perform internal processing related checks (such as clock in, training, and so on) outside the purview of a store day, flexibility has been provided to allow for only transactions with TRAN_TYPE and SUB_TRAN_TYPE of TYPE='OTHER' to be uploaded against closed store days. Upload of such transactions does not impact the store days accounting transactions and will automatically update the count of the files/transactions uploaded.

Transaction of Type PREPAY

This type of transaction is generated when a liability advance is accepted by the business without the requirement to capture item details. It carries THEAD, TTEND and an optional TCUST record. Optional Custom Attributes associated with these records lines (that is, THATT, TTATT, CATT) are also supported.

Meaning of reference number fields (When integrated from OMS):

Note

The meaning of these reference number fields may be changed through the sa_reference table.

Reference Number Column	Meaning	Req?
1	Customer Order Number	N

Design Assumptions

Table 6-95 Sales Audit Valid Transaction Type

Transaction Code	Transaction Type
OPEN	Open
CLOSE	Close
COND	Daily Store Conditions
DCLOSE	Day close indicator
LOAN	Loan
METER	Meter Reading for Fuel
NOSALE	No Sale
PAIDIN	Paid In
PAIDOU	Paid Out
PULL	Pull
PUMPT	Pump Test for Fuel
PVOID	Post Void (A transaction that was rung later into the register to void something that occurred earlier at the same store/day. A post void updates the original transaction's sub-transaction type.)
REFUND	Return of customer's original check.
RETURN	Return
SALE	Sale
TANKDP	Tank Dip
TOTAL	POS generated totals
EEXCH	Even exchange
VOID	Void (aborted transaction)
OTHER	Others
REOPEN	Reopen Store Day from POS
PREPAY	Prepayment

DCLOSE Transaction Type

When the retailer is sending only one file to the system, SAIMPTLOG.PC marks the store day record in the Sales Audit import log as partially or fully loaded in the database by looking for a transaction type of DCLOSE. However, if the retailer is sending more than one file (as in, for example, a trickle polling situation), the retailer can specify the number of files that the system should expect in combination with the DCLOSE transaction type. This ensures that the system receives all of the files, even if the DCLOSE transaction type is, for some reason, received before the final file.

For example, if 24 files are expected over a given amount of time, and the file with the DCLOSE transaction type is, for some reason, sent before the 24th file, the Merchandising system waits until the last file arrives before marking the store day record as partially or fully loaded in the database.

The import process is completed after SAIMPTLOGFIN.PC has updated the store, data, and audit status of each store day record.

The Reopen Transaction Type

When the retailer is sending transaction of type of REOPEN for store and business day system should expect REOPEN as first transaction in the file before any additional transactions.

When secondary DCLOSE transaction is sent after REOPEN transaction type system should expect count of files since the prior DLCLOSE transaction (not the full count for store/day).

SAIMPTLOGFIN.PC batch program would sum up the file counts in case of multiple DCLOSE transactions for the store day and compare against the files loaded in Sales Audit and update the store, data and audit status.

Import Total Value Adjustments From External Systems (saimpadj)

Module Name	saimpadj.pc
Description	Import Total Value Adjustments From External Systems to Sales Audit
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA07
Wrapper Script	rmswrap_in_rej.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This module posts external system adjustments to the Sales Audit total value table.

The sales audit adjustments are passed to the module in an external file.

Records that fail necessary validations would be written to the reject file. The input and reject file names are passed as arguments.

Restart/Recovery

Restart/recovery logic for file-based processing is used. The logical unit of work for this module is a parameterized number defined in the restart tables.

Record level locking is done on sa_store_day before updating.

