

Oracle® Order Management

Open Interfaces, API, & Electronic Messaging Guide

Release 12

Part No. B31585-01

April 2007

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments

Preface

1 Integrating Your Systems

Overview of Oracle Order Management APIs and Open Interfaces.....	1-1
Basic Business Needs.....	1-1
Oracle Order Management Interfaces.....	1-2
Inbound Open Interface Model.....	1-11
Components of an Open Interface.....	1-13

2 Oracle Advanced Pricing Open Interfaces and APIs

Key of Short Names.....	2-2
Agreements Public Application Program Interface.....	2-2
Functional Overview.....	2-2
Setting Up and Using Parameter Descriptions.....	2-2
Validation of Agreements Public API.....	2-26
Example of Agreements Public API.....	2-26
Attribute Mapping Application Program Interface.....	2-26
Functional Overview.....	2-27
Setting Up and Parameter Descriptions.....	2-28
Business Object for Modifier Setup Application Program Interface.....	2-48
Functional Overview.....	2-48
Setting Up and Parameter Descriptions.....	2-50
Validation of Business Object for Modifier Setup API.....	2-86
Example of Modifier Setup Application Program Interface.....	2-86

Business Object for Pricing Formulas Application Program Interface	2-87
Functional Overview.....	2-87
Setting Up and Parameter Descriptions.....	2-88
Validation of Business Object for Pricing Formulas API.....	2-99
Example of Pricing Formulas API.....	2-99
Business Object for Pricing Limits Application Program Interface	2-100
Functional Overview.....	2-100
Setting Up and Parameter Descriptions.....	2-101
Validation of Limits Public API.....	2-118
Example of Limits Public API.....	2-119
Create Publish Price Book Application Program Interface	2-119
Functional Overview.....	2-119
Setting Up and Parameter Descriptions.....	2-120
Validation of Create Publish Price Book API.....	2-123
Examples of Create_Publish_Price_Book API.....	2-126
Get Currency Application Program Interface	2-126
Functional Overview.....	2-126
Setting Up and Parameter Descriptions.....	2-126
Validation of Get_Currency API.....	2-127
Get Custom Price Application Program Interface	2-128
Functional Overview.....	2-128
Setting Up and Parameter Descriptions.....	2-128
Validation of Get Custom Price API.....	2-130
Get Price Book Application Program Interface	2-133
Functional Overview.....	2-133
Setting Up and Parameter Descriptions	2-134
Validation of Get Price Book API.....	2-144
Examples of Get_Price_Book API.....	2-145
Get Price List Application Program Interface	2-145
Functional Overview.....	2-145
Setting Up and Parameter Descriptions.....	2-146
Validation of Get_Price_List API.....	2-147
Multi-Currency Conversion Setup Application Program Interface	2-147
Functional Overview.....	2-147
Setting Up and Parameter Descriptions.....	2-148
Validation of Multi-Currency Conversion API.....	2-156
Example of Multi-Currency Conversion API.....	2-157
Price List Setup Application Program Interface	2-157
Functional Overview.....	2-157
Setting Up and Parameter Descriptions.....	2-159
Validation of Price List Setup API.....	2-178

Example of Price List Setup API.....	2-178
Price List Setup Group Application Program Interface.....	2-179
Functional Overview.....	2-179
Setting Up and Parameter Descriptions.....	2-180
Validation of Price List Group API.....	2-199
Price Request Application Program Interface.....	2-200
Functional Overview.....	2-201
Setting Up and Parameter Descriptions.....	2-202
Validation of Price Request API.....	2-260
Example of Price Request Application Program Interface.....	2-260
Pricing Data Bulk Loader API.....	2-263
Functional Overview	2-263
Setting Up and Parameter Descriptions.....	2-264
Validation of Pricing Data Bulk Loader	2-265
Examples of Pricing Data Bulk Loader API.....	2-267
Pricing Object Security - Check Function API.....	2-275
Functional Overview.....	2-276
Setting Up and Parameter Descriptions.....	2-276
Validation of Pricing Object Security API.....	2-278
Pricing Attributes Application Program Interface.....	2-278
Functional Overview.....	2-278
Attribute Mapping Application Program Interface.....	2-291
Functional Overview.....	2-291
Setting Up and Parameter Descriptions.....	2-291
Validation of Attribute Mapping API.....	2-297
Qualifiers Application Program Interface.....	2-297
Functional Overview.....	2-297
Setting Up and Parameter Descriptions	2-298
Validation of Qualifiers API.....	2-315
Example of Qualifiers API.....	2-315
Reverse Limits Application Program Interface.....	2-315
Functional Overview.....	2-316
Setting Up and Parameter Descriptions.....	2-316
Validation of Reverse Limits API.....	2-321
Example of Reverse Limits API.....	2-322
Round Price Application Program Interface.....	2-323
Functional Overview.....	2-323
Setting Up and Parameter Descriptions.....	2-324
Validation of Round Price API.....	2-325
Validate Price List with Currency Code Application Program Interface.....	2-325
Functional Overview.....	2-326

Setting Up and Parameter Descriptions.....	2-326
Validation of Validate Price List with Currency Code API.....	2-326

3 Oracle Advanced Pricing XML Transactions

Get Catalog Transaction	3-1
Setting up and Parameter Descriptions.....	3-2
Validation of Get Catalog Transaction.....	3-12
Examples of the Get Catalog Transaction	3-13
Sync Catalog Transaction	3-17
Setting Up and Parameter Descriptions.....	3-18
Validation of Sync Catalog Transaction.....	3-29
Examples of Sync Catalog Transaction.....	3-29

4 Oracle Order Management XML Transactions: Implementation Considerations

Process_PO	4-1
Setup: Customer Setup.....	4-1
Installation.....	4-6
Message Map: Seeded Workflow for Inbound XML Messages.....	4-7
Sample Business Flow for 3A4 Process PO/3A4 Acknowledge PO.....	4-9
Acknowledge_PO	4-22
Message Map.....	4-22
Setup: Customer Setup.....	4-22
Oracle XML Gateway Details.....	4-26
Show_SalesOrder	4-27
Setup.....	4-27
Additional User Setup.....	4-31
Change_SalesOrder	4-39
Setup.....	4-39
Data Archive and Purge Procedures.....	4-43
Change_PO	4-44
Setup: Customer Setup.....	4-44
Implementation Considerations.....	4-47
Message Map.....	4-47
Sample Business Flow for 3A4 Process PO/3A4 Acknowledge PO.....	4-76
Cancel_PO	4-86
Setup: Customer Setup	4-86
Implementation Considerations.....	4-90
Open Interface Tracking	4-90
Setup: Customer Setup.....	4-90
Integration Event – oracle.apps.ont.oi.xml_int.status	4-91

User Procedures.....	4-100
5 Oracle Order Management XML Transactions	
What is RosettaNet?.....	5-1
Process_PO	5-4
Acknowledge_PO.....	5-41
Show_SalesOrder	5-62
Change_salesorder.....	5-78
Inbound Change PO Request	5-96
Cancel Purchase Order	5-136
CONFIRM BOD.....	5-139
Message Map.....	5-139
Seeded Workflow for Inbound XML Messages.....	5-148
Message Details	5-150
Extension Tags.....	5-153
Open Interface Tracking	5-156
Overview.....	5-156
Major Features.....	5-157
Business Scenarios and Process Flow.....	5-158
Messages.....	5-158
Event/Event Subscription.....	5-161
Seeded workflow OEEM/Open Interface Tracking.....	5-161
Event Parameters & Population.....	5-161
6 Oracle Order Management EDI Transactions	
Oracle Order Management.....	6-1
Inbound Purchase Order (POI/850/ORDERS).....	6-2
Order Import	6-5
Review and Resolve Order Management Open Interface Exceptions.....	6-5
7 Oracle Order Management Open Interfaces and APIs	
Integrating Oracle Order Management Using Order Import.....	7-1
Process Order Application Open Interface.....	7-110
8 Oracle Order Management: Integration with Other Products	
Integrating Oracle Order Management with Oracle Receivables and Invoicing.....	8-1
Telecommunications Services Orders (MACD) Interfaces.....	8-52

9 Oracle Release Management XML Transactions

Overview of Oracle Release Management XML	9-1
Planning Schedule Transaction	9-3
Sample Document Type Definition (DTD) and XML for Planning Schedule.....	9-4
Planning Schedule XML to Database Mapping.....	9-9
Shipping Schedule Transaction	9-18
Sample Document Type Definition (DTD) and XML for Shipment Schedule.....	9-20
Shipping Schedule XML to Database Mapping.....	9-23

10 Oracle Release Management EDI Transactions

Production Sequence EDI	10-1
Shipping Schedule EDI	10-1
Planning Schedule EDI	10-2
Purpose Codes	10-2
Transaction Descriptions	10-4

11 Oracle Release Management Open Interfaces and APIs

Understanding the Demand Processor Interface Tables	11-1
Release Management Schedule Headers RLM_INTERFACE_HEADERS_ALL.....	11-1
Release Management Schedule Lines RLM_INTERFACE_LINES_ALL.....	11-23

12 Oracle Shipping Execution Open Interfaces and APIs

Overview of API Information	12-1
Shipment Processing Using APIs	12-3
Actions, APIs, and Parameters	12-16
Application Parameter Initialization	12-34
Trip Public Application Program Interface	12-35
Create_Update_Trip API Features.....	12-35
Functional Overview.....	12-35
Trip_Action API Features.....	12-41
Functional Overview.....	12-41
Trip_Action (overloaded version) API features.....	12-43
Functional Overview.....	12-44
Stop Public Application Program Interface	12-49
Create_Update_Stop API Features.....	12-49
Functional Overview.....	12-49
Stop_Action API Features.....	12-56
Functional Overview.....	12-56

Deliveries Public Application Program Interface.....	12-59
Create_Update_Delivery API Features.....	12-59
Functional Overview.....	12-59
Delivery_Action API Features.....	12-70
Functional Overview.....	12-70
Generate_Documents API Features.....	12-76
Functional Overview.....	12-76
Setting Up the Generate_Documents API.....	12-77
Exceptions Application Program Interface.....	12-78
Exception_Action API Features.....	12-78
Functional Overview.....	12-78
Setting Up the Exception_Actions API.....	12-79
Get_Exceptions API Features.....	12-87
Functional Overview.....	12-87
Setting Up the Get_Exceptions API.....	12-87
Delivery Details Public Application Program Interface.....	12-89
Detail_To_Delivery API Features.....	12-90
Functional Overview.....	12-90
Split_Line API Features.....	12-92
Functional Overview.....	12-92
Update_Shipping_Attributes API Features.....	12-95
Functional Overview.....	12-95
Autocreate_Deliveries API Features.....	12-107
Autocreate_Del_Trip API Features.....	12-109
Get_Detail_Status API Features.....	12-111
Functional Overview.....	12-111
Update_Shipping_Attributes (overloaded version) API.....	12-113
Functional Overview.....	12-113
Init_Changed_Attribute_Rec API Features.....	12-125
Functional Overview.....	12-125
AUTOCREATE_DEL_TRIP (overloaded version) API Features.....	12-136
Functional Overview.....	12-136
Container Public Application Program Interface.....	12-138
Create_Containers API Features.....	12-139
Functional Overview.....	12-139
Update Container API Features.....	12-142
Functional Overview.....	12-142
Auto_Pack API Features.....	12-149
Functional Overview.....	12-149
Container_Actions API Features.....	12-152
Functional Overview.....	12-152

Freight Costs Public Application Program Interface	12-155
Create_Update_Freight_Costs API Features.....	12-155
Functional Overview.....	12-155
Validate_Freight_Cost_Type API Features.....	12-161
Delete_Freight_Costs API Features.....	12-162
Functional Overview.....	12-162
Pick Release Application Program Interface	12-167
Create_Batch API Features.....	12-168
Functional Overview.....	12-168
Release_Batch API Features.....	12-179
Functional Overview.....	12-179
Get_Batch_Record API Features.....	12-181
Functional Overview.....	12-181
Document Public Application Program Interface	12-193
Functional Overview.....	12-193
Custom Public Application Program Interface	12-199
UI_Location_Code API Features.....	12-199
Functional Overview.....	12-199
Calculate_TP_Dates API Features.....	12-203
Functional Overview.....	12-203
Example of API Packages and Procedures	12-204

13 Oracle Shipping Execution XML Transactions

XML Overview	13-1
Oracle XML Gateway Setup	13-2
Trading Partners.....	13-3
Code Conversions.....	13-3
Using XML Messaging with Carrier Manifesting	13-4
ShowShipment Request Transaction for Carrier Manifesting.....	13-5
ShowShipment Advice Transaction for Carrier Manifesting.....	13-6
Defining Trading Partners for Carrier Manifesting.....	13-7
Overview of XML and DTD for Carrier Manifesting.....	13-10
Document Type Definition (DTD) for Carrier Manifesting.....	13-10
Sample XML Transaction for Carrier Manifesting.....	13-13
Using XML Messaging with Third Party Warehousing	13-88
ShowShipment Request Transaction for Third Party Warehousing.....	13-89
ShowShipment Advice Transaction for Third Party Warehousing.....	13-90
Defining Trading Partners for Third Party Warehousing.....	13-90
Overview of XML and DTD for Third Party Warehousing.....	13-95
Sample Document Type Definitions (DTD) for Third Party Warehousing.....	13-95

Sample XML Transactions for Third Party Warehousing.....	13-98
Using XML Messaging with International Trade Management.....	13-176
ItemSynchronization Request Transaction for ITM.....	13-176
PartySynchronization Request Transaction for ITM.....	13-177
OrderCompliance Request for ITM.....	13-177
DeliveryCompliance Request Transaction for ITM.....	13-177
ExportCompliance Response Transaction for ITM.....	13-178
Overview of XML and DTD for ITM.....	13-179
ITM DTDs.....	13-179
Sample ITM XML Transactions.....	13-187

14 Oracle Shipping Execution EDI Transactions

DSNO.....	14-1
-----------	------

A Oracle Order Management Error and Information Messages

Process_PO and Acknowledge_PO.....	A-1
Show_SalesOrder.....	A-2
Cancel_PO.....	A-2
Change SO.....	A-3
Inbound Change PO Request.....	A-4

Send Us Your Comments

Oracle Order Management Open Interfaces, API, & Electronic Messaging Guide, Release 12 **Part No. B31585-01**

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 new Applications Release Online Documentation CD available on Oracle MetaLink and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: appsdoc_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 www.oracle.com.

Preface

Intended Audience

Welcome to Release 12 of the *Oracle Order Management Open Interfaces, API, & Electronic Messaging Guide*.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- The Oracle Applications graphical user interface. To learn more about the Oracle Applications graphical user interface, read the Oracle Applications User's Guide.

See Other Information Sources for more information about Oracle Applications product information.

See Related Information Sources on page xvi for more Oracle Applications product information.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology

vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

- 1 Integrating Your Systems**
- 2 Oracle Advanced Pricing Open Interfaces and APIs**
- 3 Oracle Advanced Pricing XML Transactions**
- 4 Oracle Order Management XML Transactions: Implementation Considerations**
- 5 Oracle Order Management XML Transactions**
- 6 Oracle Order Management EDI Transactions**
- 7 Oracle Order Management Open Interfaces and APIs**
- 8 Oracle Order Management: Integration with Other Products**
- 9 Oracle Release Management XML Transactions**
- 10 Oracle Release Management EDI Transactions**
- 11 Oracle Release Management Open Interfaces and APIs**
- 12 Oracle Shipping Execution Open Interfaces and APIs**
- 13 Oracle Shipping Execution XML Transactions**
- 14 Oracle Shipping Execution EDI Transactions**
- A Oracle Order Management Error and Information Messages**

Related Information Sources

Oracle Order Management User's Guide

Oracle Order Management Implementation Manual

Using Oracle Workflow in Oracle Order Management

Oracle e-Commerce Gateway Implementation Manual

Oracle e-Commerce Gateway User's Guide

Oracle XML Gateway User's Guide

Oracle Integration Repository User's Guide

Oracle Configure to Order Process Guide

Oracle Shipping Execution User's Guide

Oracle Advanced Pricing Implementation Manual

Oracle Advanced Pricing User's Guide

Oracle Release Management Implementation Manual

Oracle Release Management User's Guide

Oracle Applications Multiple Organizations Implementation Guide

Integration Repository

The Oracle Integration Repository is a compilation of information about the service endpoints exposed by the Oracle E-Business Suite of applications. It provides a complete catalog of Oracle E-Business Suite's business service interfaces. The tool lets users easily discover and deploy the appropriate business service interface for integration with any system, application, or business partner.

The Oracle Integration Repository is shipped as part of the E-Business Suite. As your instance is patched, the repository is automatically updated with content appropriate for the precise revisions of interfaces in your environment.

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle **STRONGLY RECOMMENDS** that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using an Oracle Applications form can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

Integrating Your Systems

This chapter covers the following topics:

- Overview of Oracle Order Management APIs and Open Interfaces
- Basic Business Needs
- Oracle Order Management Interfaces
- Inbound Open Interface Model
- Components of an Open Interface

Overview of Oracle Order Management APIs and Open Interfaces

Oracle Order Management products provide a number of open interfaces that you can link to, and exchange data with, other Oracle applications and non-Oracle applications like legacy systems.

This chapter will help you understand the general model that Oracle Order Management products use for open interfaces. Other chapters in this book provide information about how to use specific open interfaces. Additional functional information about the interfaces is available in the user or implementation guide for each product. Additional technical information on these interfaces is available in the product-specific e-Technical Reference Manuals.

Basic Business Needs

APIs and open interfaces of the Oracle Order Management Suite provide you with the features you need to support the following basic business needs:

- Connect to data collection devices. You can collect material movement transactions such as receipts, issues, quality data, movements, completions, and shipments. This improves data entry speed and transaction accuracy.
- Connect to other systems such as finite scheduling packages, computer-aided

design systems, and custom and legacy manufacturing systems to create integrated enterprise wide systems.

- Connect to external systems such as the customer's purchasing system and the supplier's order entry system to better integrate the supply chain via electronic commerce.
- Control processing of inbound data imported from outside Oracle applications.
- Validate imported data to maintain data integrity within the application.
- Review, update, and resubmit imported data that failed validation.
- Export data from Oracle Order Management products to external systems.

Oracle Order Management Interfaces

Open Interface Architectures

Oracle Order Management products use three different methods for importing and exporting data:

- Interface Tables
- Interface Views (Business Views)
- Function Calls or Programmatic Interfaces (Processes)

Interface Tables

Interface tables in Oracle Order Management applications provide a temporary storage area for loading information from an external source. After the information is loaded, concurrent programs are executed to validate the information and then apply the information to the base product tables.

The benefit of an interface table is that it provides a repository where records can be processed and if errors are found, you can edit the records and resubmit them.

In the Oracle Order Management family of applications you should never write directly to the product tables. An Oracle Applications validation step is always required. You may achieve this either by loading information into the interface tables and submitting a provided concurrent program to validate and process the information or by using a function call to a programmatic interface.

Interface Views (Business Views)

Views simplify the data relationships for easier processing, whether for reporting or data export. Oracle Order Management products have defined *business views* that identify areas of key business interest. You can access this data using a tool of your choice. The OE_ORDER_HEADERS_BV is an example of a key business view.

Function Calls or Programmatic Interfaces (Processes)

As an alternative to the two step process of writing to an interface table and executing a program to process the table data, many Oracle Order Management interfaces support direct function calls. A calling application can pass appropriate parameters and execute a public function to invoke the application logic.

The benefit of a function call is that the integration is real time, as opposed to interface tables where the integration is batch.

Summary: Beyond Published Interfaces

The Oracle Cooperative Applications Initiative references many third party products that provide import and export capabilities and allow varying degrees of integration with legacy systems, supplier systems, and so on. Contact your Oracle consultant for more information about system integration.

Some Key Terms and Tables

The lists below provide information on the database tables, views, procedures, and modules used in data integration.