I/O Specification

Integration Type	Upload to Sales Audit
File Name	Determined by runtime parameters.
Integration Contract	IntCon000047

Input File Layout

Table 6-96 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type (the beginning of the input file).
	File Line Identifier	Number(10)	Sequential number	ID of the current line being read from input file.
	File head descriptor	Char(4)	IMPA	Describes file line type.
	Current date	Char(14)	N/A	File date in YYYYMMDDHH24MISS format.
FDETL	File Type Record Descriptor	Char(5)	FDETL	Identifies the file record type to upload a new deal header.
	File Line Identifier	Number(10)	Sequential number	ID of the current line being read from input file.
	Data source	Char(6)	N/A	Name of the external system that produced the file.
	New value sign	Char(1)	N/A	Sign(+/-) for the new value.
	New Value	Number(20)	N/A	Value for the total entered by Headquarters user*10000 (4 implied decimal places).
	Total seq no	Number(20)	N/A	Identifies the unique result set for this total ID, total revision, or store/day. Balancing group and index values.
	Store	Number(10)	N/A	Store number for a store/day combination.
	Business Date	Char(8)	N/A	Date for store/day combination.
	Total id	Char(10)	N/A	ID to uniquely identify the total.
	Ref no 1	Char(30)	N/A	The first reference value based by which the total is grouped.
	Ref no 2	Char(30)	N/A	The second reference value based by which the total is grouped.
	Ref no 3	Char(30)	N/A	The third reference value based by which the total is grouped.
FTAIL	File Type record descriptor	Char(5)	FTAIL	Identifies the file record type (the end of the input file).
	File Line Identifier	Number(10)	Sequential number	ID of the current line being read from input file.
	File Record Counter	Number(10)	Sequential number	Number of records/transactions in the current file (only records between head and tail).

Design Assumptions

N/A

Sales Audit Voucher Upload (savouch)

Module Name	savouch.pc
Description	Sales Audit Voucher Upload
Functional Area	Oracle Retail Sales Audit
Module Type	Integration
Module Technology	ProC
Catalog ID	RSA08
Wrapper Script	batch_savouch.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

Because gift certificates can enter the Sales Audit system as either items or tender, processing must be done to match up the sales and redemptions. This module is used to aggregate gift certificate and voucher records. It compares records in the input files to the database. If a record for the voucher does not exist on the database, the record is inserted. If the voucher already exists on the database, the record should be updated with the appropriate information. The voucher details are updated to SA_VOUCHER table.

Some retailers assign gift certificates to a given store, which means that before a gift certificate is sold at a store, it is assigned to a given store. When a retailer assigns a gift certificate to a given store, a record is written to the database. When the gift certificate is then sold by the store and redeemed by the consumer, this existing record must be updated to include the sale and redemption information. Some retailers choose not to assign gift certificates and instead simply sell gift certificates. In that case, the record will be inserted into the database when the gift certificate is sold and then updated when the gift certificate is redeemed.

Restart/Recovery

Restart/recovery logic for file-based processing is used. Records will be committed to the database when the commit_max_ctr defined in the RESTART_CONTROL table is reached.

I/O Specification

Integration Type	Upload to Sales Audit
File Name	The input file name is not fixed; the input file name is determined by a runtime parameter. Records rejected by the import process are written to a reject file. The reject file name is not fixed; the reject file name is determined by a runtime parameter.
Integration Contract	IntCon000160 (SAVO)

Input File Layout

Table 6-97 Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FHEAD	Record descriptor/	Char(5)	FHEAD	File head marker.
	Line id	Number(10)	0000000001	Unique line ID.
	Translator id	Char(5)	SAVO	Identifies transaction type.
	File create date	Char(14)	N/A	Vdate in YYYYMMDDHH24MISS format.
FDETL	Business Date	Char(8)	Business Date	Vdate in YYYYMMDD format.
	Record descriptor/	Char(5)	FDETL	File head marker.
	Line id	Number(10)	N/A	Unique line ID.
	Voucher seq Number	Number(20)	N/A	Unique identifier for an entry to the SA_VOUCHER table.
	Voucher No	Char(16)	N/A	Serial Number of the voucher.
	Tender Type Id	Number(6)	N/A	Type of Voucher (Valid values for tender type are maintained in the pos_tender_type_head table with tender_type_group as VOUCH.
	Assigned Date	Char(8)	N/A	Date the voucher was assigned.
	Assigned store	Number(10)	N/A	Store to which the voucher is assigned.
	Issuing Date	Char(8)	N/A	Date this document was issued.
	Issuing store	Number(10)	N/A	Store this document was issued from.
	Issuing Register	Char(5)	N/A	Register this document was issued from.
	Issuing Cashier	Char(10)	N/A	Cashier issuing the document.
	Issued transaction number	Number(20)	N/A	Transaction number at the time of issuance.
	Issued item seq number	Number(4)	N/A	Will hold the item sequence of the item when the voucher is sold as an item (gift voucher).
	Issued tender seq number	Number(4)	N/A	Will hold the tender sequence of the tender when the voucher is sold as a tender (Merchandise Credit).
	Issued Amount	Number(20)	N/A	Amount the voucher was issued for*10000 (4 implied decimal places).
	Issued Customer Name	Char(120)	N/A	Name of the customer, who was issued the voucher.
	Issued Customer Addr1	Char(240)	N/A	The address of the customer who was issued the voucher.

Table 6-97 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
FTAIL	Issued Customer Addr2	Char(240)	N/A	The second line address of the customer who was issued the voucher.
	Issued Customer City	Char(120)	N/A	City of the customer, the voucher is issued.
	Issued Customer State	Char(3)	N/A	State of the customer.
	Issued Customer Postal Code	Char(30)	N/A	Postal address of the customer.
	Issued Customer Country	Char(3)	N/A	Country of the customer where the voucher was issued.
	Recipient Name	Char(120)	N/A	Name of the intended recipient.
	Recipient State	Char(3)	N/A	The state of the intended recipient.
	Recipient Country	Char(3)	N/A	The country of the intended recipient.
	Redemption Date	Char(8)	N/A	Date the voucher was redeemed.
	Redemption Store	Number(10)	N/A	Store at which the voucher was redeemed.
	Redemption Register	Char(5)	N/A	Register at which the document was redeemed.
	Redemption cashier	Char(10)	N/A	Cashier redeeming the voucher.
	Redemption tran seq number	Number(20)	N/A	Transaction Number when the document was redeemed.
	Redemption Tender seq number	Number(4)	N/A	This column will hold the tender sequence of the tender within the transaction when a voucher is redeemed as tender.
	Redemption Amount	Number(20)	N/A	Amount the voucher was redeemed for*10000 (4 implied decimal places).
	Expiry Date	Char(8)	N/A	Expiry Date.
	Status	Char(1)	N/A	Indicator showing the document's status - issued or redeemed. Valid values = I - Issued, R - Redeemed.
	Comments	Char(2000)	N/A	Comments.
	Record type	Char(5)	FTAIL	Describes file record and marks the end of file.
	Line id	Number(10)	N/A	Unique file line ID.

Table 6-97 (Cont.) Input File Layout

Record Name	Field Name	Field Type	Default Value	Description
	#lines	Number(10)	N/A	Total number of transaction lines in file (not including FHEAD and FTAIL).

Design Assumptions

N/A

Sales Posting

All sales data, whether imported from Sales Audit or directly from POS and OMS solutions, are uploaded into Merchandising using one of the following scheduled inbound integrations are included in this functional area:

- [Process Multiple POSU Files \(uploadsales_all.ksh\)](#)
- [Upload POSU File for Processing \(uploadsales.ksh\)](#)

For more on sales processing, see the *Merchandising Batch Operations Guide*.