Table Key

Key	Description
Data Flow Direction	<i>Inbound</i> means into Oracle Order Management; <i>Outbound</i> means out from Oracle Order Management
Interface Man	The interface is documented in detail in the <i>Oracle Order Management Open Interfaces, API, & Electronic Messaging Guide</i>
e-TRM	The tables, views, or modules are described in the product's e-Technical Reference Manual

Oracle Order Management APIs/Open Interfaces

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
ONT	ONT	ONT	ONT	ONT	ONT
Order Import	Inbound	Table	Yes	Yes	OE_HEADER S_IFACE_AL L OE_LINES_IF ACE_ALL OE_RESERV TNS_IFACE_ ALL OE_CREDITS _IFACE_ALL OE_PRICE_A DJS_IFACE_ ALL OE_LOTSERI ALS_IFACE_ ALL OE_ACTION S_IFACE_AL L
Process Order	Inbound	Process	No	Yes	OE_ORDER_ PUB.PROCES S_ORDER
QP	QP	QP	QP	QP	QP

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Agreement Public Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	OE_PRICING_CONT_PUB.PROCESS_AGREEMENT
					OE_PRICING_CONT_PUB.GET_AGREEMENT
					OE_PRICING_CONT_PUB.LOCK_AGREEMENT
Attribute Mapping Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_ATTR_MAPPING_PUB.BUILD_CONTEXTS
Business Object for Modifier Setup Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_MODIFIERS_PUB.PROCESS_MODIFIERS
					QP_MODIFIERS_PUB.GET_MODIFIERS
					QP_MODIFIERS_PUB.LOCK_MODIFIERS

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Business Object for Pricing Formulas Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_PRICE_FORMULA_PUB.LOCK_PRICE_FORMULA QP_PRICE_FORMULA_PUB.PROCESS_PRICE_FORMULA QP_PRICE_FORMULA_PUB.GET_PRICE_FORMULA
Business Object for Pricing Limits Application Program Interface		Procedure	Yes	Yes	QP_LIMITS_PUB.PROCESS_LIMITS QP_LIMITS_PUB.GET_LIMITS QP_LIMITS_PUB.LOCK_LIMITS
Create Publish Price Book Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_PRICE_BOOK_PUB.CREATE_PUBLISH_PRICE_BOOK

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Custom Runtime Sourcing Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_RUNTIME_SOURCE.GET_NUMERIC_ATTRIBUTE_VALUE
Get Currency Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_CURRENCY_PUB.GET_CURRENCY
Get_Attribute_Text Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_UTIL_PUB.GET_ATTRIBUTE_TEXT
Get Custom Price (Used in Formulas Setup) Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_CUSTOM.GET_CUSTOM_PRICE
Get Price Book Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_PRICE_BOOK.GET_PRICE_BOOK
Get Price For Line Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_PREQ_PUB.PRICE_REQUEST

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Get Price List Currency Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_UTIL_PUB.GET_CURRENCY
Get Price List Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_UTIL_PUB.GET_PRICE_LIST
Multi-Currency Conversion Setup Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_CURRENCY_PUB.PROCESS_CURRENCY
Price List Setup	Inbound/Outbound	Procedure	Yes	Yes	QP_PRICE_LIST_PUB.PROCESS_PRICE_LIST QP_PRICE_LIST_PUB.GET_PRICE_LIST QP_PRICE_LIST_PUB.LOCK_PRICE_LIST
Price List Setup Group Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_PRICE_LIST_GRP.PROCESS_PRICE_LIST

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Price Request Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_PREQ_GRP.PRICE_REQUEST
Pricing Data Bulk Loader API	Inbound/Outbound	Procedure	Yes	Yes	QP_BULK_LOADER_PUB.LOAD_PRICING_DATA
Pricing Object Security CHECK_FUNCTION	Inbound/Outbound	Procedure	Yes	Yes	QP_SECURITY.CHECK_FUNCTION
Qualifiers Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_QUALIFIER_RULES_PUB.PROCESS_QUALIFIER_RULES QP_QUALIFIER_RULES_PUB.LOCK_QUALIFIER_RULES QP_QUALIFIER_RULES_PUB.GET_QUALIFIER_RULES QP_QUALIFIER_RULES_PUB.COPY_QUALIFIER_RULES

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Reverse Limits Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_UTIL_PUB.REVERSE_LIMITS
Round Price Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_UTIL_PUB.ROUND_PRICE
Validate_Price_list_Currency_code Application Program Interface	Inbound/Outbound	Procedure	Yes	Yes	QP_UTIL_PUB.VALIDATE_PRICE_LIST_CURRENCY_CODE
RLM	RLM	RLM	RLM	RLM	RLM
RLM_INTERFACE_HEADERS	Inbound	Table	Yes	Yes	RLM_INTERFACE_HEADERS
RLM_INTERFACE_LINES	Inbound	Table	Yes	Yes	RLM_INTERFACE_LINES
WSH	WSH	WSH	WSH	WSH	WSH
Trip Public API	Inbound	Procedure	Yes	No	WSH_TRIPS_PUB (Procedure package)
Stop Public API	Inbound	Procedure	Yes	No	WSH_TRIP_STOPS_PUB (Procedure package)

Interface/API Name	Data Flow Direction	Table, View, Process, or Procedure	Iface Man	TRM	Table, View, Module Name, or Procedure Name
Deliveries Public API	Inbound	Procedure	Yes	No	WSH_DELIVERIES_PUB (Procedure package)
Exceptions Public API	Inbound	Procedure	Yes	No	WSH_EXCEPTIONS_PUB (Procedure package)
Delivery Details Public API	Inbound	Procedure	Yes	No	WSH_DELIVERY_DETAILS_PUB (Procedure package)
Container Public API	Inbound	Procedure	Yes	No	WSH_CONTAINER_PUB (Procedure package)
Freight Costs Public API	Inbound	Procedure	Yes	No	WSH_FREIGHT_COSTS_PUB (Procedure package)
Pick Release Application Program Interface API	Inbound	Procedure	Yes	No	WSH_PICKING_BATCHES_PUB (Procedure package)

Inbound Open Interface Model

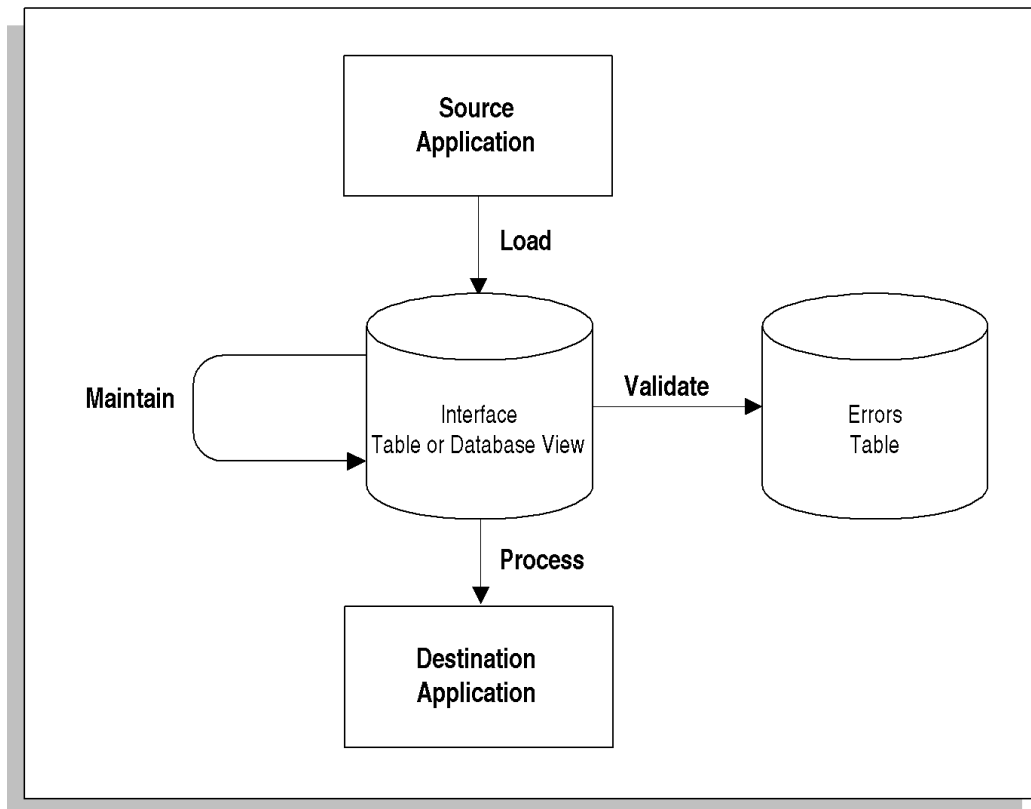
Oracle Order Management products use both inbound and outbound interfaces. For inbound transactions and data import, interface tables as well as supporting validation, processing, and maintenance programs are available. For outbound transactions and data export to external systems, database views are available and the destination application provides the validation, processing, and maintenance programs.

Discussion of Inbound Interfaces

In this manual, we discuss only inbound interfaces in detail. You can find information about the tables, views, and processes related to outbound interfaces in the e-TRM for each product. Note that the e-TRMs do *not* contain detailed, narrative descriptions about the outbound interfaces.

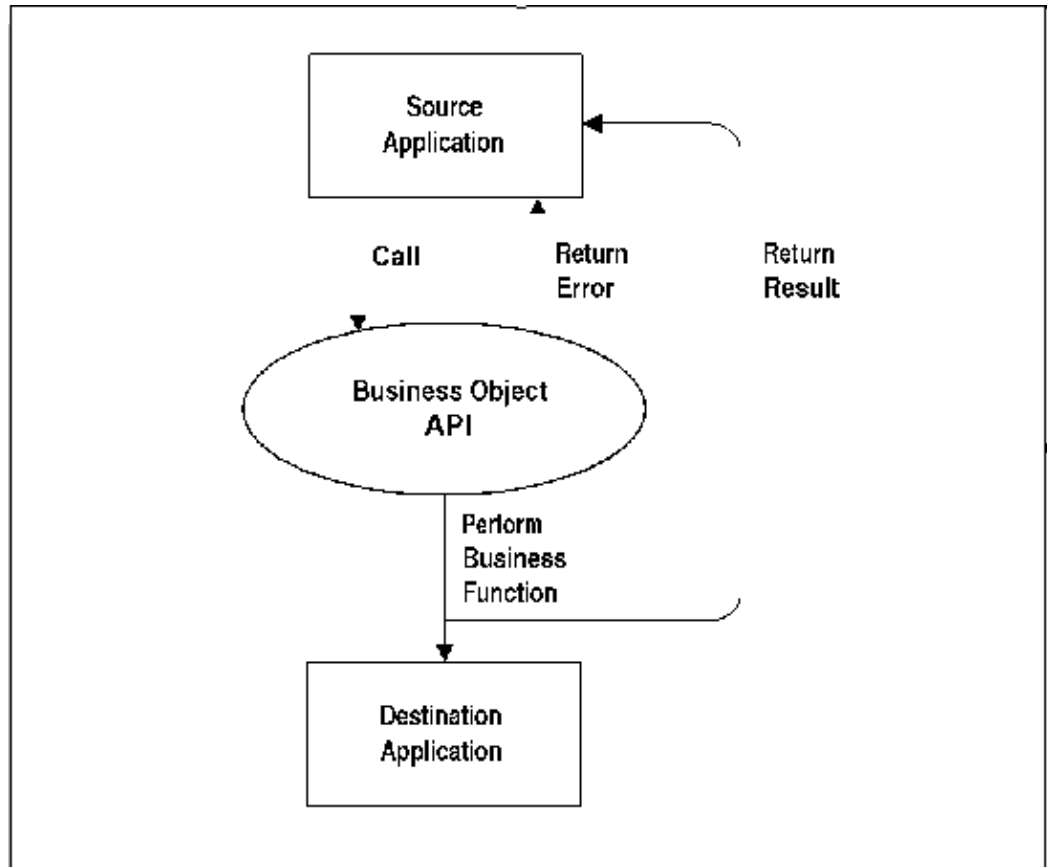
Open Interface Diagram

This diagram shows the general model for open application interfaces:



Open API Diagram

This diagram illustrates the model used by APIs such as the Service Request interfaces (Oracle Service):



Components of an Open Interface

The components described below are common to most open interfaces. Please note that the component may be implemented slightly differently than described below.

Source Application

A source application may be an external system or another module in the e-Business Suite. Data may be processed and stored in a source application.

Destination Application

A destination application may be an external system or another module in the e-Business Suite. You send data to a destination application so that the application can perform further processing and storage.

Interface Table

For inbound interfaces, the interface table is the intermediary table where the data from

the source application temporarily resides until it is validated and processed into an Oracle Order Management product. The various types of interface columns, are listed below:

Identifier Columns

Identifier columns uniquely identify rows in the interface table and provide foreign key reference to both the source and destination applications. For example, typical identifier columns for a move transaction would identify:

- The source application
- The row's unique identifier in the source application
- The destination application's unique identifier.

Control Columns

Control columns track the status of each row in the interface table as it is inserted, validated, processed with or without errors, and ultimately deleted. Additional control columns identify who last updated the row and the last update date.

Data Columns

Data columns store the specific attributes that the source application is sending to the Oracle Order Management products.

Required Columns

Required columns store the minimum information needed by the Oracle Order Management products to successfully process the interface row.

Some columns are conditionally required based on the specifics of the interface.

Derived Columns

Derived columns are created by the destination application from information in the required columns. Derived columns are usually required columns that need to contain a value so that processing may continue.

Optional Columns

The data in optional columns is not mandatory for transaction processing to take place, however the optional columns may be useful for storing additional information.

Errors Table

For inbound interfaces, the errors table stores all errors generated by the validation and processing functions. You can either handle multiple error rows or single rows to resolve the errors.

Database View

Database views are database objects that enable users of destination applications to look

at source application data.

Oracle Order Management products provide predefined views of key data that is likely to be used by destination applications.

For example, Oracle Cost Management provides work in process valuation and transaction distribution database views for use by other cost reporting destination products.

Load Function

For inbound interfaces, the load function is the set of programs that selects and accumulates data from the source application and inserts it into Oracle Order Management interface tables. The programming languages and tools used in the load function are highly dependent on the hardware and system software of the source application.

For example, if you are passing data between an Oracle based source application and an Oracle destination application, you would likely use a tool such as Pro*C or PL/SQL since these tools work in both environments. If you are bringing data from a non-Oracle based application into a product interface table, you would likely use a procedural language available on the source application to select the data and convert it into an ASCII file. Then you could use SQL*Loader to insert that file into the destination product's interface table.

For outbound interfaces, the load function is the SQL that creates the database view.

Validate Function

The validate function is the set of programs that Oracle Applications destination products use to ensure the integrity of inbound data. In the source application, you can typically validate data upon entry using techniques such as forms triggers, not null columns, data types, and so on. However, since Oracle Applications products may not be the source of this data, validation programs ensure data integrity.

In addition, the validate function can derive additional columns based on the required columns and foreign key relationships with data elsewhere in the Oracle destination application.

The validation programs check the interface table for rows requiring validation, and the programs validate and update each row indicating whether validation is complete or if errors were found. If errors were found, validation programs need to write errors to the error table of the destination application.

When the source application is a product in the e-Business Suite, the destination application should provide the validate function.

Process Function

The process function is a set of programs that processes the data from the interface table

into the Oracle destination product. The specific processing performed varies by application. For open transaction interfaces, the processing generally includes recording the transaction history, updating the inventory and order balances, and charging for costs.

Interfaces typically let you control both the frequency and the number of validated rows that the processing programs attempt to process. Upon successful completion of processing, the process function should delete the processed row from the interface table.

Occasionally, the process function may need to insert rows into the errors table.

When the source application is a product in the e-Business Suite, the destination application should provide the process function.

Maintain Function

The maintain function is generally accomplished from a window within an Oracle application. Most of these windows allow you to query, update, and resubmit interface records that have failed validation. In general you can use these windows to query unprocessed or unvalidated rows to check their current status.

When the maintain function is not called explicitly, you can use SQL*Plus to query and update the interface table rows that are in error.

When the source application is a product in the e-Business Suite, the destination application should provide the maintain function.

Oracle Advanced Pricing Open Interfaces and APIs

This chapter covers the following topics:

- Key of Short Names
- Agreements Public Application Program Interface
- Attribute Mapping Application Program Interface
- Business Object for Modifier Setup Application Program Interface
- Business Object for Pricing Formulas Application Program Interface
- Business Object for Pricing Limits Application Program Interface
- Create Publish Price Book Application Program Interface
- Get Currency Application Program Interface
- Get Custom Price Application Program Interface
- Get Price Book Application Program Interface
- Get Price List Application Program Interface
- Multi-Currency Conversion Setup Application Program Interface
- Price List Setup Application Program Interface
- Price List Setup Group Application Program Interface
- Price Request Application Program Interface
- Pricing Data Bulk Loader API
- Pricing Object Security - Check Function API
- Pricing Attributes Application Program Interface
- Attribute Mapping Application Program Interface
- Qualifiers Application Program Interface

- Reverse Limits Application Program Interface
- Round Price Application Program Interface
- Validate Price List with Currency Code Application Program Interface

Key of Short Names

A key of the short names and definitions used in the API tables are provided in the following table:

Key of Short Names

Short name	Definition
Drv	Derived
Req	Required <ul style="list-style-type: none"> • Yes: This is a required parameter. • No: This is an optional parameter.
N/A (no entry)	No value/not applicable

Agreements Public Application Program Interface

This section explains how to use the Agreements Public API and how it functions in Oracle Advanced Pricing. The Agreements Public package consists of entities to support creating and maintaining agreements.

Functional Overview

Process_Agreement processes inserts, updates, and deletes records related to agreements.

Setting Up and Using Parameter Descriptions

The following tables describe all parameters used by the public Agreements Public API. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

PROCESS_AGREEMENT

The following table shows the parameters for this structure.

PROCESS_AGREEMENT Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	IN	Number	No	No
p_init_msg_list	IN	Varchar2	No	No
p_return_values	IN	Varchar2	No	No
p_commit	IN	Varchar2	No	No
x_return_status	OUT	Varchar2	No	No
x_msg_count	OUT	Number	No	No
x_msg_data	OUT	Varchar2	No	No
p_Agreement_rec	IN	Agreement_Rec_Type	No	No
p_Agreement_value_rec	IN	Agreement_Val_Rec_Type	No	No
p_Price_LHeader_rec	IN	QP_Price_List_PUB.Price_List_Rec_Type	No	No
p_Price_LHeader_val_rec	IN	QP_Price_List_PUB.Price_List_Val_Rec_Type	No	No
p_Price_LLine_tbl	IN	QP_Price_List_PUB.Price_List_Line_Tbl_Type	No	No
p_Price_LLine_val_tbl	IN	QP_Price_List_PUB.Price_List_Val_Tbl_Type	No	No

Parameter	Usage	Type	Req	Drv
p_Pricing_Attr_tbl	IN	QP_Price_List_P UB.Pricing_Attr_ Tbl_Type	No	No
p_Pricing_Attr_val_tbl	IN	QP_Price_List_P UB.Pricing_Attr_ Val_Tbl_Type	No	No
x_Agreement_rec	OUT	Agreement_Rec_ Type	No	No
x_Agreement_val_rec	OUT	Agreement_Val_ Rec_Type	No	No
x_Price_LHeader_rec	OUT	QP_Price_List_P UB.Price_List_Re c_Type	No	No
x_Price_LHeader_val_rec	OUT	QP_Price_List_P UB.Price_List_V al_Rec_Type	No	No
x_Price_LLine_tbl	OUT	QP_Price_List_P UB.Price_List_Li ne_Tbl_Type	No	No
x_Price_LLine_val_tbl	OUT	QP_Price_List_P UB.Price_List_Li ne_Val_Tbl_Typ e	No	No
x_Pricing_Attr_tbl	OUT	QP_Price_List_P UB.Pricing_Attr_ Tbl_Type	No	No
x_Pricing_Attr_val_tbl	OUT	QP_Price_List_P UB.Pricing_Attr_ Val_Tbl_Type	No	No

AGREEMENT_REC_TYPE

The following table shows the parameters for this structure.

REC_TYPE

Parameter	Usage	Type	Req	Drv
accounting_rule_id	Null	Number	No	No
agreement_contact_id	Null	Number	No	No
agreement_id	Null	Number	Yes1	No
agreement_num	Null	Varchar2(50)	No	No
agreement_source_code	Null	Varchar2(30)	No	No
agreement_type_code	Null	Varchar2(30)	No	No
attribute1	Null	Varchar2(150)	No	No
attribute2	Null	Varchar2(150)	No	No
attribute3	Null	Varchar2(150)	No	No
attribute4	Null	Varchar2(150)	No	No
attribute5	Null	Varchar2(150)	No	No
attribute6	Null	Varchar2(150)	No	No
attribute7	Null	Varchar2(150)	No	No
attribute8	Null	Varchar2(150)	No	No
attribute9	Null	Varchar2(150)	No	No
attribute10	Null	Varchar2(150)	No	No
attribute11	Null	Varchar2(150)	No	No

Parameter	Usage	Type	Req	Drv
attribute12	Null	Varchar2(150)	No	No
attribute13	Null	Varchar2(150)	No	No
attribute14	Null	Varchar2(150)	No	No
attribute15	Null	Varchar2(150)	No	No
comments	Null	Varchar2(30)	No	No
context	Null	Varchar2(30)	No	No
created_by	Null	Number	Yes	No
creation_date	Null	Date	Yes	No
sold_to_org_id	Null	Number	No	No
end_date_active	Null	Date	No	No
freight_terms_code	Null	Varchar2(30)	No	No
invoice_contact_id	Null	Number	No	No
invoice_to_org_id	Null	Number	No	No
invoicing_rule_id	Null	Number	No	No
last_updated_by	Null	Number	Yes	No
last_update_date	Null	Date	Yes	No
last_update_log_id	Null	Number	No	No
name	Null	Varchar2(30)	Yes	No

Parameter	Usage	Type	Req	Drv
orig_system_agr_id	Null	Number	No	No
override_arule_flag	Null	Varchar2(1)	Yes	No
override_irule_flag	Null	Varchar2(1)	Yes	No
price_list_id	Null	Number	Yes2	No
pricing_contract_id	Null	Number	No	No
purchase_order_num	Null	Varchar2(50)	No	No
revision	Null	Varchar2(50)	Yes	No
revision_date	Null	Date	Yes	No
revision_reason_code	Null	Varchar2(30)	No	No
salesrep_id	Null	Number	No	No
ship_method_code	Null	Varchar2(30)	No	No
signature_date	Null	Date	No	No
start_date_active	Null	Date	No	No
term_id	Null	Number	Yes	No
return_status	Null	Varchar2(1)	No	No
db_flag	Null	Varchar2(1)	No	No
operation	Null	Varchar2(30)	Yes	No

Parameter	Usage	Type	Req	Drv
tp_attribute1	Null	Varchar2(240)	No	No
tp_attribute2	Null	Varchar2(240)	No	No
tp_attribute3	Null	Varchar2(240)	No	No
tp_attribute4	Null	Varchar2(240)	No	No
tp_attribute5	Null	Varchar2(240)	No	No
tp_attribute6	Null	Varchar2(240)	No	No
tp_attribute7	Null	Varchar2(240)	No	No
tp_attribute8	Null	Varchar2(240)	No	No
tp_attribute9	Null	Varchar2(240)	No	No
tp_attribute10	Null	Varchar2(240)	No	No
tp_attribute11	Null	Varchar2(240)	No	No
tp_attribute12	Null	Varchar2(240)	No	No
tp_attribute13	Null	Varchar2(240)	No	No
tp_attribute14	Null	Varchar2(240)	No	No
tp_attribute15	Null	Varchar2(240)	No	No
tp_attribute_category	Null	Varchar2(30)	No	No

orig_system_agr_id

The following table describes notations listed in the preceding table:

Notations

Note	Description
1	For update and delete
2	If you are not passing a price list record

AGREEMENT_TBL_TYPE

The following table shows the parameters for this structure.

AGREEMENT_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Agreement_Rec_Type	Null	Record	No	No

AGREEMENT_VAL_REC_TYPE

The following table shows the parameters for this structure.