Process Multiple POSU Files (uploadsales_all.ksh)

Module Name	uploadsales_all.ksh
Description	Process Multiple POSU Files
Functional Area	Sales Posting
Module Type	Integration
Module Technology	Ksh
Catalog ID	RMS157
Wrapper Script	batch_uploadsales.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this script is to execute the uploadsales.ksh module for all POSU files that are for upload. This wrapper will simplify the sales upload process for multiple POSU files, removing the need to call the uploadsales.ksh individually for each file.

Restart/Recovery

N/A

Locking Strategy

N/A

Security Considerations

N/A

Performance Considerations

The number of threads, the amount of waiting time, number for retries, and average volume of data should be considered. `RETRY_WAIT_TIME` shouldn't be increased significantly.

The rows, bindsize and readsize parameter of the `sqlldr` command can be configured for better performance. This gives more control over how many times the inserts are committed/executed.

Security Considerations

N/A

I/O Specification

Integration Type	Upload to Merchandising
File Name	POSU_<store>_<tran_date>_<sysdate>.<thread_val>
Integration Contract	IntCon000044

Input File Layout

Refer to the Input File Layout section in `uploadsales.doc`.

Upload POSU File for Processing (`uploadsales.ksh`)

Module Name	<code>uploadsales.ksh</code>
Description	Upload POSU File for Processing
Functional Area	Sales Posting
Module Type	Integration
Module Technology	Ksh
Catalog ID	RMS112
Wrapper Script	<code>batch_uploadsales.ksh</code>

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

The purpose of this module is to upload the contents of the POSU file from Sales Audit or 3rd Party POS to the staging table for further processing.

Restart/Recovery

N/A

Locking Strategy

N/A

Security Considerations

N/A

Performance Considerations

The number of threads, the amount of waiting time, number for retries, and average volume of data should be considered. RETRY_WAIT_TIME shouldn't be increased significantly.

The rows, bindsize and readsize parameter of the sqlldr command can be configured for better performance. This gives more control over how many times the inserts are committed/executed.

Security Considerations

N/A

I/O Specification

Integration Type	Upload to Merchandising
File Name	POSU_<store>_<tran_date>_<sysdate>.<thread_val>
Integration Contract	IntCon000044

Input File Layout

Table 6-98 Input File

Record Name	Field Name	Field Type	Default Value	Description
File Header	File Type Record Descriptor	Char(5)	FHEAD	Identifies file record type
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file
	File Type Definition	Char(4)	POSU	Identifies file as 'POS Upload'
	File Create Date	Char(14)	N/A	Date file was written by external system
	Location Number	Number(10)	N/A	Store identifier
	Vat include indicator	Char(1)	N/A	Determines whether or not the store stores values including vat. Not required but populated by Sales Audit
	Vat region	Number(4)	N/A	Vat region the given location is in. Not required but populated by Sales Audit

Table 6-98 (Cont.) Input File

Record Name	Field Name	Field Type	Default Value	Description
Transaction Header	Currency code	Char(3)	N/A	Currency of the given location. Not required but populated by sales audit
	Currency retail decimals	Number(1)	N/A	Number of decimals supported by given currency for retails. Not required but populated by Sales Audit
	File Type Record Descriptor	Char(5)	THEAD	Identifies transaction record type
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file
	Transaction Date	Char(14)	Transaction date	Date sale/return transaction was processed at the POS
	Item Type	Char(3)	REF or ITM	Item type will be represented as a REF or ITM
	Item Value	Char(25)	N/A	The ID number of an ITM or REF
	Dept	Number(4)	N/A	Dept of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Class	Number(4)	N/A	Class of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Subclass	Number(4)	N/A	Subclass of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Pack Indicator	Char(1)	N/A	Pack indicator of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Item level	Number(1)	N/A	Item level of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Tran level	Number(1)	N/A	Tran level of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Wastage Type	Char(6)	N/A	Wastage type of item sold or returned. Not required but populated by Oracle Retail Sales Audit