AGREEMENT_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accounting_rule	Null	Varchar2(240)	Null	Null
agreement_contact	Null	Varchar2(240)	Null	Null
agreement	Null	Varchar2(240)	Null	Null
agreement_type	Null	Varchar2(240)	Null	Null
customer	Null	Varchar2(240)	Null	Null
freight_terms	Null	Varchar2(240)	Null	Null
invoice_contact	Null	Varchar2(240)	Null	Null

Parameter	Usage	Type	Req	Drv
invoice_to_site_ use	Null	Varchar2(240)	Null	Null
invoicing_rule	Null	Varchar2(240)	Null	Null
override_arule	Null	Varchar2(240)	Null	Null
override_irule	Null	Varchar2(240)	Null	Null
price_list	Null	Varchar2(240)	Null	Null
revision_reason	Null	Varchar2(240)	Null	Null
salesrep	Null	Varchar2(240)	Null	Null
ship_method	Null	Varchar2(240)	Null	Null
term	Null	Varchar2(240)	Null	Null

AGREEMENT_VAL_TBL_TYPE

The following table shows the parameters for this structure.

AGREEMENT_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Agreement_Val_ Rec_Type	Null	Record	No	No

PRICE_LIST_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	No	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
currency_code	Null	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
discount_lines_flag	Null	Varchar2	No	No
end_active_date	Null	Date	No	No
freight_terms_code	Null	Varchar2	No	No
gsa_indicator	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_type_code	Null	Varchar2	No	No
program_applica tion_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update _date	Null	Date	No	No
prorate_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
rounding_factor	Null	Number	No	No
ship_method_co de	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
terms_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No
name	Null	Varchar2	Yes	No
description	Null	Varchar2	No	No
version_no	Null	Varchar2	No	No
active_flag	Null	Varchar2	No	No
mobile_download	Null	Varchar2	No	No
currency_header_id	Null	Number	No	No
pte_code	Null	Varchar2	No	Yes
list_source_code	Null	Varchar2	No	No
orig_system_header_ref	Null	Varchar2	No	No
global_flag	Null	Varchar2	No	No
org_id	Null	Number	No	Yes

Note: This API will initialize MOAC if it is not initialized prior. On the price_list_rec, if the global_flag is passed as FND_API.G_MISS_CHAR, the API will default the global_flag as Y. If global_flag is passed as N and if org_id is not null, the API will create the modifier list for that operating unit (OU). If global_flag is passed as "N" and if org_id is not passed, the API will derive the org_id from the org context if set; if not set, use the default OU for that responsibility. The API validates the Org_id to verify if the Org_Id corresponds to one of the OUs for that responsibility and raises an error if the org_id is not valid.

PRICE_LIST_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Price_List_Rec_Type	Null	Record	No	No

PRICE_LIST_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
automatic	Null	Varchar2	No	No
currency	Null	Varchar2	No	No
discount_lines	Null	Varchar2	No	No
freight_terms	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_type	Null	Varchar2	No	No
prorate	Null	Varchar2	No	No
ship_method	Null	Varchar2	No	No
terms	Null	Varchar2	No	No
currency_header	Null	Varchar2	No	No
pte	Null	Varchar2	No	No

PRICE_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Price_List_Val_Rec_Type	Null	Record	No	No

PRICE_LIST_LINE_REC_TYPE

The following table shows the parameters for this structure:

PRICE_LIST_LINE_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accrual_qty	Null	Number	No	No
accrual_uom_code	Null	Varchar2	No	No
arithmetic_operator	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	No	No
base_qty	Null	Number	No	No
base_uom_code	Null	Varchar2	No	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
effective_period_uom	Null	Varchar2	No	No
end_date_active	Null	Date	No	No
estim_accrual_rate	Null	Number	No	No
generate_using_formula_id	Null	Number	No	No
inventory_item_id	Null	Number	No	No
last_updated_by	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
list_line_type_co de	Null	Varchar2	No	No
list_price	Null	Number	No	No
modifier_level_c ode	Null	Varchar2	No	No
number_effectiv e_periods	Null	Number	No	No
operand	Null	Number	No	No
organization_id	Null	Number	No	No
override_flag	Null	Varchar2	No	No
percent_price	Null	Number	No	No
price_break_type _code	Null	Varchar2	No	No
price_by_formul a_id	Null	Number	No	No
primary_uom_fl ag	Null	Varchar2	No	No
print_on_invoice _flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
rebate_txn_type_code	Null	Varchar2	No	No
related_item_id	Null	Number	No	No
relationship_type_id	Null	Number	No	No
reprice_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
revision	Null	Varchar2	No	No
revision_date	Null	Date	No	No
revision_reason_code	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
substitution_attribute	Null	Varchar2	No	No
substitution_context	Null	Varchar2	No	No
substitution_value	Null	Varchar2	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
operation	Null	Varchar2	No	No
from_rltd_modifier_id	Null	Number	No	No
rltd_modifier_group_no	Null	Number	No	No
rltd_modifier_group_type	Null	Varchar2	No	No
product_precedence	Null	Number	No	No
Customer_Item_Id	Null	Number	Yes	No
price_break_header_index	Null	Number	No	No
list_line_no	Null	Number	No	Yes
qualification_ind	Null	Number	No	Yes

PRICE_LIST_LINE_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Price_List_Line_Rec_Type	Null	Record	No	No

PRICE_LIST_LINE_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accrual_uom	Null	Varchar2	No	No
automatic	Null	Varchar2	No	No
base_uom	Null	Varchar2	No	No
generate_using_formula	Null	Varchar2	No	No
inventory_item	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
list_line_type	Null	Varchar2	No	No
modifier_level	Null	Varchar2	No	No
organization	Null	Varchar2	No	No
override	Null	Varchar2	No	No
price_break_type	Null	Varchar2	No	No
price_by_formula	Null	Varchar2	No	No
primary_uom	Null	Varchar2	No	No
print_on_invoice	Null	Varchar2	No	No
rebate_transaction_type	Null	Varchar2	No	No
related_item	Null	Varchar2	No	Null

Parameter	Usage	Type	Req	Drv
relationship_type	Null	Varchar2	No	Null
reprice	Null	Varchar2	No	Null
revision_reason	Null	Varchar2	No	Null

PRICE_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Price_List_Line_Val_Rec_Type	Null	Record	No	No

PRICING_ATTR_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accumulate_flag	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
attribute_groupi ng_number	Null	Number	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
excluder_flag	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_line_id	Null	Number	No	No
pricing_attribute	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
pricing_attribute_context	Null	Varchar2	No	No
pricing_attribute_id	Null	Number	No	No
pricing_attribute_value_from	Null	Varchar2	No	No
pricing_attribute_value_to	Null	Varchar2	No	No
product_attribute	Null	Varchar2	Yes	No
product_attribute_context	Null	Varchar2	Yes	No
product_attribute_value	Null	Varchar2	Yes	No
product_uom_code	Null	Varchar2	Yes	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
request_id	Null	Number	No	No
pricing_attribute_value_from_number	Null	Number	No	Yes
pricing_attribute_value_to_number	Null	Number	No	Yes
qualification_ind	Null	Number	No	Yes

Parameter	Usage	Type	Req	Drv
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No
PRICE_LIST_LI NE_index	Null	Number	No	No
from_rltd_modif ier_id	Null	Number	No	No
comparison_ope rator_code	Null	Varchar2	Yes	No
product_attribut e_datatype	Null	Varchar2	No	Yes
pricing_attrubut e_datatype	Null	Varchar2	No	Yes
list_header_id	Null	Number	No	Yes
pricing_phase_id	Null	Number	No	Yes

PRICING_ATTR_TBL_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Rec _Type	Null	Record	No	No

PRICING_ATTR_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accumulate	Null	Varchar2	No	No
excluder	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
pricing_attribute	Null	Varchar2	No	No
product_uom	Null	Varchar2	No	No
pricing_attribute_desc	Null	Varchar2	No	No
pricing_attr_value_from_desc	Null	Varchar2	No	No
pricing_attr_value_to_desc	Null	Varchar2	No	No

PRICE_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Val_Rec_Type	Null	Record	No	No

Note: This API will initialize MOAC if it has not been initialized prior. For more information, see the *Oracle Multi-Organization Reference Manual*.

Validation of Agreements Public API

Standard Validation

Oracle Advanced Pricing validates all required columns in the Agreements Public API.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Agreements Public API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Example of Agreements Public API

The following example scripts are located in the directory \$qp/patch/115/sql:

- QPAGXMP1.sql: Insert an agreement and a price list header.
- QPAGXMP2.sql: Insert an agreement, a price list header, and lines.
- QPAGXMP3.sql: Create an agreement with a standard price list.
- QPAGXMP4.sql: Update an agreement which has a standard price list.
- QPAGXMP5.sql: Update agreement record and create agreement price list.
- QPAGXMP6.sql: Update an agreement record and update an agreement price list header.
- QPAGXMP7.sql: Delete an agreement.

Attribute Mapping Application Program Interface

This section explains how to use the Attribute Mapping APIs and how it functions in Oracle Advanced Pricing. Currently, in the new model, there are three Attribute Mapping packages. They are :

- QP_ATTR_MAP_PUB
- QP_ATTRIBUTES_PUB
- QP_ATTR_MAPPING_PUB

Functional Overview

The Public package QP_ATTR_MAP_PUB is a Business Object API, based on the following tables:

- QP_LOOKUPS (Type : QP_PTE_TYPE)
- QP_PTE_SOURCE_SYSTEMS
- QP_PTE_REQUEST_TYPES_B/TL
- QP_PTE_SEGMENTS
- QP_ATTRIBUTE_SOURCING
- QP_SOURCESYSTEM_FNAREA_MAP

The QP_ATTR_MAP_PUB model and its object name is Attr_Map and the relationship of the tables is shown below. The package QP_ATTR_MAP_PUB contains the following APIs and record type definitions:

Note: The API processes one record with each call. The calling application must populate the global records that referenced in the attribute mapping rules. The output of the API is a PL/SQL table with each record having the context name, attribute name, and attribute value.

- Pte_Val_Rec_Type
- Pte_Val_Rec_Type
- Rqt_Rec_Type
- Rqt_Tbl_Type
- Rqt_Val_Rec_Type
- Ssc_Rec_Type
- Ssc_Tbl_Type

- Ssc_Val_Rec_Type
- Psg_Rec_Type
- Psg_Val_Rec_Type
- Sou_Rec_Type
- Sou_Val_Rec_Type
- Fna_Rec_Type
- Fna_ValRec_Type
- PROCEDURE Process_Attr_Mapping
- PROCEDURE Lock_Attr_Mapping
- PROCEDURE Get_Attr_Mapping

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Attribute Mapping API QP_ATTR_MAP_PUB. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

Procedure PROCESS_ATTR_MAPPING

The following table shows the parameters for this structure. This API will add, update or delete a Source System, Request type, PTE-Attribute link, Attribute Mapping rules depending on the input parameter.

PROCESS_ATTR_MAPPING Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	Yes	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
x_return_status	Out	Number	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Pte_Rec_Type	No	No
p_PTE_rec	In	Pte_Val_Rec_Type	No	No
p_PTE_val_rec	In	Pte_Val_Rec_Type	No	No
p_RQT_tbl	In	Rqt_Tbl_Type	No	No
p_RQT_val_tbl	In	Rqt_Val_Tbl_Type	No	No
p_SSC_tbl	In	Ssc_Tbl_Type	No	No
p_SSC_val_tbl	In	Ssc_Val_Tbl_Type	No	No
p_PSG_tbl	In	Psg_Tbl_Type	No	No
p_PSG_val_tbl	In	Psg_Val_Tbl_Type	No	No
p_SOU_tbl	In	Sou_Tbl_Type	No	No
p_SOU_val_tbl	In	Sou_Val_Tbl_Type	No	No
p_FNA_tbl	In	Fna_Tbl_Type	No	No
p_FNA_val_tbl	In	Fna_Val_Tbl_Type	No	No
x_PTE_rec	Out	Pte_Rec_Type	No	No
x_PTE_val_rec	Out	Pte_Val_Rec_Type	No	No

Parameter	Usage	Type	Req	Drv
x_RQT_tbl	Out	Rqt_Tbl_Type	No	No
x_RQT_val_tbl	Out	Rqt_Val_Tbl_Type	No	No
x_SSC_tbl	Out	Ssc_Tbl_Type	No	No
x_SSC_val_tbl	Out	Ssc_Val_Tbl_Type	No	No
x_PSG_tbl	Out	Psg_Tbl_Type	No	No
x_PSG_val_tbl	Out	Psg_Val_Tbl_Type	No	No
x_SOU_tbl	Out	Sou_Tbl_Type	No	No
x_SOU_val_tbl	Out	Sou_Val_Tbl_Type	No	No
x_FNA_tbl	Out	Fna_Tbl_Type	No	No
x_FNA_val_tbl	Out	Fna_Val_Tbl_Type	No	No

p_api_version_number

This the version number of the API.

P_PTE_REC

The following table shows the parameters for this structure.

P_PTE_REC Parameters

Parameter	Usage	Type	Req	Drv
Description	Null	Varchar2	No	No
Enabled_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
End_date_active	Null	Date	No	No
Lookup_code	Null	Varchar2	No	No
Lookup_type	Null	Varchar2	No	No
meaning	Null	Varchar2	No	No
Start_date_active	Null	Date	No	No

P_PTE_VAL_REC

The following table shows the parameters for this structure.

P_PTE_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
lookup	Null	Varchar2	No	No

The following table shows the parameters for the tables and their structures.

Tables and Structures

Table Name	Usage	Type
P_RQT_TBL	Null	TABLE OF Rqt_Rec_Type
P_RQT_VAL_TBL	Null	TABLE OF Pte_Val_Rec_Type
P_SSC_TBL	Null	TABLE OF Ssc_Rec_Type
P_SSC_VAL_TBL	Null	TABLE OF Ssc_Val_Rec_Type
P_PSG_TBL	Null	TABLE OF Psg_Rec_Type

Table Name	Usage	Type
P_PSG_VAL_TBL	Null	TABLE OF Psg_Val_Rec_Type
P_SOU_TBL	Null	TABLE OF Sou_Rec_Type
P_SOU_VAL_TBL	Null	TABLE OF Sou_Val_Rec_Type
P_FNA_TBL	Null	TABLE OF Fna_Rec_Type
P_FNA_VAL_TBL	Null	TABLE OF Fna_Val_Rec_Type

X_PTE_REC_TYPE

The following table shows the parameters for this structure.

X_PTE_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Description	Null	Varchar2	No	No
Enabled_flag	Null	Varchar2	No	No
End_date_active	Null	Date	No	No
Lookup_code	Null	Varchar2	No	No
Lookup_type	Null	Varchar2	No	No
meaning	Null	Varchar2	No	No
Start_date_active	Null	Date	No	No

X_PTE_VAL_REC

The following table shows the parameters for this structure.

X_PTE_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
lookup	Null	Varchar2	No	No

The following table shows the parameters for the following table structures (no parameters are used for the following):

Table Parameters

Table Name	Usage	Type	Req	Drv
X_RQT_TBL	Null	TABLE OF Rqt_Rec_Type	No	No
X_RQT_VAL_TBL	Null	TABLE OF Pte_Val_Rec_Type	No	No
X_SSC_TBL	Null	TABLE OF Ssc_Rec_Type	No	No
X_SSC_VAL_TBL	Null	TABLE OF Ssc_Val_Rec_Type	No	No
X_PSG_TBL	Null	TABLE OF Psg_Rec_Type	No	No
X_PSG_VAL_TBL	Null	TABLE OF Psg_Val_Rec_Type	No	No
X_SOU_TBL	Null	TABLE OF Sou_Rec_Type	No	No
X_SOU_VAL_TBL	Null	TABLE OF Sou_Val_Rec_Type	No	No

Table Name	Usage	Type	Req	Drv
X_FNA_TBL	Null	TABLE OF Fna_Rec_Type	No	No
X_FNA_VAL_TB L	Null	TABLE OF Fna_Val_Rec_Ty pe	No	No

P_RQT_REC

The following table shows the parameters for this structure.

P_RQT_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled_flag	Null	Varchar2	No	No
line_level_global_struct	Null	Varchar2	No	No
line_level_view_name	Null	Varchar2	No	No
order_level_global_struct	Null	Varchar2	No	No
order_level_view_name	Null	Varchar2	No	No
pte_code	Null	Varchar2	No	No
request_type_code	Null	Varchar2	No	No
request_type_desc	Null	Varchar2	No	No
row_id	Null	Varchar2	No	No

P_RQT_VAL_REC

The following table shows the parameters for this structure.

P_RQT_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
pte	Null	Varchar2	No	No
request_type	Null	Varchar2	No	No
row	Null	Varchar2	No	No

P_SSC_REC

The following table shows the parameters for this structure.

P_SSC_REC Parameters

Parameter	Usage	Type	Req	Drv
application_short_name	Null	Varchar2	No	No
enabled_flag	Null	Varchar2	No	No
pte_code	Null	Varchar2	No	No
pte_source_system_id	Null	Varchar2	No	No

P_SSC_VAL_REC

The following table shows the parameters for this structure.

P_SSC_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
pte	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
pte_source_system	Null	Varchar2	No	No

P_PSG_REC

The following table shows the parameters for this structure.

P_PSG_REC Parameters

Parameter	Usage	Type	Req	Drv
limits_enabled	Null	Varchar2	No	No
lov_enabled	Null	Varchar2	No	No
pte_code	Null	Varchar2	No	No
seeded_sourcing_method	Null	Varchar2	No	No
segment_id	Null	Number	No	No
segment_level	Null	Varchar2	No	No
segment_pte_id	Null	Number	No	No
sourcing_enabled	Null	Varchar2	No	No
sourcing_status	Null	Varchar2	No	No
user_sourcing_method	Null	Varchar2	No	No

P_PSG_VAL_REC

The following table shows the parameters for this structure.

P_PSG_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
pte	Null	Varchar2	No	No
segment	Null	Varchar2	No	No
segment_pte	Null	Varchar2	No	No

P_SOU_REC

The following table shows the parameters for this structure.

P_SOU_REC Parameters

Parameter	Usage	Type	Req	Drv
attribute_sourcin g_id	Null	Number	No	No
attribute_sourcin g_level	Null	Varchar2	No	No
application_id	Null	Number	No	No
enabled_flag	Null	Varchar2	No	No
request_type_co de	Null	Varchar2	No	No
seeded_flag	Null	Varchar2	No	No
seeded_sourcing _type	Null	Varchar2	No	No
seeded_value_str ing	Null	Varchar2	No	No
segment_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
user_sourcing_type	Null	Varchar2	No	No
user_value_string	Null	Varchar2	No	No
PSG_index	Null	Number	No	No

P_SOU_VAL_REC

The following table shows the parameters for this structure.

P_SOU_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
attribute_sourcing	Null	Varchar2	No	No
enabled	Null	Varchar2	No	No
request_type	Null	Varchar2	No	No
seeded	Null	Varchar2	No	No
segment	Null	Varchar2	No	No

P_FNA_REC

The following table shows the parameters for this structure.

P_FNA_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled_flag	Null	Varchar2	No	No
functional_area_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
pte_source_system_fnarea_id	Null	Number	No	No
pte_source_system_id	Null	Number	No	No
request_id	Null	Number	No	No
seeded_flag	Null	Varchar2	No	No
SSC_index	Null	Number	No	No

P_FNA_VAL_REC

The following table shows the parameters for this structure.

P_FNA_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
functional_area	Null	Varchar2	No	No
pte_source_system_fnarea	Null	Varchar2	No	No
pte_source_system	Null	Varchar2	No	No
seeded	Null	Varchar2	No	No

Procedure LOCK_ATTR_MAPPING

The following table shows the parameters for this structure. User can use this API to lock PTE-Attribute link and all its Attribute Mapping rules from getting updated by a different user concurrently.

LOCK_ATTR_MAPPING Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	Yes	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
x_return_status	Out	Number	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Pte_Rec_Type	No	No
p_PTE_rec	In	Pte_Val_Rec_Type	No	No
p_PTE_val_rec	In	Pte_Val_Rec_Type	No	No
p_RQT_tbl	In	Rqt_Tbl_Type	No	No
p_RQT_val_tbl	In	Rqt_Val_Tbl_Type	No	No
p_SSC_tbl	In	Ssc_Tbl_Type	No	No
p_SSC_val_tbl	In	Ssc_Val_Tbl_Type	No	No
p_PSG_tbl	In	Psg_Tbl_Type	No	No
p_PSG_val_tbl	In	Psg_Val_Tbl_Type	No	No
p_SOU_tbl	In	Sou_Tbl_Type	No	No
p_SOU_val_tbl	In	Sou_Val_Tbl_Type	No	No

Parameter	Usage	Type	Req	Drv
p_FNA_tbl	In	Fna_Tbl_Type	No	No
p_FNA_val_tbl	In	Fna_Val_Tbl_Type	No	No
x_PTE_rec	Out	Pte_Rec_Type	No	No
x_PTE_val_rec	Out	Pte_Val_Rec_Type	No	No
x_RQT_tbl	Out	Rqt_Tbl_Type	No	No
x_RQT_val_tbl	Out	Rqt_Val_Tbl_Type	No	No
x_SSC_tbl	Out	Ssc_Tbl_Type	No	No
x_SSC_val_tbl	Out	Ssc_Val_Tbl_Type	No	No
x_PSG_tbl	Out	Psg_Tbl_Type	No	No
x_PSG_val_tbl	Out	Psg_Val_Tbl_Type	No	No
x_SOU_tbl	Out	Sou_Tbl_Type	No	No
x_SOU_val_tbl	Out	Sou_Val_Tbl_Type	No	No
x_FNA_tbl	Out	Fna_Tbl_Type	No	No
x_FNA_val_tbl	Out	Fna_Val_Tbl_Type	No	No

p_api_version_number

This the version number of the API.

P_PTE_REC_TYPE

The following table shows the parameters for this structure.

P_PTE_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Description	Null	Varchar2	No	No
Enabled_flag	Null	Varchar2	No	No
End_date_active	Null	Date	No	No
Lookup_code	Null	Varchar2	No	No
Lookup_type	Null	Varchar2	No	No
meaning	Null	Varchar2	No	No
Start_date_active	Null	Date	No	No

P_PTE_VAL_REC

The following table shows the parameters for this structure.

P_PTE_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
lookup	Null	Varchar2	No	No

The following table shows the parameters for the following table structures:

Table Structure Parameters

Table Name	Usage	Type
P_RQT_TBL	Null	TABLE OF Rqt_Rec_Type
P_RQT_VAL_TBL	Null	TABLE OF Rqt_Val_Rec_Type

Table Name	Usage	Type
P_SSC_TBL	Null	TABLE OF Ssc_Rec_Type
P_SSC_VAL_TBL	Null	TABLE OF Ssc_Val_Rec_Type
P_PSG_TBL	Null	TABLE OF Psg_Rec_Type
P_PSG_VAL_TBL	Null	TABLE OF Psg_Val_Rec_Type
P_SOU_TBL	Null	TABLE OF Sou_Rec_Type
P_SOU_VAL_TBL	Null	TABLE OF Sou_Val_Rec_Type
P_FNA_TBL	Null	TABLE OF Fna_Rec_Type
P_FNA_VAL_TBL	Null	TABLE OF Fna_Val_Rec_Type

X_PTE_REC_TYPE

The following table shows the parameters for this structure.

X_PTE_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Description	Null	Varchar2	No	No
Enabled_flag	Null	Varchar2	No	No
End_date_active	Null	Date	No	No
Lookup_code	Null	Varchar2	No	No
Lookup_type	Null	Varchar2	No	No
meaning	Null	Varchar2	No	No
Start_date_active	Null	Date	No	No

X_PTE_VAL_REC

The following table shows the parameters for this structure.

X_PTE_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
lookup	Null	Varchar2	No	No

The following table shows the parameters for the following table structures:

Table Structure Parameters

Table Name	Usage	Type
X_RQT_TBL	Null	TABLE OF Rqt_Rec_Type
X_RQT_VAL_TBL	Null	TABLE OF Rqt_Val_Rec_Type
P_SSC_TBL	Null	TABLE OF Ssc_Rec_Type
P_SSC_VAL_TBL	Null	TABLE OF Ssc_Val_Rec_Type
X_PSG_TBL	Null	TABLE OF Psg_Rec_Type
X_PSG_VAL_TBL	Null	TABLE OF Psg_Val_Rec_Type
X_SOU_TBL	Null	TABLE OF Sou_Val_Rec_Type
X_SOU_VAL_TBL	Null	TABLE OF Sou_Val_Type
P_FNA_TBL	Null	TABLE OF Fna_Rec_Type
P_FNA_VAL_TBL	Null	TABLE OF Fna_Val_Rec_Type

Procedure GET_ATTR_MAPPING

The following table shows the parameters for this structure. This API will fetch the records for an Source System, Request Type, PTE_Attribute link, Attribute Mapping

rules given the input parameter.

GET_ATTR_MAPPING Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	Yes	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
x_return_status	Out	Number	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Pte_Rec_Type	No	No
p_lookup_code	In	Varchar2	No	No
p_lookup	In	Varchar2	No	No
x_PTE_rec	Out	Pte_Rec_Type	No	No
x_PTE_val_rec	Out	Pte_Val_Rec_Type	No	No
x_RQT_tbl	Out	Rqt_Tbl_Type	No	No
x_RQT_val_tbl	Out	Rqt_Val_Tbl_Type	No	No
x_SSC_tbl	Out	Ssc_Tbl_Type	No	No
x_SSC_val_tbl	Out	Ssc_Val_Tbl_Type	No	No

Parameter	Usage	Type	Req	Drv
x_PSG_tbl	Out	Psg_Tbl_Type	No	No
x_PSG_val_tbl	Out	Psg_Val_Tbl_Type	No	No
x_SOU_tbl	Out	Sou_Tbl_Type	No	No
x_SOU_val_tbl	Out	Sou_Val_Tbl_Type	No	No
x_FNA_tbl	Out	Fna_Tbl_Type	No	No
x_FNA_val_tbl	Out	Fna_Val_Tbl_Type	No	No

p_api_version_number

This the version number of the API.

p_lookup_code

This is the code for the Pricing Transaction Entity; for example, ORDFUL for Order Fulfillment, DEMAND for Demand Planning etc.

p_lookup

This is the type of the Pricing Lookups. For Pricing Transaction Entities, it is QP_PTE_TYPE.

X_PTE_REC_TYPE

The following table shows the parameters for this structure.

X_PTE_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Description	Null	Varchar2	No	No
Enabled_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
End_date_active	Null	Date	No	No
Lookup_code	Null	Varchar2	No	No
Lookup_type	Null	Varchar2	No	No
meaning	Null	Varchar2	No	No
Start_date_active	Null	Date	No	No

X_PTE_VAL_REC

The following table shows the parameters for this structure.

X_PTE_VAL_REC Parameters

Parameter	Usage	Type	Req	Drv
enabled	Null	Varchar2	No	No
lookup	Null	Varchar2	No	No

The following table shows the parameters for the following table structures:

Table Structure Parameters

Table Name	Usage	Type
X_RQT_TBL	Null	TABLE OF Rqt_Rec_Type
X_RQT_VAL_TBL	Null	TABLE OF Rqt_Val_Rec_Type
X_SSC_TBL	Null	TABLE OF Ssc_Rec_Type
X_SSC_VAL_TBL	Null	TABLE OF Ssc_Val_Rec_Type
X_PSG_TBL	Null	TABLE OF Psg_Rec_Type

Table Name	Usage	Type
X_PSG_VAL_TBL	Null	TABLE OF Psg_Val_Rec_Type
X_SOU_TBL	Null	TABLE OF Sou_Rec_Type
X_SOU_VAL_TBL	Null	TABLE OF Sou_Val_Rec_Type
X_FNA_TBL	Null	TABLE OF Fna_rec_type
X_FNA_VAL_TBL	Null	TABLE OF Fna_Val_Rec_type

Business Object for Modifier Setup Application Program Interface

This section explains how to use the Business Object for Modifier Setup API and how it functions in Oracle Advanced Pricing. The Business Object for Modifier Setup package consists of entities to set up modifiers.

Functional Overview

The package QP_Modifiers_PUB.Process Modifiers contains the following public record type and table of records entities:

- **Process_Modifiers:** QP_Modifiers_PUB.Process_Modifiers is a Public API. It takes two record types and six table types as input parameters. Use this API to insert, update and delete modifiers or to set up a modifier list header for a given p_MODIFIER_LIST_rec record structure.

You can:

- Set up multiple modifier lines by giving multiple modifier definitions in the p_MODIFIERS_tbl table structure.
- Attach multiple qualifiers either at the header level (modifier list) or at the line level (modifier) by giving multiple qualifiers in the p_QUALIFIERS_tbl table structure.
- Attach multiple pricing attributes to modifier lines by giving the pricing attributes in the p_PRICING_ATTR_tbl table structure.
- **Modifier_List_Rec_Type:** Corresponds to the columns in the modifier header tables QP_LIST_HEADERS_B and QP_LIST_HEADERS_TL.
- **Modifier_List_Tbl_Type**

- `Modifier_List_Val_Rec_Type`: Corresponds to the columns in the modifier header table `QP_LIST_HEADERS_B`.
- `Modifier_List_Val_Tbl_Type`
- `Modifiers_Rec_Type`: Corresponds to the columns in the modifier and related modifiers tables `QP_LIST_LINES` and `QP_RLTD_MODIFIERS`.
- `Modifiers_Tbl_Type`
- `Modifiers_Val_Rec_Type`: Corresponds to the columns in the modifier table `QP_LIST_LINES`.
- `Modifiers_Val_Tbl_Type`
- `Qualifiers_Rec_Type`: Corresponds to the columns in the qualifier table `QP_QUALIFIERS`.
- `Qualifiers_Tbl_Type`
- `Qualifiers_Val_Rec_Type`: Corresponds to the columns in the qualifier table `QP_QUALIFIERS`.
- `Qualifiers_Val_Tbl_Type`
- `Pricing_Attr_Rec_Type`: Corresponds to the columns in the pricing attributes table `QP_PRICING_ATTRIBUTES`.
- `Pricing_Attr_Tbl_Type`
- `Pricing_Attr_Val_Rec_Type`: Corresponds to the columns in the pricing attributes table `QP_PRICING_ATTRIBUTES`.
- `Pricing_Attr_Val_Tbl_Type`
 - Currently, the `QP_Modifiers_PUB.Process_Modifiers` API does not evaluate or validate the required default attributes when a GSA price list is created. When you create a GSA price list using the `QP_Modifiers_PUB.Process_Modifiers` API, you need to initialize the modifier list line attributes in synchronization with the GSA Price List window.
 - Updating Modifier Records using `QP_GLOBALS.G_OPR_UPDATE`: If you get the error message "PL/SQL: Function returned without value in Package" when using the API `QP_GLOBALS.G_OPR_UPDATE` to update the modifier

records, then use the header id/line id of the modifier to be updated rather than the modifier name.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Business Object for Modifier Setup. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

Important: Date and DateTime fields

Dates and date times are stored in the following default canonical formats--use these formats to pass any Dates or Datetime fields to the API:

- Date: YYYY/MM/DD
- DateTime: YYYY/MM/DD HH:MM:SS

These formats should also be used for pricing and qualifier attribute values with the datatype of Date or DateTimes.

PROCESS_MODIFIERS

The following table shows the parameters for this structure.

PROCESS_MODIFIERS Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
x_msg_data	Out	Varchar2	No	No
p_MODIFIER_LI ST_rec	In	Modifier_List_ Rec_Type	No	No
p_MODIFIER_LI ST_val_rec	In	Modifier_List_ Val_Rec_Type	No	No
p_MODIFIERS_t bl	In	Modifiers_Tbl_ Type	No	No
p_MODIFIERS_ val_tbl	In	Modifiers_Val_ Tbl_Type	No	No
p_QUALIFIERS_ tbl	In	Qualifiers_Tbl_ Type	No	No
p_QUALIFIERS_ val_tbl	In	Qualifiers_Val_ Tbl_Type	No	No
p_PRICING_AT TR_tbl	In	Pricing_Attr_ Tbl_Type	No	No
p_PRICING_AT TR_val_tbl	In	Pricing_Attr_Val_ _Tbl_Type	No	No
x_MODIFIER_LI ST_rec	Out	Modifier_List_ Rec_Type	No	No
x_MODIFIER_LI ST_val_rec	Out	Modifier_List_ Val_Rec_Type	No	No
x_MODIFIERS_t bl	Out	Modifiers_Tbl_ Type	No	No
x_MODIFIERS_v al_tbl	Out	Modifiers_Val_ Tbl_Type	No	No
x_QUALIFIERS_ tbl	Out	Qualifiers_Tbl_ Type	No	No

Parameter	Usage	Type	Req	Drv
x_QUALIFIERS_val_tbl	Out	Qualifiers_Val_Tbl_Type	No	No
x_PRICING_ATTR_tbl	Out	Pricing_Attr_Tbl_Type	No	No
x_PRICING_ATTR_val_tbl	Out	Pricing_Attr_Val_Tbl_Type	No	No

p_init_sg_list

Default Value: FND_API.G_FALSE

p_return_values

Default Value: FND_API.G_FALSE

p_commit

Default Value: FND_API.G_FALSE

p_MODIFIER_LIST_rec

Default Value: G_MISS_MODIFIER_LIST_REC

p_MODIFIER_LIST_val_rec

Default Value: G_MISS_MODIFIER_LIST_VAL_REC

p_MODIFIERS_tbl

Default Value: G_MISS_MODIFIERS_TBL

p_MODIFIERS_val_tbl

Default Value: G_MISS_MODIFIERS_VAL_TBL

p_QUALIFIERS_tbl

Default Value: G_MISS_QUALIFIERS_TBL

p_QUALIFIERS_val_tbl

Default Value: G_MISS_QUALIFIERS_VAL_TBL

p_PRICING_ATTR_tbl

Default Value: G_MISS_PRICING_ATTR_TBL

p_PRICING_ATTR_val_tbl

Default Value: G_MISS_PRICING_ATTR_VAL_TBL

MODIFIER_LIST_REC_TYPE

The following table shows the parameters for this structure.

Note: For the structure MODIFIER_LIST_REC_TYPE, the ask_for_flag is only populated for modifier lists of type Promotion and Deal. This is not supported for Discount List, Surcharge List, or Freight and Special Charges modifier lists.

MODIFIER_LIST_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
currency_code	Null	Varchar2	Yes	No
discount_lines_flag	Null	Varchar2	No	No
end_date_active	Null	Date	Yes	No
freight_terms_code	Null	Varchar2	No	No
gsa_indicator	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_login	Null	Number	No	No
list_header_id	Null	Number	No	No
list_type_code	Null	Varchar2	Yes	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
prorate_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
request_id	Null	Number	No	No
rounding_factor	Null	Number	No	No
ship_method_code	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
terms_id	Null	Number	No	No
source_system_code	Null	Varchar2	Yes	No
active_flag	Null	Varchar2	Yes	No
parent_list_header_id	Null	Number	No	No
start_date_active_first	Null	Date	No	No
end_date_active_first	Null	Date	No	No
active_date_first_type	Null	Varchar2	No	No
start_date_active_second	Null	Date	No	No
end_date_active_second	Null	Date	No	No
active_date_second_type	Null	Varchar2	No	No
ask_for_flag	Null	Varchar2	No	No
return_status	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
db_flag	Null	Varchar2	No	No
version_no	Null	Varchar2	Yes	No
operation	Null	Varchar2	Yes	No
name	Null	Varchar2	Yes	No
pte_code	Null	Varchar2	Yes	No
description	Null	Varchar2	Yes	No
List_Source_code	Null	Varchar2	Yes	No
Orig_System_Header_Ref	Null	Varchar2	Yes	No
Shareable_flag	Null	Varchar2	Yes	No
Global_flag	Null	Varchar2	No	No
Org_id	Null	Number	No	Yes

attribute1-15

Default Value: FND_API.G_MISS_CHAR

automatic_flag

Default Value: FND_API.G_MISS_CHAR

comments

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

currency_code

Default Value: FND_API.G_MISS_CHAR

discount_lines_flag

Default Value: FND_API.G_MISS_CHAR

end_date_active

Default Value: FND_API.G_MISS_DATE

freight_terms_code

Default Value: FND_API.G_MISS_CHAR

gsa_indicator

Default Value: FND_API.G_MISS_CHAR

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

list_header_id

Default Value: FND_API.G_MISS_NUM

list_type_code

Default Value: FND_API.G_MISS_CHAR

program_application_id

Default Value: FND_API.G_MISS_NUM

program_id

Default Value: FND_API.G_MISS_NUM

program_update_date

Default Value: FND_API.G_MISS_DATE

prorate_flag

Default Value: FND_API.G_MISS_CHAR

request_id

Default Value: FND_API.G_MISS_NUM

rounding_factor

Default Value: FND_API.G_MISS_NUM

ship_method_code

Default Value: FND_API.G_MISS_CHAR

start_date_active

Default Value: FND_API.G_MISS_DATE

terms_id

Default Value: FND_API.G_MISS_NUM

source_system_code

Default Value: FND_API.G_MISS_CHAR

active_flag

Default Value: FND_API.G_MISS_CHAR

parent_list_header_id

Default Value: FND_API.G_MISS_NUM

start_date_active_first

Default Value: FND_API.G_MISS_DATE

end_date_active_first

Default Value: FND_API.G_MISS_DATE

active_date_first_type

Default Value: FND_API.G_MISS_CHAR

start_date_active_second

Default Value: FND_API.G_MISS_DATE

end_date_active_second

Default Value: FND_API.G_MISS_DATE

active_date_second_type

Default Value: FND_API.G_MISS_CHAR

ask_for_flag

Default Value: FND_API.G_MISS_CHAR

return_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

name

Default Value: FND_API.G_MISS_CHAR

description

Default Value: FND_API.G_MISS_CHAR

List_Source_code

Default Value: FND_API.G_MISS_CHAR

Orig_System_Header_Ref

Default Value: FND_API.G_MISS_CHAR

Shareable_flag

Default Value: FND_API.G_MISS_CHAR

Global_flag

Default Value: Y

Org_Id

Default Value: FND_API.G_MISS_NUM

Note: This API will initialize MOAC if it is not initialized prior. On the modifier_list_rec, if the global_flag is passed as FND_API.G_MISS_CHAR, the API will default the global_flag as 'Y'. If global_flag is passed as 'N' and if org_id is not null, the API will create the modifier list for that operating unit (OU). If global_flag is passed as "N" and if org_id is not passed, the API will derive the org_id from the org context if set; if not, use the default OU for that responsibility. The API will validate the Org_id to check if the Org_Id corresponds to one of the OUs for that responsibility and raise an error if org_id is not valid.

MODIFIER_LIST_TBL_TYPE

The following table shows the parameters for this structure.

MODIFIER_LIST_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Modifier_List_Record_Type	Null	Record	No	No
List_Source_code	Null	Varchar2	Yes	No

MODIFIER_LIST_VAL_REC_TYPE

The following table shows the parameters for this structure.

MODIFIER_LIST_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
automatic	Null	Varchar2	No	No
currency	Null	Varchar2	No	No
discount_lines	Null	Varchar2	No	No
freight_terms	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_type	Null	Varchar2	No	No
prorate	Null	Varchar2	No	No
ship_method	Null	Varchar2	No	No
terms	Null	Varchar2	No	No
List_Source_code	Null	Varchar2	Yes	No

automatic

Default Value: FND_API.G_MISS_CHAR

currency

Default Value: FND_API.G_MISS_CHAR

discount_lines

Default Value: FND_API.G_MISS_CHAR

freight_terms

Default Value: FND_API.G_MISS_CHAR

list_header

Default Value: FND_API.G_MISS_CHAR

list_type

Default Value: FND_API.G_MISS_CHAR

prorate

Default Value: FND_API.G_MISS_CHAR

ship_method

Default Value: FND_API.G_MISS_CHAR

terms

Default Value: FND_API.G_MISS_CHAR

List_Source_code

Default Value: FND_API.G_MISS_CHAR

MODIFIER_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

MODIFIER_LIST_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Modifier_List_Val_Rec_Type	Null	Record	No	No

MODIFIERS_REC_TYPE

The following table shows the parameters for this structure:

MODIFIERS_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
arithmetic_operator	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	Yes	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
effective_period_uom	Null	Varchar2	No	No
end_date_active	Null	Date	Yes	No
estim_accrual_rate	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
generate_using_formula_id	Null	Number	No	No
inventory_item_id	Null	Number	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
list_line_type_co de	Null	Varchar2	Yes	No
list_price	Null	Number	No	No
modifier_level_c ode	Null	Varchar2	Yes	No
number_effectiv e_periods	Null	Number	No	No
operand	Null	Number	No	No
organization_id	Null	Number	No	No
override_flag	Null	Varchar2	No	No
percent_price	Null	Number	No	No
price_break_type _code	Null	Varchar2	Yes	No
price_by_formul a_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
print_on_invoice_flag	Null	Varchar2	No	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
rebate_trxn_type_code	Null	Varchar2	No	No
related_item_id	Null	Number	No	No
relationship_type_id	Null	Number	No	No
request_id	Null	Number	No	No
revision	Null	Varchar2	No	No
revision_date	Null	Date	No	No
revision_reason_code	Null	Varchar2	No	No
start_date_active	Null	Date	Yes	No
substitution_attribute	Null	Varchar2	No	No
substitution_context	Null	Varchar2	No	No
substitution_value	Null	Varchar2	No	No
accrual_flag	Null	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
pricing_group_sequence	Null	Number	Yes	No
incompatibility_grp_code	Null	Varchar2	No	No
list_line_no	Null	Varchar2	No	No
from_rltd_modifier_id	Null	Number	No	No
to_rltd_modifier_id	Null	Number	No	No
rltd_modifier_group_no	Null	Number	No	No
rltd_modifier_group_type	Null	Varchar2	No	No
pricing_phase_id	Null	Number	Yes	No
product_precedence	Null	Number	Yes	No
expiration_period_start_date	Null	Date	No	No
number_expiration_periods	Null	Number	No	No
expiration_period_uom	Null	Varchar2	No	No
expiration_date	Null	Date	No	No
estim_gl_value	Null	Number	No	No
benefit_price_list_line_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
benefit_limit	Null	Number	No	No
charge_type_code	Null	Varchar2	No	No
charge_subtype_code	Null	Varchar2	No	No
benefit_qty	Null	Number	No	No
benefit_uom_code	Null	Varchar2	No	No
accrual_conversion_rate	Null	Number	No	No
proration_type_code	Null	Varchar2	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No

arithmetic_operator

Default Value: FND_API.G_MISS_CHAR

attribute1-15

Default Value: FND_API.G_MISS_CHAR

automatic_flag

Default Value: FND_API.G_MISS_CHAR

comments

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

effective_period_uom

Default Value: FND_API.G_MISS_CHAR

end_date_active

Default Value: FND_API.G_MISS_DATE

estim_accrual_rate

Default Value: FND_API.G_MISS_NUM

generate_using_formula_id

Default Value: FND_API.G_MISS_NUM

inventory_item_id

Default Value: FND_API.G_MISS_NUM

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

list_header_id

Default Value: FND_API.G_MISS_NUM

list_line_id

Default Value: FND_API.G_MISS_NUM

list_line_type_code

Default Value: FND_API.G_MISS_CHAR

list_price

Default Value: FND_API.G_MISS_NUM

modifier_level_code

Default Value: FND_API.G_MISS_CHAR

number_effective_periods

Default Value: FND_API.G_MISS_NUM

operand

Default Value: FND_API.G_MISS_NUM

organization_id

Default Value: FND_API.G_MISS_NUM

override_flag

Default Value: FND_API.G_MISS_CHAR

percent_price

Default Value: FND_API.G_MISS_NUM

price_break_type_code

Default Value: FND_API.G_MISS_CHAR

price_by_formula_id

Default Value: FND_API.G_MISS_NUM

primary_uom_flag

Default Value: FND_API.G_MISS_CHAR

print_on_invoice_flag

Default Value: FND_API.G_MISS_CHAR

program_application_id

Default Value: FND_API.G_MISS_NUM

program_id

Default Value: FND_API.G_MISS_NUM

program_update_date

Default Value: FND_API.G_MISS_DATE

rebate_trxn_type_code

Default Value: FND_API.G_MISS_CHAR

related_item_id

Default Value: FND_API.G_MISS_NUM

relationship_type_id

Default Value: FND_API.G_MISS_NUM

reprice_flag

Default Value: FND_API.G_MISS_CHAR

request_id

Default Value: FND_API.G_MISS_NUM

revision

Default Value: FND_API.G_MISS_CHAR

revision_date

Default Value: FND_API.G_MISS_DATE

revision_reason_code

Default Value: FND_API.G_MISS_CHAR

start_date_active

Default Value: FND_API.G_MISS_DATE

substitution_attribute

Default Value: FND_API.G_MISS_CHAR

substitution_context

Default Value: FND_API.G_MISS_CHAR

substitution_value

Default Value: FND_API.G_MISS_CHAR

accrual_flag

Default Value: FND_API.G_MISS_CHAR

pricing_group_sequence

Default Value: FND_API.G_MISS_NUM

incompatibility_grp_code

Default Value: FND_API.G_MISS_CHAR

list_line_no

Default Value: FND_API.G_MISS_CHAR

from_rltd_modifier_id

Default Value: FND_API.G_MISS_NUM

to_rltd_modifier_id

Default Value: FND_API.G_MISS_NUM

rltd_modifier_grp_no

Default Value: FND_API.G_MISS_NUM

rltd_modifier_grp_type

Default Value: FND_API.G_MISS_CHAR

pricing_phase_id

Default Value: FND_API.G_MISS_NUM

product_precedence

Default Value: FND_API.G_MISS_NUM

expiration_period_start_date

Default Value: FND_API.G_MISS_DATE

number_expiration_periods

Default Value: FND_API.G_MISS_NUM

expiration_period_uom

Default Value: FND_API.G_MISS_CHAR

expiration_date

Default Value: FND_API.G_MISS_DATE

estim_gl_value

Default Value: FND_API.G_MISS_NUM

benefit_price_list_line_id

Default Value: FND_API.G_MISS_NUM

benefit_limit

Default Value: FND_API.G_MISS_NUM

charge_type_code

Default Value: FND_API.G_MISS_CHAR

charge_subtype_code

Default Value: FND_API.G_MISS_CHAR

benefit_qty

Default Value: FND_API.G_MISS_NUM

benefit_uom_code

Default Value: FND_API.G_MISS_CHAR

accrual_conversion_rate

Default Value: FND_API.G_MISS_NUM

proration_type_code

Default Value: FND_API.G_MISS_CHAR

return_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

MODIFIERS_TBL_TYPE

The following table shows the parameters for this structure.