Table 6-98 (Cont.) Input File

Record Name	Field Name	Field Type	Default Value	Description
	Wastage Percent	Number(12)	N/A	Wastage Percent*10000 (4 implied decimal places.), wastage percent of item sold or returned. Not required but populated by Oracle Retail Sales Audit
	Transaction Type	Char(1)	S - sales R - return	Transaction type code to specify whether transaction is a sale or a return
	Drop Shipment Indicator	Char(1)	Y N	Indicates whether the transaction is a drop shipment or not. If it is a drop shipment, indicator will be 'Y'. This field is not required, but will be defaulted to 'N' if blank
	Total Sales Quantity	Number(12)	N/A	Total sales quantity * 10000 (4 implied decimal places), number of units sold at a particular location
	Selling UOM	Char(4)	N/A	UOM at which this item was sold
	Sales Sign	Char(1)	P - positive N - negative	Determines if the Total Sales Quantity and Total Sales Value are positive or negative
	Total Sales Value	Number(20)	N/A	Total Sales Value * 10000 (4 implied decimal places), sales value, net sales value of goods sold
	Last Modified Date	Char(14)	N/A	For VBO future use
	Catchweight Indicator	Char(1)	NULL	Indicates if the item is a catch weight item. Valid values are 'Y' or NULL
	Actual Weight Quantity	Number(12)	NULL	Actual Weight Quantity*10000 (4 implied decimal places), the actual weight of the item, only populated if catchweight_ind = 'Y'
	Sub Trantype Indicator	Char(1)	NULL	Tran type for Sales Audit Valid values are 'A', 'D', NULL
	Total Igtax Value	Number(20)	N/A	Total Igtax Value * 10000 (4 implied decimal places), goods sold or returned
	Sales Type	Char(1)	N/A	Indicates whether the line item is a Regular Sale, a customer order serviced by OMS (External CO) or a customer order serviced by a store (In Store CO). Valid values are 'R', 'E', or 'I'

Table 6-98 (Cont.) Input File

Record Name	Field Name	Field Type	Default Value	Description
	No Inventory Return Indicator	Char(1)	N/A	Contains an indicator that identifies a return without inventory. This is generally a non-required column, but in case of Returns, this is required. Valid values are 'Y' or 'N'
	Return Disposition	Char(10)	N/A	Contains the disposition code published by RWMS as part of the returns upload to OMS
	Return Warehouse	Number(10)	N/A	Contains the physical warehouse ID for the warehouse identifier where the item was returned
	Customer Order No	Char(48)	N/A	This column contains the customer order number ID.
	Fulfillment Order No	Char(48)	N/A	This column contains the fulfillment order number ID.
	Fulfillment Loc Type	Char(2)	N/A	This column contains the fulfillment location type. Code for the fulfillment loc type from code_detail where code_type = 'FLTP'
	Fulfillment Loc	Number(10)	N/A	This column contains the fulfillment loc ID.
	Orig Store	Number(10)	N/A	This column contains the original store value for a Return transaction.
	POS Tran Id	Number(20)	N/A	This column contains the unique identifier for a sale transaction. This is an Optional field. Customer needs to provide a unique identifier for a sale transaction. If not provided, Merchandising assigns a unique value from a DB sequence.
	Posting Store	Number(10)		This column contains the store at which the item sale/return should be accounted for in case of cross-store sales happening at co-located stores. It is expected that this field will be populated only for items that are checked out at a different store from the one at which they are originally managed.