MODIFIERS_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Modifiers_Rec_Type	Null	Record	No	No

MODIFIERS_VAL_REC_TYPE

The following table shows the parameters for this structure.

MODIFIERS_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accrual_type	Null	Varchar2	No	No
accrual_uom	Null	Varchar2	No	No
automatic	Null	Varchar2	No	No
generate_using_formula	Null	Varchar2	No	No
gl_class	Null	Varchar2	No	No
inventory_item	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
list_line_type	Null	Varchar2	No	No
list_price_uom	Null	Varchar2	No	No
modifier_level	Null	Varchar2	No	No
organization	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
override	Null	Varchar2	No	No
price_break_type	Null	Varchar2	No	No
price_by_formula	Null	Varchar2	No	No
primary_uom	Null	Varchar2	No	No
print_on_invoice	Null	Varchar2	No	No
rebate_subtype	Null	Varchar2	No	No
rebate_transaction_type	Null	Varchar2	No	No
related_item	Null	Varchar2	No	No
relationship_type	Null	Varchar2	No	No
reprice	Null	Varchar2	No	No
revision_reason	Null	Varchar2	No	No

accrual_type

Default Value: FND_API.G_MISS_CHAR

accrual_uom

Default Value: FND_API.G_MISS_CHAR

automatic

Default Value: FND_API.G_MISS_CHAR

generate_using_formula

Default Value: FND_API.G_MISS_CHAR

gl_class

Default Value: FND_API.G_MISS_CHAR

inventory_item

Default Value: FND_API.G_MISS_CHAR

list_header

Default Value: FND_API.G_MISS_CHAR

list_line

Default Value: FND_API.G_MISS_CHAR

list_line_type

Default Value: FND_API.G_MISS_CHAR

list_price_uom

Default Value: FND_API.G_MISS_CHAR

modifier_level

Default Value: FND_API.G_MISS_CHAR

organization

Default Value: FND_API.G_MISS_CHAR

override

Default Value: FND_API.G_MISS_CHAR

price_break_type

Default Value: FND_API.G_MISS_CHAR

price_by_formula

Default Value: FND_API.G_MISS_CHAR

primary_uom

Default Value: FND_API.G_MISS_CHAR

print_on_invoice

Default Value: FND_API.G_MISS_CHAR

rebate_subtype

Default Value: FND_API.G_MISS_CHAR

rebate_transaction_type

Default Value: FND_API.G_MISS_CHAR

related_item

Default Value: FND_API.G_MISS_CHAR

relationship_type

Default Value: FND_API.G_MISS_CHAR

reprice

Default Value: FND_API.G_MISS_CHAR

revision_reason

Default Value: FND_API.G_MISS_CHAR

MODIFIERS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

MODIFIERS_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Modifiers_Val_Record_Type	Null	Record	No	No

QUALIFIERS_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
comparison_operator_code	Null	Varchar2	Yes	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
created_from_rule_id	Null	Number	No	No
creation_date	Null	Date	No	No
end_date_active	Null	Date	No	No
excluder_flag	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
program_update_date	Null	Date	No	No
qualifier_attribute	Null	Varchar2	Yes	No
qualifier_attr_value	Null	Varchar2	Yes	No
qualifier_context	Null	Varchar2	Yes	No
qualifier_grouping_no	Null	Number	Yes	No
qualifier_precedence	Null	Number	Yes	No
qualifier_id	Null	Number	No	No
qualifier_rule_id	Null	Number	No	No
request_id	Null	Number	No	No
start_date_active	Null	Date	Yes	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No

attribute1-15

Default Value: FND_API.G_MISS_CHAR

comparison_operator_code

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

created_from_rule_id

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

end_date_active

Default Value: FND_API.G_MISS_DATE

excluder_flag

Default Value: FND_API.G_MISS_CHAR

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

list_header_id

Default Value: FND_API.G_MISS_NUM

list_line_id

Default Value: FND_API.G_MISS_NUM

program_application_id

Default Value: FND_API.G_MISS_NUM

program_id

Default Value: FND_API.G_MISS_NUM

program_update_date

Default Value: FND_API.G_MISS_DATE

qualifier_attribute

Default Value: FND_API.G_MISS_CHAR

qualifier_attr_value

Default Value: FND_API.G_MISS_CHAR

qualifier_context

Default Value: FND_API.G_MISS_CHAR

qualifier_grouping_no

Default Value: FND_API.G_MISS_NUM

qualifier_id

Default Value: FND_API.G_MISS_NUM

qualifier_rule_id

Default Value: FND_API.G_MISS_NUM

request_id

Default Value: FND_API.G_MISS_NUM

start_date_active

Default Value: FND_API.G_MISS_DATE

return_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

QUALIFIERS_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Qualifiers_Rec_Type	Null	Record	No	No

QUALIFIERS_VAL_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
comparison_operator	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
created_from_rule	Null	Varchar2	No	No
excluder	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
qualifier	Null	Varchar2	No	No
qualifier_rule	Null	Varchar2	No	No

comparison_operator

Default Value: FND_API.G_MISS_CHAR

created_from_rule

Default Value: FND_API.G_MISS_CHAR

excluder

Default Value: FND_API.G_MISS_CHAR

list_header

Default Value: FND_API.G_MISS_CHAR

list_line

Default Value: FND_API.G_MISS_CHAR

qualifier

Default Value: FND_API.G_MISS_CHAR

qualifier_rule

Default Value: FND_API.G_MISS_CHAR

QUALIFIERS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Qualifiers_Val_Record_Type	Null	Record	No	No

PRICING_ATTR_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accumulate_flag	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
attribute_grouping_no	Null	Number	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
excluder_flag	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_line_id	Null	Number	No	No
pricing_attribute	Null	Varchar2	No	No
pricing_attribute _context	Null	Varchar2	No	No
pricing_attribute _id	Null	Number	No	No
pricing_attr_valu e_from	Null	Varchar2	No	No
pricing_attr_valu e_to	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
product_attribute	Null	Varchar2	Yes	No
product_attribute_context	Null	Varchar2	Yes	No
product_attribute_value	Null	Varchar2	Yes	No
product_uom_code	Null	Varchar2	Yes	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
product_attribute_datatype	Null	Varchar2	No	No
pricing_attribute_datatype	Null	Varchar2	No	No
comparison_operator_code	Null	Varchar2	Yes	No
request_id	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No
MODIFIERS_index	Null	Number	Yes	No

accumulate_flag

Default Value: FND_API.G_MISS_CHAR

attribute1-15

Default Value: FND_API.G_MISS_CHAR

attribute_grouping_no

Default Value: FND_API.G_MISS_NUM

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

excluder_flag

Default Value: FND_API.G_MISS_CHAR

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

list_line_id

Default Value: FND_API.G_MISS_NUM

pricing_attribute

Default Value: FND_API.G_MISS_CHAR

pricing_attribute_context

Default Value: FND_API.G_MISS_CHAR

pricing_attribute_id

Default Value: FND_API.G_MISS_NUM

pricing_attr_value_from

Default Value: FND_API.G_MISS_CHAR

pricing_attr_value_to

Default Value: FND_API.G_MISS_CHAR

product_attribute

Default Value: FND_API.G_MISS_CHAR

product_attribute_context

Default Value: FND_API.G_MISS_CHAR

product_attr_value

Default Value: FND_API.G_MISS_CHAR

product_uom_code

Default Value: FND_API.G_MISS_CHAR

program_application_id

Default Value: FND_API.G_MISS_NUM

program_id

Default Value: FND_API.G_MISS_NUM

program_update_date

Default Value: FND_API.G_MISS_DATE

product_attribute_datatype

Default Value: FND_API.G_MISS_CHAR

pricing_attribute_datatype

Default Value: FND_API.G_MISS_CHAR

comparison_operator_code

Default Value: FND_API.G_MISS_CHAR

request_id

Default Value: FND_API.G_MISS_NUM

return_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

MODIFIERS_index

Default Value: FND_API.G_MISS_NUM

PRICING_ATTR_TBL_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Rec_Type	Null	Record	No	No

PRICING_ATTR_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
accumulate	Null	Varchar2	No	No
excluder	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
pricing_attribute	Null	Varchar2	No	No
product_uom	Null	Varchar2	No	No

accumulate

Default Value: FND_API.G_MISS_CHAR

excluder

Default Value: FND_API.G_MISS_CHAR

list_line

Default Value: FND_API.G_MISS_CHAR

pricing_attribute

Default Value: FND_API.G_MISS_CHAR

product_uom

Default Value: FND_API.G_MISS_CHAR

PRICING_ATTR_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Val _Rec_Type	Null	Record	No	No

Validation of Business Object for Modifier Setup API

Standard Validation

Oracle Advanced Pricing validates all required columns in the Business Object for Modifier Setup API.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Business Object for Modifier Setup API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Example of Modifier Setup Application Program Interface

The following example scripts of the Modifier Setup Application Program Interface are located in the directory \$qp/patch/115/sql:

- QPXEXDS1.sql: This script provides a line level discount of 8% on all products.
- QPXEXDS2.sql: This script provides an 8% discount when you buy more than 5 quantities of item 62081.
- QPXEXDS3.sql: This script provides a 10% discount when customer "1000" buys more than 2 units of item 62081.
- QPXEXSUR.sql: This script charges a 2% surcharge on all products.

- QPXEXTSN.sql: This script provides Payment Terms of 2/10 NET 30 if customer "1000" buys more than 2 units of item 62081.
- QPXEXOID.sql: This script provides an other item discount: buy 1 of item 62081, get 1 free
- QPXEXPRG.sql: This script provides a promotional good: buy 1 of item 45, and 1 of item 63, then get 1 item 62081 at 20% discount.
- QPXEXCIE.sql: This script provides a coupon issue: buy 2 of item 45, get coupon for 20% discount.
- QPXEXPBH.sql: This script provides a price break: buy 1-100 of item 45, get 20% discount.

Business Object for Pricing Formulas Application Program Interface

This section explains how to use the Business Object for Pricing Formulas API and how it functions in Oracle Advanced Pricing. The Business Object for Pricing Formulas package consists of entities to support the Formulas window.

Functional Overview

The Formulas window is based on the following APIs. However, the modal Formula Factors window is based on the Modifiers public API QP_Modifiers_PUB.Process_Modifiers.

The package QP_Price_Formula_PUB contains the following public record type and table of records definitions:

- Formula_Rec_Type: A record type corresponding to the columns in the Formula Headers view (QP_PRICE_FORMULAS_VL).
- Formula_Tbl_Type
- Formula_Val_Rec_Type: A record type used to store values corresponding to IDs in the formula record.
- Formula_Val_Tbl_Type
- Formula_Lines_Rec_Type: A record type corresponding to the columns in the Formula Lines table (QP_PRICE_FORMULA_LINES).
- Formula_Lines_Tbl_Type
- Formula_Lines_Val_Rec_Type: A record type used to store values corresponding to IDs in the Formula Lines record.

- Formula_Lines_Val_Tbl_Type
- QP_Price_formula_PUB.Process_Price_Formula: Performs the insert, update, and delete of price formula header and price formula lines.
- QP_Price_formula_PUB.Lock_Price_Formula: Locks price formula header and price formula lines records prior to updates.
- QP_Price_formula_PUB.Get_Price_Formula: Retrieves the price formula header and lines for a given formula.

Setting Up and Parameter Descriptions

The following chart describes all parameters and the inbound and outbound parameters. Additional information on these parameters follows.

Important: Date and DateTime fields

Dates and date times are stored in the following default canonical formats--use these formats to pass any Dates or Datetime fields to the API:

- Date: YYYY/MM/DD
- DateTime: YYYY/MM/DD HH:MM:SS

These formats should also be used for pricing and qualifier attribute values with the datatype of Date or DateTimes.

QP_PRICE_FORMULA.PROCESS_PRICE_FORMULA

The following table shows the parameters for this structure.

QP_PRICE_FORMULA.PROCESS_PRICE_FORMULA Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
p_commit	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Number	No	No
x_msg_data	Out	Varchar2	No	No
p_FORMULA_rec	In	Formula_Rec_Type	No	No
p_FORMULA_val_rec	In	Formula_Val_Rec_Type	No	No
p_FORMULA_LINES_tbl	In	Formula_Lines_Tbl_Type	No	No
p_FORMULA_LINES_val_tbl	In	Formula_Lines_Val_Tbl_Type	No	No
x_FORMULA_rec	Out	Formula_Rec_Type	No	No
x_FORMULA_val_rec	Out	Formula_Val_Rec_Type	No	No
x_FORMULA_LINES_tbl	Out	Formula_Lines_Tbl_Type	No	No
x_FORMULA_LINES_val_tbl	Out	Formula_Lines_Val_Tbl_Type	No	No

p_init_msg_list

Default Value: FND_API.G_FALSE

p_return_values

Default Value: FND_API.G_FALSE

p_commit

Default Value: FND_API.G_FALSE

p_FORMULA_rec

Default Value: G_MISS_FORMULA_REC

p_FORMULA_val_rec

Default Value: G_MISS_FORMULA_VAL_REC

p_FORMULA_LINES_tbl

Default Value: G_MISS_FORMULA_LINES_TBL

p_FORMULA_LINES_val_tbl

Default Value: G_MISS_FORMULA_LINES_VAL_TBL

FORMULA_REC_TYPE

The following table shows the parameters for this structure:

FORMULA_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
description	Null	Varchar2	No	No
end_date_active	Null	Date	No	No
formula	Null	Varchar2	Yes	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
name	Null	Varchar2	Yes	No
price_formula_id	Null	Number	No	No
start_date_active	Null	Date	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No

attribute1-15

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

description

Default Value: FND_API.G_MISS_CHAR

end_date_active

Default Value: FND_API.G_MISS_DATE

formula

Default Value: FND_API.G_MISS_CHAR

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

name

Default Value: FND_API.G_MISS_CHAR

price_formula_id

Default Value: Comes from the sequence QP_PRICE_FORMULAS_B_S

start_date_active

Default Value: FND_API.G_MISS_DATE

step_number

Default Value: FND_API.G_MISS_NUM

retun_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

FORMULA_TBL_TYPE

The following table shows the parameters for this structure.

FORMULA_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Formula_Rec_Type	Null	Record	No	No

FORMULA_VAL_REC_TYPE

The following table shows the parameters for this structure.

FORMULA_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
price_formula	Null	Number	No	No

price formula

Default Value: FND_API.G_MISS_CHAR

FORMULA_VAL_TBL_TYPE

The following table shows the parameters for this structure.

FORMULA_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Formula_Val_Rec_Type	Null	Record	No	No

FORMULA_LINES_REC_TYPE

The following table shows the parameters for this structure.

FORMULA_LINES_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
end_date_active	Null	Date	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
numeric_constant	Null	Number	Yes1	No
price_formula_id	Null	Number	No	No
price_formula_line_id	Null	Number	No	No
formula_line_type_code	Null	Varchar2	Yes	No
price_list_line_id	Null	Number	Yes2	No
price_modifier_list_id	Null	Number	Yes3	No
pricing_attribute	Null	Varchar2	Yes4	No
pricing_attribute_context	Null	Varchar2	Yes5	No
start_date_active	Null	Date	No	No
step_number	Null	number	Yes	No
return_status	Null	Varchar2	No	No
reqd_flag	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No

The following table describes notations listed in the preceding table:

Notations

Note	Description
1	Conditionally Required when formula_line_type_code is 'NUM'
2	Conditionally Required when formula_line_type_code is 'PLL'
3	Conditionally Required when formula_line_type_code is 'ML'
4	Conditionally Required when formula_line_type_code is 'PRA'
5	Conditionally Required when formula_line_type_code is 'PRA'

attribute1-15

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

end_date_active

Default Value: FND_API.G_MISS_DATE

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

numeric_constant

Default Value: FND_API.G_MISS_NUM

price_formula_id

Default Value: FND_API.G_MISS_NUM

price_formula_line_id

Default Value: Comes for the sequence QP_PRICE_FORMULA_LINES_S

formula_line_type_code

Default Value: FND_API.G_MISS_CHAR

price_list_line_id

Default Value: FND_API.G_MISS_NUM

price_modifier_list_id

Default Value: FND_API.G_MISS_NUM

pricing_attribute

Default Value: FND_API.G_MISS_CHAR

pricing_attribute_context

Default Value: FND_API.G_MISS_CHAR

start_date_active

Default Value: FND_API.G_MISS_DATE

step_number

Default Value: FND_API.G_MISS_NUM

reqd_flag

Default Value: FND_API.G_MISS_CHAR

retun_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

FORMULA_LINES_TBL_TYPE

The following table shows the parameters for this structure.

FORMULA_LINES_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Formula_Lines_ Rec_Type	Null	Record	No	No

FORMULA_LINES_VAL_REC_TYPE

The following table shows the parameters for this structure.

FORMULA_LINES_VAL_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
price_formula	Null	Varchar2	No	No
price_formula_line	Null	Varchar2	No	No
price_formula_line_type	Null	Varchar2	No	No
price_list_line	Null	Varchar2	No	No
price_modifier_list	Null	Varchar2	No	No

price_formula

Default Value: FND_API.G_MISS_CHAR

price_formula_line

Default Value: FND_API.G_MISS_CHAR

price_formula_line_type

Default Value: FND_API.G_MISS_CHAR

price_list_line

Default Value: FND_API.G_MISS_CHAR

price_modifier_list

Default Value: FND_API.G_MISS_CHAR

FORMULA_LINES_VAL_TBL_TYPE

The following table shows the parameters for this structure.

FORMULA_LINES_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Formula_Lines_ Val_Rec_Type	Null	Record	No	No

Validation of Business Object for Pricing Formulas API

Validations:

Standard Validation

Oracle Advanced Pricing validates all required columns in the Business Object for Pricing Formulas API. For more information, see: *Oracle Pricing Technical Reference Manual*.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Business Object for Pricing Formulas API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Example of Pricing Formulas API

The following example script is located in the directory \$qp/patch/115/sql:

QPPFXMP1.sql: This sample script inserts a pricing formula with seven different pricing formula lines to demonstrate the six formula line types supported in Advanced Pricing formulas:

- Price List Line (PLL)

- Function (FUNC)
- List Price (LP)
- Numeric Constant (NUM)
- Pricing Attribute (PRA)
- Factor List (ML)
- Modifier Value (MV)

Business Object for Pricing Limits Application Program Interface

This section explains how to use the Business Object for Pricing Limits API and how it functions in Oracle Advanced Pricing. The Business Object for Pricing Limits package consists of entities to support the Limits window.

Functional Overview

The Limits window is based on the following APIs:

The package QP_Limits_PUB contains the following public record type and table of records definitions:

- **Limits_Rec_Type**: A record type corresponding to the columns in the Pricing Limits view (QP_LIMITS).
- **Limits_Tbl_Type**
- **Limits_Val_Rec_Type**: A record type used to store values corresponding to IDs in the Limits record.
- **Limits_Val_Tbl_Type**
- **Limit_Attrs_Rec_Type**: A record type corresponding to the columns in the Limit Attributes. View (QP_LIMIT_ATTRIBUTES)
- **Limit_Attrs_Tbl_Type**
- **Limit_Attrs_Val_Rec_Type**: A record type used to store values corresponding to IDs in the Limits Attributes record.
- **Limit_Attrs_Val_Tbl_Type**
- **Limit_Balances_Rec_Type** : A record type corresponding to the columns in the Limit Balances. View (QP_LIMIT_BALANCES)

- Limit_Balances_Tbl_Type
- Limit_Balances_Val_Rec_Type : A record type used to store values corresponding to IDs in the Limits Balances record.
- Limit_Balances_Val_Tbl_Type

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public API QP_LIMITS_PUB.PROCESS_LIMITS. All of the inbound and outbound parameters are listed.

Important: Date and DateTime fields

Dates and date times are stored in the following default canonical formats--use these formats to pass any Dates or Datetime fields to the API:

- Date: YYYY/MM/DD
- DateTime: YYYY/MM/DD HH:MM:SS

These formats should also be used for pricing and qualifier attribute values with the datatype of Date or DateTimes.

PROCEDURE Process_Limits

Additional information on these parameters follows.

QP_LIMITS_PUB.PROCESS_LIMITS Parameters

Parameter	Usage	Type	Default
p_api_version_number	IN	Number	
p_init_msg_list	IN	Varchar2	FND_API.G_FALSE
p_return_values	IN	Varchar2	FND_API.G_FALSE
p_commit	IN	Varchar2	FND_API.G_FALSE
x_return_status	OUT	Varchar2	

Parameter	Usage	Type	Default
x_msg_count	OUT	Number	
x_msg_data	OUT	Varchar2	
p_LIMITS_rec	IN	Limits_Rec_Type	G_MISS_LIMITS_REC
p_LIMITS_val_rec	IN	Limits_Val_Rec_Type	G_MISS_LIMITS_VAL_REC
p_LIMIT_ATTRS_tbl	IN	Limit_Attrs_Tbl_Type	G_MISS_LIMIT_ATTRS_TBL
p_LIMIT_ATTRS_val_tbl	IN	Limit_Attrs_Val_Tbl_Type	G_MISS_LIMIT_ATTRS_VAL_TBL
p_LIMIT_BALANCE_S_tbl	IN	Limit_Balances_Tbl_Type	G_MISS_LIMIT_BALANCES_TBL
p_LIMIT_BALANCE_S_val_tbl	IN	Limit_Balances_Val_Tbl_Type	G_MISS_LIMIT_BALANCES_VAL_TBL
x_LIMITS_rec	OUT	Limits_Rec_Type	
x_LIMITS_val_rec	OUT	Limits_Val_Rec_Type	
x_LIMIT_ATTRS_tbl	OUT	Limit_Attrs_Tbl_Type	
x_LIMIT_ATTRS_val_tbl	OUT	Limit_Attrs_Val_Tbl_Type	
x_LIMIT_BALANCES_tbl	OUT	Limit_Balances_Tbl_Type	
x_LIMIT_BALANCES_val_tbl	OUT	Limit_Balances_Val_Tbl_Type	

PROCEDURE Lock_Limits

The following chart describes all parameters used by the public API QP_LIMITS_PUB.LOCK_LIMITS. All of the inbound and outbound parameters are

listed. Additional information on these parameters follows.

QP_LIMITS_PUB.LOCK_LIMITS Parameters

Parameter	Usage	Type	Default
p_api_version_number	IN	Number	
p_init_msg_list	IN	Varchar2	FND_API.G_FALSE
p_return_values	IN	Varchar2	FND_API.G_FALSE
x_return_status	OUT	Varchar2	
x_msg_count	OUT	Number	
x_msg_data	OUT	Varchar2	
p_LIMITS_rec	IN	Limits_Rec_Type	G_MISS_LIMITS_REC
p_LIMITS_val_rec	IN	Limits_Val_Rec_Type	G_MISS_LIMITS_VAL_REC
p_LIMIT_ATTRS_tbl	IN	Limit_Attrs_Tbl_Type	G_MISS_LIMIT_ATTRS_TBL
p_LIMIT_ATTRS_val_tbl	IN	Limit_Attrs_Val_Tbl_Type	G_MISS_LIMIT_ATTRS_VAL_TBL
p_LIMIT_BALANCE_S_tbl	IN	Limit_Balances_Tbl_Type	G_MISS_LIMIT_BALANCES_TBL
p_LIMIT_BALANCE_S_val_tbl	IN	Limit_Balances_Val_Tbl_Type	G_MISS_LIMIT_BALANCES_VAL_TBL
x_LIMITS_rec	OUT	Limits_Rec_Type	
x_LIMITS_val_rec	OUT	Limits_Val_Rec_Type	
x_LIMIT_ATTRS_tbl	OUT	Limit_Attrs_Tbl_Type	

Parameter	Usage	Type	Default
x_LIMIT_ATTRS_val_tbl	OUT	Limit_Attrs_Val_Tbl_Type	
x_LIMIT_BALANCES_tbl	OUT	Limit_Balances_Tbl_Type	
x_LIMIT_BALANCES_val_tbl	OUT	Limit_Balances_Val_Tbl_Type	

PROCEDURE Get_Limits

The following chart describes all parameters used by the public API QP_LIMITS_PUB.GET_LIMITS. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

QP_LIMITS_PUB.GET_LIMITS Parameters

Parameter	Usage	Type	Req	Default
p_api_version_number	IN	Number	Yes	
p_init_msg_list	IN	Varchar2	No	FND_API.G_FALSE
p_return_values	IN	Varchar2	No	FND_API.G_FALSE
x_return_status	OUT	Varchar2	No	
x_msg_count	OUT	Number	No	
x_msg_data	OUT	Varchar2	No	
p_limit_id	IN	Number	No	FND_API.G_MISS_NUM
p_limit	IN	Varchar2	No	FND_API.G_MISS_CHAR

Parameter	Usage	Type	Req	Default
x_LIMITS_rec	OUT	Limits_Rec_Type	No	
x_LIMITS_val_rec	OUT	Limits_Val_Rec_Type	No	
x_LIMIT_ATTRS_tbl	OUT	Limit_Attrs_Tbl_Type	No	
x_LIMIT_ATTRS_val_tbl	OUT	Limit_Attrs_Val_Tbl_Type	No	
x_LIMIT_BALANCES_tbl	OUT	Limit_Balances_Tbl_Type	No	
x_LIMIT_BALANCES_val_tbl	OUT	Limit_Balances_Val_Tbl_Type	No	

PL/SQL Record Structures

For each column of the PL/SQL record structure, the following information has been documented:

- *Datatype*: data type for this field
- *Req*: X if required at entry, *Blank* if Optional
- *Drv*: For internal use, users cannot update these fields
- *Default*: Defaulted value for this field

Limits_Rec_Type

For column descriptions, please refer to the Oracle Pricing TRM for the table QP_LIMITS.

The Derived value for the following parameters is Null.

QP_LIMITS

Parameter	Type	Req	Default
attribute1	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute2	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute3	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute4	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute5	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute6	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute7	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute8	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute9	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute10	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute11	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute12	Varchar2(240)	No	FND_API.G_MISS_C HAR
attribute13	Varchar2(240)	No	FND_API.G_MISS_C HAR

Parameter	Type	Req	Default
attribute14	Varchar2(240)	No	FND_API.G_MISS_CHAR
attribute15	Varchar2(240)	No	FND_API.G_MISS_CHAR
Amount	Number	Yes	FND_API.G_MISS_NUM
Basis	Varchar2(30)	Yes	FND_API.G_MISS_CHAR
Context	Varchar2(30)	No	FND_API.G_MISS_CHAR
created_by	Number	Yes	FND_API.G_MISS_CHAR
creation_date	Date	Yes	FND_API.G_MISS_DATE
last_updated_by	Number	Yes	FND_API.G_MISS_NUM
last_update_date	Date	Yes	FND_API.G_MISS_DATE
last_update_login	Number	No	FND_API.G_MISS_NUM
limit_exceed_action_code	Varchar2(30)	Yes	FND_API.G_MISS_CHAR
limit_id	Number	Yes	FND_API.G_MISS_NUM
limit_level_code	Varchar2(30)	Yes	FND_API.G_MISS_CHAR
limit_number	Number	Yes	FND_API.G_MISS_NUM

Parameter	Type	Req	Default
list_header_id	Number	Yes	FND_API.G_MISS_NUM
list_line_id	Number	No	FND_API.G_MISS_NUM
limit_hold_flag	Varchar2(1)	Yes	FND_API.G_MISS_CHAR
multival_attr1_type	Varchar2(30)	No	FND_API.G_MISS_CHAR
multival_attr1_context	Varchar2(30)	No	FND_API.G_MISS_CHAR
multival_attribute1	Varchar2(30)	No	FND_API.G_MISS_CHAR
multival_attr1_datatype	Varchar2(10)	No	FND_API.G_MISS_CHAR
multival_attr2_type	Varchar2(30)	No	FND_API.G_MISS_CHAR
multival_attr2_context	Varchar2(30)	No	FND_API.G_MISS_CHAR
multival_attribute2	Varchar2(30)	No	FND_API.G_MISS_CHAR
multival_attr2_datatype	Varchar2(10)	No	FND_API.G_MISS_CHAR
organization_flag	Varchar2(1)	Yes	FND_API.G_MISS_CHAR
program_application_id	Number	No	FND_API.G_MISS_NUM
program_id	Number	No	FND_API.G_MISS_NUM

Parameter	Type	Req	Default
program_update_date	Date	No	FND_API.G_MISS_DATE
request_id	Number	No	FND_API.G_MISS_NUM
return_status	Varchar2(1)	No	FND_API.G_MISS_CHAR
db_flag	Varchar2(1)	No	FND_API.G_MISS_CHAR
operation	Varchar2(30)	No	FND_API.G_MISS_CHAR

Limits_Tbl_Type

The following table shows the parameters for this structure:

Limits_Tbl_Type

Parameter	Type	Req	Drv	Default
Limits_Rec_Type	Record			

Limits_Val_Rec_Type

The following table shows the parameters for this structure:

Limits_Val_Rec_Type

Parameter	Type	Req	Drv	Default
limit_exceed_action	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
limit	Varchar2(240)	No	No	FND_API.G_MISS_CHAR

Parameter	Type	Req	Drv	Default
limit_level	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
list_header	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
list_line	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
organization	Varchar2(240)	No	No	FND_API.G_MISS_CHAR

Limits_Val_Tbl_Type

The following table shows the parameters for this structure:

Limits_Val_Tbl_Type

Parameter	Type	Req	Drv	Default
Limits_Val_Rec_Type	Record			

Limit_Attrs_Rec_Type

For column descriptions, please refer to the Oracle Pricing TRM for the table QP_LIMIT_ATTRIBUTES.

Limit_Attrs_Rec_Type

Parameter	Type	Req	Drv	Default
attribute1	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute2	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute3	Varchar2(240)	No	No	FND_API.G_MISS_CHAR

Parameter	Type	Req	Drv	Default
attribute4	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute5	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute6	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute7	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute8	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute9	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute10	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute11	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute12	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute13	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute14	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute15	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
comparison_operator_code	Varchar2(30)	Yes	No	FND_API.G_MISS_CHAR
context	Varchar2(30)	No	No	FND_API.G_MISS_CHAR

Parameter	Type	Req	Drv	Default
created_by	Number	Yes	No	FND_API.G_MISS_CHAR
creation_date	Date	Yes	No	FND_API.G_MISS_DATE
last_updated_by	Number	Yes	No	FND_API.G_MISS_NUM
last_update_date	Date	Yes	No	FND_API.G_MISS_DATE
last_update_logi n	Number	No	No	FND_API.G_MISS_NUM
limit_attribute	Varchar2(30)	Yes	No	FND_API.G_MISS_CHAR
limit_attribute_c ontext	Varchar2(30)	Yes	No	FND_API.G_MISS_CHAR
limit_attribute_i d	Number	Yes	No	FND_API.G_MISS_NUM
limit_attribute_t ype	Varchar2(30)	Yes	No	FND_API.G_MISS_CHAR
limit_attr_dataty pe	Varchar2(10)	Yes	No	FND_API.G_MISS_CHAR
limit_attr_value	Varchar2(240)	Yes	No	FND_API.G_MISS_CHAR
limit_id	Number	No	No	FND_API.G_MISS_NUM
program_applica tion_id	Number	No	No	FND_API.G_MISS_NUM
program_id	Number	No	No	FND_API.G_MISS_NUM

Parameter	Type	Req	Drv	Default
program_update_date	Date	No	No	FND_API.G_MISS_DATE
request_id	Number	No	No	FND_API.G_MISS_NUM
return_status	Varchar2(1)	No	No	FND_API.G_MISS_CHAR
db_flag	Varchar2(1)	No	No	FND_API.G_MISS_CHAR
operation	Varchar2(30)	No	No	FND_API.G_MISS_CHAR

Limit_Attrs_Tbl_Type

The following table shows the parameters for this structure:

Limit_Attrs_Tbl_Type

Parameter	Type	Req	Drv	Default
Limit_Attrs_Rec_Type	Record			

Limit_Attrs_Val_Rec_Type

The following table shows the parameters for this structure:

Limit_Attrs_Val_Rec_Type

Parameter	Type	Req	Default
comparison_operator	Varchar2(240)	No	FND_API.G_MISS_CHAR
limit_attribute	Varchar2(240)	No	FND_API.G_MISS_CHAR

Parameter	Type	Req	Default
limit	Varchar2(240)	No	FND_API.G_MISS_CHAR

Limit_Attrs_Val_Tbl_Type

The following table shows the parameters for this structure:

Limit_Attrs_Val_Tbl_Type

Parameter	Type	Req	Drv	Default
Limit_Attrs_Val_Rec_Type	Record			

Limit_Balances_Rec_Type

The following table shows the parameters for this structure. For column descriptions, see the Oracle Pricing e-TRM for the table QP_LIMIT_BALANCES.

Limit_Balances_Rec_Type

Parameter	Type	Req	Drv	Default
attribute1	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute2	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute3	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute4	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute5	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute6	Varchar2(240)	No	No	FND_API.G_MISS_CHAR

Parameter	Type	Req	Drv	Default
attribute7	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute8	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute9	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute10	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute11	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute12	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute13	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute14	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
attribute15	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
available_amount	Number	Yes	No	FND_API.G_MISS_NUM
consumed_amount	Number	Yes	No	FND_API.G_MISS_NUM
context	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
created_by		Yes	No	FND_API.G_MISS_CHAR
creation_date	Date	Yes	No	FND_API.G_MISS_DATE

Parameter	Type	Req	Drv	Default
last_updated_by	Number	Yes	No	FND_API.G_MISS_NUM
last_update_date	Date	Yes	No	FND_API.G_MISS_DATE
last_update_logi n	Number	No	No	FND_API.G_MISS_NUM
limit_balance_id	Number	Yes	No	FND_API.G_MISS_NUM
limit_id	Number	Yes	No	FND_API.G_MISS_NUM
organization_attr _context	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
organization_attr ibute	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
organization_attr _value	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
multival_attr1_t ype	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
multival_attr1_c ontext	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
multival_attribut e1	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
multival_attr1_v alue	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
multival_attr1_d atatype	Varchar2(10)	No	No	FND_API.G_MISS_CHAR
multival_attr2_t ype	Varchar2(30)	No	No	FND_API.G_MISS_CHAR

Parameter	Type	Req	Drv	Default
multival_attr2_context	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
multival_attribute2	Varchar2(30)	No	No	FND_API.G_MISS_CHAR
multival_attr2_value	Varchar2(240)	No	No	FND_API.G_MISS_CHAR
multival_attr2_data_type	Varchar2(10)	No	No	FND_API.G_MISS_CHAR
program_application_id	Number	No	No	FND_API.G_MISS_NUM
program_id	Number	No	No	FND_API.G_MISS_NUM
program_update_date	Date	No	No	FND_API.G_MISS_DATE
request_id	Number	No	No	FND_API.G_MISS_NUM
reserved_amount	Number	Yes	No	FND_API.G_MISS_NUM
return_status	Varchar2(1)	No	No	FND_API.G_MISS_CHAR
db_flag	Varchar2(1)	No	No	FND_API.G_MISS_CHAR
operation	Varchar2(30)	No	No	FND_API.G_MISS_CHAR

Limit_Balances_Tbl_Type

The following table shows the parameters for this structure:

Limit_Balances_Tbl_Type

Parameter	Type	Req	Drv	Default
Limit_Balances_ Rec_Type	Record			

Limit_Balances_Val_Rec_Type

The following table shows the parameters for this structure:

Limit_Balances_Val_Rec_Type

Parameter	Type	Req	Drv	Default
limit_balance	Varchar2(240)			FND_API.G_MI SS_CHAR
limit	Varchar2(240)			FND_API.G_MI SS_CHAR

Limit_Balances_Val_Tbl_Type

The following table shows the parameters for this structure:

Limit_Balances_Val_Tbl_Type

Parameter	Type	Req	Drv	Default
Limit_Balances_ Val_Rec_Type	Record			

Validation of Limits Public API

Validations:

Standard Validation

Oracle Advanced Pricing validates all required columns in the Limits Public API.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Limits

Public API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	Null
Failure	4	actual error message

Example of Limits Public API

The following example script is located in the directory \$qp/patch/115/sql:

- insert_limits.sql: Example of inserting limits.
- update_limit_avlbl_amt.sql: Example of how to update available amount on a limit balance.
- delete_limits.sql: Example of deleting a limits record.

Create Publish Price Book Application Program Interface

This section explains how to use the Create_Publish_Price_Book API and how it functions in Oracle Advanced Pricing.

Functional Overview

The Create_Publish_Price_Book PL/SQL API can be used by applications integrating with Oracle Advanced Pricing to initiate a Concurrent Request to create and publish a new price book or to publish an existing price book. The concurrent program name is QP: Price Book Generate and Publish. The request id of the concurrent request submitted is available in the output parameter x_request_id of this API. You can use the request id to monitor the concurrent request status either from the price book user interface (UI), the View Concurrent Requests UI, or programmatically using the Oracle Application Object Library (FND) API FND_CONCURRENT.GET_REQUEST_STATUS (For more information on this FND API, see: Oracle Applications Coding Standards Guide).

The package QP_PRICE_BOOK_PUB contains the Procedure Create_Publish_Price_Book. The package specification QP_PRICE_BOOK_PUB contains the following public record type and table of records type definitions used in the Create_Publish_Price_Book API:

- pb_input_header_rec: record type corresponding to the columns in view QP_PB_INPUT_HEADERS_VL
- pb_input_lines_rec: record type corresponding to the columns in table QP_PB_INPUT_LINES
- pb_input_lines_tbl: table of records of type pb_input_lines_rec

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public API Create_Publish_Price_Book. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

Procedure CREATE_PUBLISH_PRICE_BOOK

The following table lists the parameters for this Procedure

Parameter	Usage	Type	Req	Drv
p_pb_input_header_rec	In	Pb_input_header_rec	No	No
p_pb_input_lines_tbl	In	Pb_input_lines_tbl	No	No
x_request_id	Out	Number	No	No
x_return_status	Out	Varchar2	No	No
x_retcode	Out	Number	No	No
x_err_buf	Out	Varchar2	No	No
x_price_book_messages_tbl	Out	Price_book_messages_tbl	No	No

Type PB_INPUT_HEADER_REC

The following table lists the parameters in this structure:

Parameter	Usage	Type	Req	Drv
customer_context	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
customer_attribute	Null	Varchar2	No	No
customer_attr_value	Null	Varchar2	Yes	No
cust_account_id	Null	Number	No	No
currency_code	Null	Varchar2	Yes	No
limit_products_by	Null	Varchar2	No	No
product_context	Null	Varchar2	No	No
product_attribute	Null	Varchar2	No	No
product_attr_value	Null	Varchar2	No	No
effective_date	Null	Date	Yes	No
item_quantity	Null	Number	Yes	No
pub_template_code	Null	Varchar2	No	No
pub_language	Null	Varchar2	No	No
pub_territory	Null	Varchar2	No	No
pub_output_document_type	Null	Varchar2	No	No
dlv_xml_flag	Null	Varchar2	No	No
dlv_xml_site_id	Null	Number	No	No
dlv_email_flag	Null	Varchar2	No	No
dlv_email_addresses	Null	Varchar2	No	No
dlv_printer_flag	Null	Varchar2	No	No
dlv_printer_name	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
generation_time_code	Null	Varchar2	No	No
gen_schedule_date	Null	Date	No	No
org_id	Null	Number	Yes	No
price_book_type_code	Null	Varchar2	Yes	No
price_based_on	Null	Varchar2	No	No
pl_agr_bsa_id	Null	Number	No	No
pricing_perspective_code	Null	Varchar2	No	No
publish_existing_pb_flag	Null	Varchar2	No	No
overwrite_existing_pb_flag	Null	Varchar2	No	No
request_origination_code	Null	Varchar2	No	No
request_type_code	Null	Varchar2	No	Yes
price_book_name	Null	Varchar2	Yes	No
pl_agr_bsa_name	Null	Varchar2	No	No
pub_template_name	Null	Varchar2	No	No

Type PB_INPUT_LINES_REC

The following table lists the parameters in this structure

Parameter	Usage	Type	Req	Drv
context	Null	Varchar2	No	No
attribute	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute_value	Null	Varchar 2	No	No
attribute_type	Null	Varchar 2	No	No

Type PB_INPUT_LINES_TBL

The following table lists the record type of which this structure is a plsql table:

Parameter	Usage	Type	Req	Drv
pb_input_lines_rec	Null	Record	No	No

Validation of Create Publish Price Book API

Standard Validation

Oracle Advanced Pricing validates all required columns in the Create_Publish_Price_Book API:

- For the input parameter p_pb_input_header_rec the columns price_book_name, price_book_type_code, customer_attr_value, currency_code and org_id are required.
- Price_book_type_code if Null, will be defaulted to 'F' for Full price book.
- Customer_Attr_Value refers to the Party_Id of the customer. Internal users of Advanced Pricing must specify a value since the system cannot determine a default value for this column. For external users, Customer_Attr_Value if Null, will be defaulted with the customer associated with the user who is logged in to the application.

Note: Internal user refers to the customer who has implemented Oracle Advanced Pricing while external user refers to the customer's customer.