Table 6-98 (Cont.) Input File

Record Name	Field Name	Field Type	Default Value	Description
	Consignment Rate	Number(12)		This column contains the consignment rate that should be applied while posting the sales/returns to Merchandising. (4 implied decimal places)
	Consignment Unit Cost	Number(20)		This column contains the consignment unit cost that should be applied while posting the sales/returns to Merchandising. (4 implied decimal places)
	Inventory Identifier Type	Char(6)		This column contains the inventory identifier type passed in Sales/Return transactions. Valid values are found under the Inventory Identifier Types (IIDT) code type; for example, Lot (L), Expiry Date (E), Import Document (D).
	Inventory Id	Char(120)		This column contains the inventory ID value being passed in sales/return transactions.
Transaction Tax	File Type Record Descriptor	Char(5)	TTAX	Identifies the file record type
	File Line Identifier	Number(10)	Specified by external system	Sequential file line number
	Tax Code	Char(6)	N/A	Holds the tax code associated to the item
	Tax Rate	Number(20)	N/A	Tax rate*10000000000(10 implied decimal places), holds the tax rate for the tax code associated to the item
	Total Tax Value	Number(20)	N/A	Total Tax value*10000(4 implied decimal places), total tax amount for the line item
Transaction Detail	File Type Record Descriptor	Char(5)	TDETL	Identifies transaction record type
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file
	Promotional Tran Type	Char(6)	N/A	Code for promotional type from code_detail, code_type = 'PRMT'
	Promotion Number	Number(10)	N/A	Promotion number from the Merchandising

Table 6-98 (Cont.) Input File

Record Name	Field Name	Field Type	Default Value	Description
Transaction Trailer	Sales Quantity	Number(12)	N/A	Sales quantity*10000 (4 implied decimal places.), number of units sold in this prom type
	Sales Value	Number(20)	N/A	Sales value*10000 (4 implied decimal places.), value of units sold in this prom type
	Discount Value	Number(20)	NA	Discount quantity*10000 (4 implied decimal places.), value of discount given in this prom type
	Promotion Component	Number(10)	N/A	Links the promotion to additional pricing attributes
	File Type Record Descriptor	Char(5)	TTAIL	Identifies file record type
File Trailer	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file
	Transaction Count	Number(6)	Specified by external system	Number of TDETL records in this transaction set
	File Type Record Descriptor	Char(5)	FTAIL	Identifies file record type
	File Line Identifier	Number(10)	Specified by external system	ID of current line being processed by input file
	File Record Counter	Number(10)	N/A	Number of records/transactions processed in current file (only records between fhead & ftail)

Fields expected in POSU format based on changes adopted:

	V16	V16+ Customer Order Changes	V19	V19.3	V24	V25
Fulfillment Order No	No	Yes	Yes	Yes	Yes	Yes
Fulfillment Loc Type	No	Yes	Yes	Yes	Yes	Yes
Fulfillment Loc	No	Yes	Yes	Yes	Yes	Yes
Orig Store	No	Yes	Yes	Yes	Yes	Yes
POS Tran Id	No	No	Yes	Yes	Yes	Yes
Posting Store	No	No	No	Yes	Yes	Yes
Consignment Rate/Unit Cost	No	No	No	No	Yes	Yes
Inventory Identifier Type	No	No	No	No	No	Yes
Inventory Id	No	No	No	No	No	Yes

Design Assumptions

Multiple taxes for an item if sent from POS to Sales Audit, will be summed to a single tax in Merchandising and assigned one of the applicable tax codes.

Rolling up transactions to the item/store/price point

The program uploadsales.ksh requires that transactions be rolled up the item/store/price point level. The tables below give a hypothetical (though not particularly realistic) example of the type of rollup required by upload_sales.ksh.

Table 6-99 Sales for Item Number 1234 (at one store during one period of the day)

Transaction Number	Number of Items Sold	Amount (in specified currency unit)	Price point (price reason)
167	1	9.99	Regular
395	2	18.00	Promotional
843	1	7.99	Clearance
987	3	27.00	Promotional
1041	1	9.99	Regular
1265	4	31.96	Clearance

Note

The variation of the price per item in different transactions. This is the result of the price applied at the time of sale—the price point. Now look at the next table that shows the same transactions rolled up by item and price point.