Oracle Advanced Pricing validates all conditionally required fields in the Create_Publish_Price_Book API:

- If publish_existing_pb_flag = N (that is, new price book will be generated), then the following fields are required:

1. Currency_Code if null will be defaulted as follows – If the multi-currency functionality is not installed or enabled for the Pricing Perspective application, and if p_pb_input_header_rec.limit_products_by is PRICE_LIST, then currency_code is defaulted from the corresponding price list id specified in p_pb_input_header_rec.pl_agr_bsa_id. However, in cases where a default value is not determinable for the currency_code, the user must specify a value for this column.
 2. Org_id if null will be defaulted based on Multi-Org Access Control rules. However, if the system cannot determine a default value, the user must specify a value for this column.
 3. Item_Quantity and Effective_Date, if null, are defaulted to 1 and current system date respectively but are required for price book generation.
 4. Pricing_Perspective_Code: If null, the value defaults from the system profile option QP: Internal Default Pricing Perspective for internal customers; for external customers, the value defaults from the profile option QP: External Default Pricing Perspective. However, it is required to generate a price book.
 5. Request_Type_Code defaults from the profile QP: Pricing Perspective Request Type. It is required to generate a price book.
 6. Limit_products_by is required. If limit_products_by is ITEM, ITEM_CATEGORY or ALL_ITEMS then product_attr_value is required. If limit_products_by = PRICE_LIST then pl_agr_bsa_id field is required.
 7. If price_based_on is not null then pl_agr_bsa_id is required.
- If either dlv_printer_flag or dlv_email_flag = 'Y', then pub_template_code, pub_language and pub_territory are required fields.
 - If pub_template_code is populated and either a) dlv_email_flag = Y or b) both dlv_email_flag and dlv_printer_flag are N, then pub_output_document_type is required.
 - If dlv_email_flag = Y, then a delivery e-mail address is required. One or more e-mail address may be specified separated by commas.
 - If dlv_xml_flag = Y then dlv_xml_site_id is required.
 - If generation_time_code = SCHEDULE then gen_schedule_date is required.

For more information, see: Oracle eTRM (eTechnical Reference Manual) available on [OracleMetaLink](#).

Other Validation

Oracle Advanced Pricing validates each field against a valid set of values in the

Create_Publish_Price_Book API. The valid set of values is the same as the set of values one that can be selected from the price book UI. Additionally, all flags – publish_existing_pb_flag, overwrite_existing_price_book_flag, dlv_email_flag, dlv_printer_flag, dlv_xml_flag – can be Y or N (null defaults to N) only. A price_book_name, customer_attr_value and price_book_type_code combination is unique. You can overwrite an existing combination by specifying overwrite_existing_pb_flag = Y. A validation error is raised if this flag is N and the price book already exists.

Publish_existing_pb_flag indicates that an existing price book is to be published. A validation error occurs if the price book to be published does not already exist. For Basic Pricing users, generating and publishing a delta price book and publishing an XML Message is not supported. A validation error is raised if price_book_type_code is D or dlv_xml_flag = Y.

An external user can only generate or publish a price book with customer_attr_value that matches the Customer_Id of the user or matches a parent organization of the Customer_id of the user. If the Customer party associated with the user is a Person party who belongs to an Organization party (a record exists in the Trading Community Architecture (TCA) table hz_relationships with a party_type of PARTY_RELATIONSHIP for the Customer party of the user and the Object (object_id) of the relationship is an Organization party), then the Person can create a price book for himself or for the Organization to which he belongs, but not for any other customer.

An external user cannot create or publish a price book using the Purchasing pricing perspective. An internal user can create or publish a price book with pricing perspective Purchasing, but cannot use the "Send XML message" publishing option for such price books. While creating or publishing a price book from the Purchasing pricing perspective, the value -1 must be passed for the input parameter p_pb_input_header_rec.customer_attr_value and null must be passed for the input parameter p_pb_input_header_rec.cust_account_id. While creating a price book with Purchasing pricing perspective, the input parameter p_pb_input_header_rec.price_based_on can be PRICE_LIST (or null) but not AGREEMENT or BSA.

Error Handling

The output parameter x_return_status can have the values - fnd_api.g_ret_sts_success (S) or fnd_api.g_ret_sts_error (E) – indicating Success or Error respectively. The parameter x_return_status is E when either validation fails or if the concurrent request submission failed (such as x_request_id = 0). The output parameter x_request_id, if 0, indicates that the concurrent request submission failed to generate a Request Id. Similarly, output parameter x_retcode can have values 0 (success), 1 (warning) or 2 (error) indicating the status of concurrent request submission and x_err_buf contains any corresponding message. The output parameter x_price_book_messages_tbl contains any validation error messages associated with the price book create or publish request.

Examples of Create_Publish_Price_Book API

The following example scripts are located in the directory \$qp/patch/115/sql:

- QPPBXMP1.sql: Create a price book.
- QPPBXMP2.sql: Create and Publish (print, e-mail, and view document) a price book.

Get Currency Application Program Interface

The Get_Currency API retrieves all the currency codes for a given price list. The package QP_UTIL_PUB contains the procedure Get_Currency. This section explains how to use the Get_Currency API and how it functions in Oracle Advanced Pricing.

Functional Overview

While processing an order using a particular price list, Order Management (OM) displays a valid set of currency codes when the list of values (LOV) is selected. OM calls this API to get the valid currency codes for the given price list and pricing effective date. If pricing effective date is not passed by calling application, the current date defaults. If the profile options QP: Multi Currency Installed and QP: Multi Currency Usage are set to Y (Yes), then currency codes are retrieved by joining the view fnd_currencies_vl, table qp_list_headers_b and table qp_currency_details for the passed price list id; otherwise, all the effective currencies are retrieved from view fnd_currencies_vl.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the Get_Currency API. All of the inbound and outbound parameters are listed.

Get_Currency

Parameter	Usage	Type	Req	Drv
l_price_list_id	IN	Number	Yes	No
l_pricing_effective_date	IN	Date	No	No
l_currency_code_tbl	OUT	currency_code_tbl	No	No

CURRENCY_REC

The following table shows the parameters for this structure.

CURRENCY_REC

Parameter	Usage	Type	Req	Drv
currency_code	Null	Varchar2	Yes	No
currency_name	Null	Varchar2	Yes	No
currency_precision	Null	Number	Yes	No

CURRENCY_CODE_TBL

The following table shows the parameters for this structure.

CURRENCY_CODE_TBL

Parameter	Usage	Type	Req	Drv
Currency_rec	Null	Record	No	No

Validation of Get_Currency API

Validations:

Standard Validation

The caller is responsible for passing the right parameters to Get_Currency.

For specific information on the data implied by these columns, see your *Oracle Pricing Technical Reference Manual* for details.

Other Validation

None.

Error Handling

If any exception occurs in the Get_Currency API, it does not return any Currency Codes.

Note: The Package Specification and Body files are QPXRTCNS.pls and QPXRTCNB.pls and are available under the source control directory \$QP_TOP/patch/115/sql.

Get Custom Price Application Program Interface

This section explains how to use the Get_Custom Price API (used in Formulas Setup) and how it functions in Oracle Advanced Pricing.

The pricing engine calls the Get_Custom_Price API when evaluating a formula that contains a formula line (step) of type Function. One or more formulas may be set up to contain a formula line of type Function and the same Get_Custom_price API is called each time. Therefore, the user must code the logic in the API based on the price_formula_id that is passed as an input parameter to the API.

Note: The Get Custom Price API can be customized by adding custom code.

Functional Overview

The package specification QP_CUSTOM contains the specification for the Get_Custom_Price API. The package body/function body is not shipped with Oracle Advanced Pricing. The user must create the Package Body for QP_CUSTOM containing the function body for Get_Custom_Price which must adhere to the Function specification (in terms of parameters and return value) provided in the QP_CUSTOM package specification. For the engine to make a call to Get_Custom_Price API, set the profile QP: Get Custom Price Customized at the site level to Yes. The Pricing engine displays an error if this profile is set and QP_CUSTOM. Get_Custom_Price is not created in the database in the applications schema:

- API to all procedures within and outside of this package
- Get_Custom_Price: A customizable function.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Get Custom Price API. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow. These parameters are input parameters to the Get_Custom_Price API and are passed by the Engine.

The following table shows the parameters for this structure.

Get_Custom_Price

Parameter	Usage	Type	Req	Drv
p_price_formula_id	In	Number	Yes	No
p_list_price	In	Number	No	No
p_price_effective_date	In	Date	Yes	No
p_req_line_attrs_tbl	In	Table (REQ_LINE_ATTRS_TBL)	Yes	No
	Return	Number	Yes	No

REQ_LINE_ATTRS_TBL

The following table shows the parameters for this structure.

REQ_LINE_ATTRS_TBL

Parameter	Usage	Type	Req	Drv
rec_line_attrs_rec	Null	Record (REQ_LINE_ATTRS_REC)	No	No

REQ_LINE_ATTRS_REC

The following table shows the parameters for this structure.

REQ_LINE_ATTRS_REC

Parameter	Usage	Type	Req	Drv
line_index	Null	Number	Yes	No
attribute_type	Null	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
context	Null	Varchar2	Yes	No
attribute	Null	Varchar2	Yes	No
value	Null	Varchar2	Yes	No

Validation of Get Custom Price API

The following describes the validations for the Get Custom Price API:

Standard Validation

The user is responsible for all code and validations in the Get Custom Price API. For more information, see: Oracle Pricing Technical Reference Manual.

Other Validation

None

Error Handling

The user is responsible for all error handling in the Get Custom Price API. If any user-coded validation fails, the API should raise exceptions to propagate it to the calling module.

Sample Code1: The following is a sample code showing how the body of the Get_Custom_Price function is coded in the file \$QP_TOP/patch/115/sql/.

The user must use the function specification of Get_Custom_Price as well as any type definitions from \$QP_TOP/patch/115/sql/.

The parameters to Get_Custom_Price are always fixed and not customizable. But the user can use the input parameters passed by the pricing engine in their custom code. The function returns a number. The user can code the function to return the desired value which must be a number. The return value is used in the evaluation of the formula.

For example, consider a formula having an expression 1*2 where 1 and 2 are step numbers. Each step number corresponds to a formula line. Each formula line has a type:

- Step 1 corresponds to a formula line of type Numeric Constant with a component of '200,' and
- Step 2 corresponds to a formula line of type Function. This means the value returned by the QP_CUSTOM.Get_Custom_Price function will be used as the value for this step.

To evaluate the Formula, the pricing engine first obtains the value of each step and substitutes the step with its value in the expression. So step 1 is substituted by the value

which is 200. Step 2 is substituted with the value returned by `Get_Custom_Price` which must be customized by the user (the Profile Option mentioned earlier must also be set to Yes to use this `Get_Custom_Price` functionality).

If `Get_Custom_Price` is customized as below:

```

PACKAGE BODY QP_CUSTOM AS
/*****
***
The Get_Custom_Price Function name and parameters are not customizable
but the
body can be been customized. The parameters are:
p_price_formula_id: Primary key of the formula that uses the
Get_Custom_Price
function
p_list_price: List price of the price list line to which the formula
using Get_
Custom_Price is attached. May have null value.
p_price_effective_date: Current date when Formula is being evaluated by
the
pricing engine.
p_req_line_attrs_tbl: PL/SQL table of records containing Context,
Attribute,
Attribute Value records for Product and Pricing Attributes and a column
indicating the type - whether Product Attribute or Pricing Attribute.
Also the engine passes the Pricing Attributes and Product Attributes of
only the
current line to which the formula is attached.
The parameters are passed to the function by the pricing engine and can
be used
in the function body.
*****/
**/
FUNCTION Get_Custom_Price
    p_price_formula_id IN NUMBER,
    p_list_price IN NUMBER,
    p_price_effective_date IN DATE,
    p_req_line_attrs_tbl IN
QP_FORMULA_PRICE_CALC_PVT.REQ_LINE_ATTRS_TBL)

RETURN NUMBER IS
v_requested_item VARCHAR2(240);
v_weight NUMBER;

BEGIN
IF p_price_formula_id = 1726 -- Assume this is the internal Id/primary
key for
the sample Formula 1*2
THEN
    --Loop through the PL/SQL table of records passed by the Engine as
an input parameter and
    --containing Pricing Attributes and Product Attributes of the Price
List
    Line or Modifier Line to which
    -- the current formula is attached.
FOR i IN 1..p_req_line_attrs_tbl.count LOOP

IF p_req_line_attrs_tbl(i).attribute_type = 'PRODUCT'

AND /*Attribute Type is Product*/
    p_req_line_attrs_tbl(i).context = 'ITEM'

AND
    p_req_line_attrs_tbl(i).attribute = 'PRICING_ATTRIBUTE1'
THEN
    -- For this combination of Product Context and Attribute, the
Attribute Value is the Inventory Item Id

```

```

v_requested_item := p_req_line_attrs_tbl(i).value;
END IF;

        IF p_req_line_attrs_tbl(i).attribute_type = 'PRICING'
AND      /*Attribute Type is Pricing*/
        p_req_line_attrs_tbl(i).context = 'MIXED'
AND
        p_req_line_attrs_tbl(i).attribute = 'PRICING_ATTRIBUTE4'
THEN
        --For this combination of Pricing Context and Attribute, let's
say, the
        Attribute Value is the Weight of
        --the item to which the formula is attached.
        v_weight := p_req_line_attrs_tbl(i).value;
        END IF;
END LOOP; /*For Loop*/

        RETURN v_weight;

EXCEPTION

        WHEN OTHERS THEN

RETURN NULL;
END Get_Custom_Price;
END QP_CUSTOM;
Then if v_weight has a value '1.2' then Get_Custom_Price returns a value
of '1.2'
Therefore, the pricing engine evaluates the formula as 200*1.2 = 240.

```

Get Price Book Application Program Interface

This section explains how to use the Get_Price_Book API and how it functions in Oracle Advanced Pricing.

Functional Overview

The Get_Price_Book PL/SQL API can be used by applications integrating with Oracle Advanced Pricing to query an existing price book. This API maybe used to query a price book that has been created from the price book user interface (UI), the Create_Publish_Price_Book public API, or from a Get_Catalog Inbound XML Message. Before querying a price book, ensure that the concurrent request generating the price book has completed successfully.

The package QP_PRICE_BOOK_PUB contains the Procedure Get_Price_Book and a version of Get_Price_Book with an additional output parameter x_documents_rec. The package specification QP_PRICE_BOOK_PUB contains the following public record type and table of records type definitions used in the Get_Price_Book API:

- price_book_header_rec: record type corresponding to the view– QP_PRICE_BOOK_HEADERS_V
- price_book_lines_rec: record type corresponding to the view

QP_PRICE_BOOK_LINES_V

- price_book_lines_tbl: table of records of type price_book_lines_rec
- price_book_line_details_rec: record type corresponding to the view QP_PRICE_BOOK_LINE_DETAILS_V
- price_book_line_details_tbl: table of records of type price_book_line_details_rec · price_book_attributes_rec - record type corresponding to the view QP_PRICE_BOOK_ATTRIBUTES_V
- price_book_attributes_tbl: table of records of type price_book_attributes_rec
- price_book_break_lines_rec: record type corresponding to the view QP_PRICE_BOOK_BREAK_LINES_V
- price_book_break_lines_tbl: table of records of type price_book_break_lines_rec
- price_book_messages_tbl: table of records where record type corresponds to the ROWTYPE of table QP_PRICE_BOOK_MESSAGES
- documents_rec: record type corresponding to the table QP_DOCUMENTS (used in the overloaded version)
- VARCHAR_TBL: table of VARCHAR2(240)

Setting Up and Parameter Descriptions

The following table describes all parameters used by the public API Get_Price_Book. All of the inbound and outbound parameters are listed.

Procedure GET_PRICE_BOOK

The following table lists the parameters for this procedure:

Parameter	Usage	Type	Req	Drv
p_price_book_name	In	Varchar2	Yes	No
p_customer_id	In	Varchar2	Yes	No
p_price_book_type_code	In	Varchar2	Yes	No
x_price_book_header_rec	Out	price_book_header_rec	No	No

Parameter	Usage	Type	Req	Drv
x_price_book_lines_tbl	Out	price_book_lines_tbl	No	No
x_price_book_line_details_tbl	Out	price_book_line_details_tbl	No	No
x_price_book_attributes_tbl	Out	price_book_attributes_tbl	No	No
x_price_book_break_lines_tbl	Out	price_book_break_lines_tbl	No	No
x_price_book_messages_tbl	Out	price_book_messages_tbl	No	No
x_return_status	Out	Varchar2	No	No
x_query_messages	Out	VARCHAR_TBL	No	No

Procedure GET_PRICE_BOOK (overloaded version)

The following table lists the parameters for this procedure:

Note: The overloaded version of Get_Price_Book procedure has an additional output parameter x_documents_rec. This overloaded version of Get_Price_Book API may be called instead, to additionally get the formatted price book document (stored in the database as a .pdf, .rtf, etc. file) and related information (mime type, file name, etc).

Parameter	Usage	Type	Req	Drv
p_price_book_name	In	Varchar2	Yes	No
p_customer_id	In	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
p_price_book_type_code	In	Varchar2	Yes	No
x_price_book_header_rec	Out	price_book_header_rec	No	No
x_price_book_lines_tbl	Out	price_book_lines_tbl	No	No
x_price_book_line_details_tbl	Out	price_book_line_details_tbl	No	No
x_price_book_attributes_tbl	Out	price_book_attributes_tbl	No	No
x_price_book_break_lines_tbl	Out	price_book_break_lines_tbl	No	No
x_price_book_messages_tbl	Out	price_book_messages_tbl	No	No
x_documents_rec	Out	documents_rec	No	No
x_return_status	Out	Varchar2	No	No
x_query_messages	Out	VARCHAR_TBL	No	No

Type PRICE_BOOK_HEADER_REC

The following table lists the parameters in this structure:

Parameter	Usage	Type	Req	Drv
price_book_header_id	Null	Number	No	No
price_book_type_code	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
currency_code	Null	Varchar2	No	No
effective_date	Null	Date	No	No
org_id	Null	Number	No	No
customer_id	Null	Number	No	No
cust_account_id	Null	Number	No	No
document_id	Null	Number	No	No
item_category	Null	Number	No	No
price_based_on	Null	Varchar2	No	No
pl_agr_bsa_id	Null	Number	No	No
pricing_perspective_code	Null	Varchar2	No	No
item_quantity	Null	Number	No	No
request_id	Null	Number	No	No
request_type_code	Null	Varchar2	No	No
pb_input_header_id	Null	Number	No	No
pub_status_code	Null	Varchar2	No	No
price_book_name	Null	Varchar2	No	No
pl_agr_bsa_name	Null	Varchar2	No	No
creation_date	Null	Date	No	No
created_by	Null	Number	No	No
last_update_date	Null	Date	No	No

Parameter	Usage	Type	Req	Drv
last_updated_by	Null	Number	No	No
last_update_login	Null	Number	No	No
price_book_type	Null	Varchar2	No	No
currency	Null	Varchar2	No	No
operating_unit	Null	Varchar2	No	No
customer_name	Null	Varchar2	No	No

Type PRICE_BOOK_LINES_REC

The following table lists the parameters in this structure

Parameter	Usage	Type	Req	Drv
price_book_line_id	Null	Number	No	No
price_book_header_id	Null	Number	No	No
item_number	Null	Number	No	No
product_uom_code	Null	Varchar2	No	No
list_price	Null	Number	No	No
net_price	Null	Number	No	No
sync_action_code	Null	Varchar2	No	No
line_status_code	Null	Varchar2	No	No
creation_date	Null	Date	No	No
created_by	Null	Number	No	No
last_update_date	Null	Date	No	No

Parameter	Usage	Type	Req	Drv
last_updated_by	Null	Number	No	No
last_update_login	Null	Number	No	No
description	Null	Varchar2	No	No
customer_item_number	Null	Varchar2	No	No
customer_item_desc	Null	Varchar2	No	No
display_item_number	Null	Varchar2	No	No
sync_action	Null	Varchar2	No	No

Type PRICE_BOOK_LINES_TBL

The following table lists the record type of which this structure is a plsql table:

Parameter	Usage	Type	Req	Drv
price_book_lines_rec	Null	Record	No	No

Type PRICE_BOOK_LINE_DETAILS_REC

The following table lists the parameters in this structure:

Parameter	Usage	Type	Req	Drv
price_book_line_det_id	Null	Number	No	No
price_book_line_id	Null	Number	No	No
price_book_header_id	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
list_line_no	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
list_price	Null	Number	No	No
modifier_operand	Null	Number	No	No
modifier_application_method	Null	Varchar2	No	No
adjustment_amount	Null	Number	No	No
adjusted_net_price	Null	Number	No	No
list_line_type_code	Null	Varchar2	No	No
price_break_type_code	Null	Varchar2	No	No
creation_date	Null	Date	No	No
created_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_updated_by	Null	Number	No	No
last_update_login	Null	Number	No	No
list_name	Null	Varchar2	No	No
list_line_type	Null	Varchar2	No	No
price_break_type	Null	Varchar2	No	No
application_method_name	Null	Varchar2	No	No

Type PRICE_BOOK_LINE_DETAILS_TBL

The following table lists the record type of which this structure is a plsql table

Parameter	Usage	Type	Req	Drv
price_book_line_details_rec	Null	Record	No	No

Type PRICE_BOOK_ATTRIBUTES_REC

The following table lists the parameters in this structure

Parameter	Usage	Type	Req	Drv
price_book_attribute_id	Null	Number	No	No
price_book_line_det_id	Null	Number	No	No
price_book_line_id	Null	Number	No	No
price_book_header_id	Null	Number	No	No
pricing_prod_context	Null	Varchar2	No	No
pricing_prod_attribute	Null	Varchar2	No	No
comparison_operator_code	Null	Varchar2	No	No
pricing_prod_attr_value_from	Null	Varchar2	No	No
pricing_attr_value_to	Null	Varchar2	No	No
pricing_prod_attr_datatype	Null	Varchar2	No	No
attribute_type	Null	Varchar2	No	No
creation_date	Null	Date	No	No
created_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_updated_by	Null	Number	No	No
last_update_login	Null	Number	No	No
context_name	Null	Varchar2	No	No
attribute_name	Null	Varchar2	No	No
attribute_value_name	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute_value_to_name	Null	Varchar2	No	No
comparison_operator_name	Null	Varchar2	No	No

Type PRICE_BOOK_ATTRIBUTES_TBL

The following table lists the record type of which this structure is a plsql table:

Parameter	Usage	Type	Req	Drv
price_book_attributes_rec	Null	Record	No	No

Type PRICE_BOOK_BREAK_LINES_REC

The following table lists the parameters in this structure

Parameter	Usage	Type	Req	Drv
price_book_break_line_id	Null	Number	No	No
price_book_header_id	Null	Number	No	No
price_book_line_id	Null	Number	No	No
price_book_line_det_id	Null	Number	No	No
pricing_context	Null	Varchar2	No	No
pricing_attribute	Null	Varchar2	No	No
comparison_operator_code	Null	Varchar2	No	No
pricing_attr_value_from	Null	Varchar2	No	No
pricing_attr_value_to	Null	Varchar2	No	No
pricing_attribute_datatype	Null	Varchar2	No	No
operand	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
application_method	Null	Varchar2	No	No
creation_date	Null	Date	No	No
created_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_updated_by	Null	Number	No	No
last_update_login	Null	Number	No	No
context_name	Null	Varchar2	No	No
attribute_name	Null	Varchar2	No	No
attribute_value_name	Null	Varchar2	No	No
attribute_value_to_name	Null	Varchar2	No	No
comparison_operator_name	Null	Varchar2	No	No
application_method_name	Null	Varchar2	No	No
recurring_value	Null	Number	No	No

Type PRICE_BOOK_BREAK_LINES_TBL

The following table lists the record type of which this structure is a plsql table:

Parameter	Usage	Type	Req	Drv
price_book_break_lines_rec	Null	Record	No	No

Type PRICE_BOOK_MESSAGES_TBL

The following table lists the record type of which this structure is a plsql table:

Parameter	Usage	Type	Req	Drv
QP_PRICE_BOOK_MESSAGES% ROWTYPE	Null	Record	No	No

Type PRICE_BOOK_MESSAGES_TBL

Parameter	Usage	Type	Req	Drv
VARCHAR2(240)	Null	Record	No	No

Type DOCUMENTS_REC

The following table lists the parameters in this structure:

Parameter	Usage	Type	Req	Drv
document_id	Null	Number	No	No
document_content	Null	Blob	No	No
document_content_type	Null	Varchar2	No	No
document_name	Null	Varchar2	No	No
creation_date	Null	Date	No	No
created_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_updated_by	Null	Number	No	No
last_update_login	Null	Number	No	No

Validation of Get Price Book API

Standard Validation

Oracle Advanced Pricing validates all required input parameters in the Get Price Book API. For more information, see: Oracle eTRM (eTechnical Reference Manual) available on [OracleMetaLink](#).