Table 6-100 Sales for Item Number 1234

Number of Items Sold	Price Reason (price point)	Total Amount for Item-Price point (in currency)
2	Regular price	19.98
5	Promotional price	45.00
5	Clearance price	39.95

uploadsales.ksh takes the totals and looks for any discounts for transactions in the POSU file. It applies the discounts to an expected total dollar amount using the price listed for that item from the pricing table (PRICE_HIST). It next compares this expected total against the reported total. If the program finds a discrepancy between the two amounts, it is reported. If the two totals match, the rollup is considered valid. If value-added tax (VAT) is included in any sales transaction amounts, it is removed from those transactions prior to the validation process.

Reject File

The module produces a reject file similar to the input file if it is found to have missing or duplicate FHEAD or FTAIL records. Records in these types of files are loaded to the svc_posupld_load table, but not in the svc_posupld_staging table.

Forecasting

Merchandising has the ability to upload forecast data from an external source. Forecasts can be uploaded by week or by day.

Scheduled forecast integration processes include:

- [Daily Demand Item Forecast Subscription API](#)
- [Weekly Demand Item Forecast Subscription API](#)
- [Weekly/Daily Item Forecast Upload \(load_item_forecast\)](#)

Daily Demand Item Forecast Subscription API

This section describes the Daily Demand Item Forecast Subscription API.

Functional Area

Foundation

Design Overview

This API is used to import daily forecast data from Oracle Retail Inventory Planning Optimization - Demand Forecasting to Merchandising. It uses BDI (Bulk Data Integration), which is an integration layer that facilitates the bulk transfer of information between solutions. On this particular integration, the data flow is from Inventory Planning Optimization - Demand Forecasting to BDI, and then BDI to Merchandising. To accomplish this data transfer, BDI will invoke a Merchandising owned API that will pull data from the BDI integration layer and load into the Merchandising daily forecast table (DAILY_ITEM_FORECAST).

Note

The job that manages this import is scheduled in Inventory Planning Optimization - Demand Forecasting, rather than as part of the Merchandising batch schedule.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Daily Demand Item Forecast	Import daily demand item forecast from BDI	DlyDmdFst_Tx_BdiInterfaceModule.xml

Tables

TABLE	SELECT	INSERT	UPDATE	DELETE
DLY_DMND_FRCST_IN	Yes	No	No	No
DAILY_ITEM_FORECAST	No	Yes	Yes	Yes

Weekly Demand Item Forecast Subscription API

This section describes the Weekly Demand Item Forecast Subscription BDI.

Functional Area

Foundation

Design Overview

This API is used to import weekly forecast data from Oracle Retail Inventory Planning Optimization - Demand Forecasting to Merchandising. It uses BDI (Bulk Data Integration), which is an integration layer that facilitates the bulk transfer of information between solutions. On this particular integration stream, the data flow is from Inventory Planning Optimization - Demand Forecasting to BDI, and then BDI to Merchandising. To accomplish this data transfer, BDI will invoke a Merchandising-owned API that will pull data from BDI integration layer BDI table and load into the Merchandising weekly forecast table (ITEM_FORECAST). This process begins by preserving the previous 4 weeks of forecasted sales data in ITEM_FORECAST_HIST and then the ITEM_FORECAST table is truncated. Then the new forecast data is imported.

Note

The job that manages this import is scheduled in Inventory Planning Optimization - Demand Forecasting, rather than as part of the Merchandising batch schedule.

Data Definition XML

The BDI interface staging tables are generated based on the XML schema definition.

Data Flow	Description	XML Schema Definition (XSD)
Weekly Demand Item Forecast	Import weekly demand item forecast from BDI	WklyDmdFst_Tx_BdiInterfaceModule.xml

Tables

TABLE	SELECT	INSERT	UPDATE	DELETE
WKLY_DMND_FRCST_IN	Yes	No	No	No
ITEM_FORECAST	No	Yes	Yes	Yes
ITEM_FORECAST_HIST	No	Yes	No	Yes

Weekly/Daily Item Forecast Upload (load_item_forecast)

Module Name	load_item_forecast.ksh
Description	Load daily/weekly item forecast from Oracle Retail Inventory Planning Optimization Cloud Service - Demand Forecasting
Functional Area	Integration - Forecast
Module Type	Integration

Module Technology	Ksh
Catalog ID	N/A
Wrapper Script	rmswrap_shell_in.ksh

Schedule

Oracle Retail Merchandising Batch Schedule

Design Overview

This script loads item forecast data into the Merchandising forecast tables.

The forecast data comes from Inventory Planning Optimization - Demand Forecasting in a CSV (comma separated) format file. Merchandising expects a single comma-delimited input file (that is, a csv file) in the format specified in the sqlldr control scripts `load_item_forecast.ctl` (for Weekly) and `load_daily_item_forecast.ctl` (for Daily). Please refer to [Integration Contract](#) for more details. A run-time parameter (that is, run type) of D or W indicates whether the Daily or Weekly forecast data is being loaded into Merchandising. If the forecast is a daily forecast, information is written to the `DAILY_ITEM_FORECAST` table. If the forecast is a weekly forecast, information is written to the `ITEM_FORECAST` table. Depending on the run type parameter, the batch truncates the respective forecast table prior to loading.

Restart/Recovery

Evaluate the successful load of the data.

In case of any failures:

SQL load – SQL load dumps invalid records that do not meet certain technical requirements (that is, data type inconsistencies, and so on). The rejected record is written either to a bad file or to a discard file. The discard file contains records that do not satisfy conditions such as missing or invalid record types. Records with other technical issues are written to the bad file.

Note

A non-fatal code is returned by the program and a message will be written to the log file if reject files are created.

User Action: When such conditions exist, you may update either the bad or the discard file and attempt to reload using the same files. You may also fix the data input file and reload, so that the item forecast tables will be truncated and upload item forecast tables with the corrected the data.

Integration Contract

If a run-time parameter of 'weekly' is used, the input file is a single comma-delimited file (that is, a CSV file):

Field Name	Field Type	Required	Description
EOW_DATE	Date(8)	Yes	Item_forecast.eow_date (YYYYMMDD)

Field Name	Field Type	Required	Description
ITEM	Char(25)	Yes	Item_forecast.item
LOC	Char(10)	Yes	Item_forecast.loc
FORECAST_SALES	Double(14)	Yes	Item_forecast.forecast_sales Note - this field can contain decimal quantities. Unlike quantity fields in Merchandising ProC Batch files, this qty field is not assumed to be extended to significant digits.
FORECAST_STD_DEV	Double(14)	Yes	Item_forecast.forecast_std_dev Note - this field can contain decimal quantities. Unlike quantity fields in Merchandising ProC Batch files, this qty field is not assumed to be extended to significant digits.

If a run-time parameter of 'daily' is used, the input file is a single comma-delimited file (that is, a CSV file):

Field Name	Field Type	Required	Description
DATA_DATE	Date(8)	Yes	Daily_item_forecast.data_date (YYYYMMDD)
ITEM	Char(25)	Yes	Daily_item_forecast.item
LOC	Char(10)	Yes	Daily_item_forecast.loc
FORECAST_SALES	Double(14)	Yes	Daily_item_forecast.forecast_sales Note - this field can contain decimal quantities. Unlike quantity fields in Merchandising ProC Batch files, this qty field is not assumed to be extended to significant digits.
FORECAST_STD_DEV	Double(14)	Yes	Daily_item_forecast.forecast_std_dev Note - this field can contain decimal quantities. Unlike quantity fields in Merchandising ProC Batch files, this qty field is not assumed to be extended to significant digits.

I/O Specification

N/A

Design Assumption

Domain is not a relevant concept any more. Domain_id on ITEM_FORECAST and DAILY_ITEM_FORECAST will always be 1.