Other Validation

Oracle Advanced Pricing ensures that an external user can only query a price book whose customer id matches the customer id of the user or the parent organization of the customer id of the user. If the Customer party associated with the user is a Person party who belongs to an Organization party (for example, a record exists in the Trading Community Architecture (TCA) table `hz_relationships` with a `party_type` of `PARTY_RELATIONSHIP` for the Customer party of the user and the Object (`object_id`) of the relationship is an Organization party), then the Person can query a price book created for himself or for the Organization to which he belongs, but not for any other customer.

An external user cannot query a price book with the pricing perspective Purchasing. An internal user can query a price book with the pricing perspective Purchasing. However, these price books have a `customer_id` of -1. So the internal user must pass -1 for the input parameter `p_customer_id` while calling the `Get_Price_Book` API.

Error Handling

If any validation of the input parameters fails, the API returns error status to the calling module from the output parameter `x_return_status`. Possible values are S or E for Success and Error respectively. Validation error messages are returned in the output parameter `x_query_messages`. Note that the output parameter `x_price_book_messages_tbl` contains error messages associated with the generation of the price book.

Examples of Get_Price_Book API

The following example scripts are located in the directory `$qp/patch/115/sql`:

- `QPPBXMP3.sql`: Query a price book using the `Get_Price_Book` API.
- `QPPBXMP4.sql`: Query a price book using the overloaded version of the `Get_Price_Book` API. This version has an additional output parameter `x_documents_rec` which is the record containing columns relating to the view (formatted) document.

Get Price List Application Program Interface

This section explains how to use the `Get_Price_List` API and how it functions in Oracle Advanced Pricing.

The `Get Price List` API retrieves all the price lists for a given currency code and/or agreement id. The package `QP_UTIL_PUB` contains the procedure `Get_Price_List`.

Functional Overview

While processing an order using a particular currency code and/or agreement id, the OM displays a valid set of price lists when the LOV is activated. The OM calls this API to get the valid price lists for the given currency code, and/or agreement id and pricing

effective date. If the pricing effective date is not passed by calling application, the current date defaults. If the profile options QP: Multi Currency Installed and QP: Multi Currency Usage are set to Y (Yes), then the price lists are retrieved by joining the view qp_list_headers_vl and table qp_currency_details; otherwise, only the view qp_list_headers_vl is used.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the Get_Price_List API. All of the inbound and outbound parameters are listed.

Get_Price_List

Parameter	Usage	Type	Req	Drv
l_currency_code	IN	Varchar2	Yes	No
l_pricing_effective_date	IN	Date	No	No
l_agreement_id	IN	Number	No	No
l_price_list_tbl	OUT	price_list_tbl	No	No

PRICE_LIST_REC

The following table shows the parameters for this structure.

PRICE_LIST_REC

Parameter	Usage	Type	Req	Drv
price_list_id	Null	Number	Yes	No
name	Null	Varchar2	Yes	No
Description	Null	Varchar2	Yes	No
start_date_active	Null	Date	No	No
end_date_active	Null	Date	No	No

PRICE_LIST_TBL

The following table shows the parameters for this structure.

PRICE_LIST_TBL

Parameter	Usage	Type	Req	Drv
price_list_rec	Null	Record	No	No

Validation of Get_Price_List API

Validations:

Standard Validation

The caller is responsible for passing the right parameters to Get_Price_List.

For specific information on the data implied by these columns, see: Oracle Pricing Technical Reference Manual for details.

Other Validation

None.

Error Handling

If any exception occurs in the Get_Price_List API, it does not return any price lists.

Note: The Package Specification and Body files are QPXRTCNS.pls and QPXRTCNB.pls and are available under the source control directory \$QP_TOP/patch/115/sql.

Multi-Currency Conversion Setup Application Program Interface

This section explains how to use the Multi-Currency Conversion Setup API and how it functions in Oracle Advanced Pricing. The Multi-Currency Conversion Setup package consists of entities to set up Multi-Currency Conversion.

Functional Overview

The Multi-Currency Conversion Setup package QP_Currency_PUB.Process_Currency contains the following public record type and table of records entities:

- **Process_Currency:** QP_Currency_PUB.Process_Currency: Takes two record types and two table types as input parameters. Use this API to insert, and update Multi-Currency Conversion and to set up a Multi-Currency Conversion for a given p_CURR_LISTS_rec record structure. The Multi-Currency Conversion can not be deleted but it can be inactivated by setting the effective dates.

You can:

- Set up multiple multi-currency conversion lines by giving multiple multi-currency conversion line definitions in the p_CURR_DETAILS_tbl table structure.
- Curr_Lists_Rec_Type: Corresponds to the columns in the multi-currency header tables QP_CURRENCY_LISTS_B and QP_CURRENCY_LISTS_TL.
- Curr_Lists_Val_Rec_Type: Attributes that store the meaning of id or code columns in the multi-currency header table QP_CURRENCY_LISTS_B, for example, Base Currency.
- Curr_Details_Rec_Type: Corresponds to columns in the multi-currency conversion line table QP_CURRENCY_DETAILS.
- Curr_Details_Tbl_Type: Table of Curr_Details_Rec_Type.
- Curr_Details_Val_Rec_Type: Attributes that store the meaning of id or code columns in the multi-currency conversion line table QP_CURRENCY_DETAILS, for example, Markup_Formula.
- Curr_Details_Val_Tbl_Type: Table of Curr_Details_Val_Rec_Type.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Multi-Currency Conversion Setup. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

PROCESS_CURRENCY

The following table shows the parameters for this structure.

PROCESS_CURRENCY

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Number	No	No
x_msg_data	Out	Varchar2	No	No
p_CURR_LISTS_ rec	In	Curr_Lists_Rec_ Type	No	No
p_CURR_LISTS_ val_rec	In	Curr_Lists_Val_ Rec_Type	No	No
p_CURR_DETAI LS_tbl	In	Curr_Details_Tbl _Type	No	No
p_CURR_DETAI LS_val_tbl	In	Curr_Details_Va l_Tbl_Type	No	No
x_CURR_LISTS_ rec	Out	Curr_Lists_Rec_ Type	No	No
x_CURR_LISTS_ val_rec	Out	Curr_Lists_Val_ Rec_Type	No	No
x_CURR_DETAI LS_tbl	Out	Curr_Details_Tbl _Type	No	No
x_CURR_DETAI LS_val_tbl	Out	Curr_Details_Va l_Tbl_Type	No	No

CURR_LISTS_REC_TYPE

The following table shows the parameters for this structure.

CURR_LISTS_REC_TYPE

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
base_currency_code	Null	Varchar2	Yes	No
Context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
currency_header_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
Description	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
Name	Null	Varchar2	No	No
base_rounding_f actor	Null	Number	No	No
base_markup_fo rmula_id	Null	Number	No	No
base_markup_op erator	Null	Varchar2	No	No
base_markup_va lue	Null	Number	No	No
program_applica tion_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update _date	Null	Date	No	No
prorate_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
Operation	Null	Varchar2	Yes	No

CURR_LISTS_TBL_TYPE

The following table shows the parameters for this structure.

CURR_LISTS_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Curr_Lists_Rec_Type	Null	Record	No	No

CURR_LISTS_VAL_REC_TYPE

The following table shows the parameters for this structure.

CURR_LISTS_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
base_currency	Null	Varchar2	No	No
currency_header	Null	Varchar2	No	No
base_markup_formula	Null	Varchar2	No	No

CURR_LISTS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

CURR_LISTS_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Curr_Lists_Val_Rec_Type	Null	Record	No	No

CURR_DETAILS_REC_TYPE

The following table shows the parameters for this structure.

CURR_DETAILS_REC_TYPE

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
Context	Null	Varchar2	No	No
conversion_date	Null	Date	No	No
conversion_date _type	Null	Varchar2	No	No
conversion_type	Null	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
currency_detail_id	Null	Number	No	No
currency_header_id	Null	Number	No	No
end_date_active	Null	Date	No	No
fixed_value	Null	Number	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
markup_formula_id	Null	Number	No	No
markup_operato r	Null	Varchar2	No	No
markup_value	Null	Number	No	No
price_formula_id	Null	Number	No	No
program_applica tion_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update _date	Null	Date	No	No
request_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
rounding_factor	Null	Number	No	No
selling_rounding_factor	Null	Number	No	No
start_date_active	Null	Date	No	No
to_currency_code	Null	Varchar2	Yes	No
curr_attribute_type	Null	Varchar2	No	No
curr_attribute_context	Null	Varchar2	No	No
curr_attribute	Null	Varchar2	No	No
curr_attribute_value	Null	Varchar2	No	No
Precedence	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No

CURR_DETAILS_TBL_TYPE

The following table shows the parameters for this structure.

CURR_DETAILS_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Curr_Details_Record_Type	Null	Record	No	No

CURR_DETAILS_VAL_REC_TYPE

The following table shows the parameters for this structure.

CURR_DETAILS_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
currency_detail	Null	Varchar2	No	No
currency_header	Null	Varchar2	No	No
markup_formula	Null	Varchar2	No	No
price_formula	Null	Varchar2	No	No
to_currency	Null	Varchar2	No	No

CURR_DETAILS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

CURR_DETAILS_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Curr_Details_Val_Rec_Type	Null	Record	No	No

Validation of Multi-Currency Conversion API

Validations:

Standard Validation

Oracle Advanced Pricing validates all required columns in the Multi-Currency Conversion API. For more information, see: Oracle Pricing Technical Reference Manual.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Multi-Currency Conversion API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Example of Multi-Currency Conversion API

The following example script is located in the directory \$qp/patch/115/sql:

QPXEXCUR.sql: This script creates a multi-currency conversion list converting USD to FRF and GBPn.

Attaching a multi-currency conversion to a Price List

The multi-currency conversion can be attached to a Price List by passing the currency header id (of the multi-currency conversion to be attached) to the record QP_Price_List_PUB.Price_List_Rec_Type while using the Application Program Interface (QP_Price_List_PUB.Process_Price_List()) for Price List.

Attaching a multi-currency conversion to 'AGR' agreement

The multi-currency conversion can be attached to a Agreement by passing the currency header id (of the multi-currency conversion to be attached) to the record QP_PRICE_LIST_PUB.Price_List_Rec_Type while using the Application Program Interface (OE_Pricing_Cont_PUB.Process_Agreement ()) for Agreement.

Defaulting a multi-currency conversion to a 'STANDARD' agreement from the price list

The multi-currency conversion of price list gets defaulted while creating an agreement of type STANDARD. No extra steps are needed. Just specify the price list id, as used to be, while creating an agreement of type STANDARD using the Application Program Interface (OE_Pricing_Cont_PUB.Process_Agreement ()) for Agreement.

Price List Setup Application Program Interface

This section explains how to use the Price List Setup API and how it functions in Oracle Advanced Pricing. The Price List Setup package consists of entities to set up price lists.

Functional Overview

The Price List Setup package QP_Price_List_PUB.Process_Price_List contains the following public record type and table of records entities:

- Process_Price_List: QP_Price_List_PUB.Process_Price_List: Takes two record types and six table types as input parameters. Use this API to insert, update, and delete

price lists and to set up a price list for a given p_PRICE_LIST_rec record structure.

You can:

- Set up multiple price list lines by giving multiple price list line definitions in the p_PRICE_LIST_LINE_tbl table structure.
- Attach multiple qualifiers at the price list header level by giving multiple qualifiers in the p_QUALIFIERS_tbl table structure.
- Attach multiple pricing attributes to price list lines by giving the pricing attributes in the p_PRICING_ATTR_tbl table structure.
- Price_List_Rec_Type: Corresponds to the columns in the price list header tables QP_LIST_HEADERS_B and QP_LIST_HEADERS_TL.
- Price_List_Val_Rec_Type: Attributes that store the meaning of id or code columns in the price list header table QP_LIST_HEADERS_B, for example, Currency.
- Price_List_Line_Rec_Type: Corresponds to columns in the price list line table and related modifiers tables QP_LIST_LINES and QP_RLTD_MODIFIERS.
- Price_List_Line_Tbl_Type: Table of Price_List_Line_Rec_Type.
- Price_List_Line_Val_Rec_Type: Attributes that store the meaning of id or code columns in the price list line table QP_LIST_LINES, for example, Price_By_Formula.
- Price_List_Line_Val_Tbl_Type: Table of Price_List_Line_Val_Rec_Type.
- Qualifiers_Rec_Type: Corresponds to the columns in the qualifier table QP_QUALIFIERS.
- Qualifiers_Tbl_Type: Table of Qualifiers_Rec_Type.
- Qualifiers_Val_Rec_Type: Made up of attributes that store the meaning of id or code columns in the qualifiers table QP_QUALIFIERS, for example, Qualifier_Rule.
- Qualifiers_Val_Tbl_Type: Table of Qualifiers_Val_Rec_Type.
- Pricing_Attr_Rec_Type: Corresponds to the columns in the pricing attributes table QP_PRICING_ATTRIBUTES.
- Pricing_Attr_Tbl_Type: Table of Pricing_Attr_Rec_Type.
- Pricing_Attr_Val_Rec_Type: Attributes that store the meaning of id or code columns in the pricing attributes table QP_PRICING_ATTRIBUTES, for example, Accumulate.

- Pricing_Attr_Val_Tbl_Type: Table of Pricing_Attr_Val_Rec_Type.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Price List Setup. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

PROCESS_PRICE_LIST

The following table shows the parameters for this structure.

PROCESS_PRICE_LIST

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Varchar2	No	No
p_PRICE_LIST_rec	In	Price_List_Rec_Type	No	No
p_PRICE_LIST_val_rec	In	Price_List_Val_Rec_Type	No	No
p_PRICE_LIST_tbl	In	Price_List_Line_tbl_type	No	No
p_PRICE_LIST_val_tbl	In	Price_List_Line_Val_tbl_type	No	No

Parameter	Usage	Type	Req	Drv
p_QUALIFIERS_tbl	In	QP_Qualifier _Rules_ PUB.Qualifie rs_tbl_Type	No	No
p_QUALIFIERS_val_tbl	In	QP_Qualifier _Rules_ PUB.Qualifie rs_ val_tbl_Type	No	No
p_PRICING_ATTR_tbl	In	Pricing_Attr_ tbl_type	No	No
p_PRICING_ATTR_val_tbl	In	Pricing_Attr_ val_tbl_type	No	No
x_PRICE_LIST_rec	Out	Price_List_Re c_Type	No	No
x_PRICE_LIST_val_rec	Out	Price_List_V al_Rec_Type	No	No
x_PRICE_LIST_LINE_tbl	Out	Price_List_Li ne_tbl_type	No	No
x_PRICE_LIST_LINE_val_tbl	Out	Price_List_Li ne_Val_tbl_t ype	No	No
x_QUALIFIERS_tbl	Out	QP_Qualifier _Rules_ PUB.Qualifie rs_tbl_Type	No	No
x_QUALIFIERS_val_tbl	Out	QP_Qualifier _Rules_ PUB.Qualifie rs_ val_tbl_Type	No	No
x_PRICING_ATTR_tbl	Out	Pricing_Attr_ tbl_type	No	No

Parameter	Usage	Type	Req	Drv
x_PRICING_ATTR_val_tbl	In	Pricing_Attr_val_tbl_type	No	No
p_check_duplicate_lines*	In	VARCHAR2	No	No
Note: The default value is Y (Yes).				

***Note:** p_check_duplicate_lines: This parameter controls the duplicate check for price list lines through QP_Price_List_PUB.Process_Price_list and the OE_Pricing_Cont_PUB.Process_Agreement API. The parameter is optional and the default value is set to NULL meaning the API does the duplicate check. For any parameter value other than N (No), then the API checks for duplicate lines. The global variable G_CHECK_DUP_PRICELIST_LINES defined in package QP_GLOBALS holds the passed parameter value for the session.

PRICE_LIST_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_REC_TYPE

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	No	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
currency_code	Null	Varchar2	Yes	No
discount_lines_flag	Null	Varchar2	No	No
end_date_active	Null	Date	No	No
freight_terms_code	Null	Varchar2	No	No
global_flag	Null	Varchar2	No	No
gsa_indicator	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_source_code	Null	Varchar2	No	No
list_type_code	Null	Varchar2	No	No
org_id	Null	Number	No	Yes
orig_system_he ader_ref	Null	Varchar2	No	No
program_applica tion_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update _date	Null	Date	No	No
prorate_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
rounding_factor	Null	Number	No	No
ship_method_co de	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
terms_id	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
operation	Null	Varchar2	Yes	No
name	Null	Varchar2	Yes	No
description	Null	Varchar2	No	No
version_no	Null	Varchar2	No	No
Active_flag (Default value is Yes.)	Null	Varchar2	No	No
mobile_download	Null	Varchar2	No	No
currency_header_id	Null	Number	No	No
locked_from_list_header_id	Null	Number	No	No
Source_System_code	Null	Varchar2	Yes	No
Shareable_flag	Null	Varchar2	Yes	No
Sold_to_org_id	Null	Number	Yes	No

Note: This API will initialize MOAC if it is not initialized prior. On the price_list_rec, if the global_flag is passed as FND_API.G_MISS_CHAR, the API will default the global_flag as "Y." If global_flag is passed as "N" and if org_id is not null, the API will create the modifier list for that operating unit (OU). If global_flag is passed as "N" and org_id is not passed, the API will derive the org_id from the org context if set; if not, use the default OU for that responsibility. The API will validate the Org_id to check if the Org_Id corresponds to one of the OUs for that responsibility and raise an error if org_id is not valid.

PRICE_LIST_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Rec_Type	Null	Record	No	No

PRICE_LIST_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
automatic	Null	Varchar2	No	No
currency	Null	Varchar2	No	No
discount_lines	Null	Varchar2	No	No
freight_terms	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_type	Null	Varchar2	No	No
prorate	Null	Varchar2	No	No
ship_method	Null	Varchar2	No	No
terms	Null	Varchar2	No	No
currency_header	Null	Varchar2	No	No

PRICE_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Val_Rec_Type	Null	Record	No	No

PRICE_LIST_LINE_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_REC_TYPE

Parameter	Usage	Type	Req	Drv
accrual_qty	Null	Number	No	No
accrual_uom_code	Null	Varchar2	No	No
arithmetic_operator	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	No	No
base_qty	Null	Number	No	No
base_uom_code	Null	Varchar2	No	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
effective_period_uom	Null	Varchar2	No	No
end_date_active	Null	Date	No	No
estim_accrual_rate	Null	Number	No	No
generate_using_formula_id	Null	Number	No	No
inventory_item_id	Null	Number	No	No
last_updated_by	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
list_line_type_co de	Null	Varchar2	No	No
list_price	Null	Number	No	No
modifier_level_c ode	Null	Varchar2	No	No
number_effectiv e_periods	Null	Number	No	No
operand	Null	Number	No	No
organization_id	Null	Number	No	No
override_flag	Null	Varchar2	No	No
percent_price	Null	Number	No	No
price_break_type _code	Null	Varchar2	No	No
price_by_formul a_id	Null	Number	No	No
primary_uom_fl ag	Null	Varchar2	No	No
print_on_invoice _flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
rebate_txn_type_code	Null	Varchar2	No	No
related_item_id	Null	Number	No	No
relationship_type_id	Null	Number	No	No
reprice_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
revision	Null	Varchar2	No	No
revision_date	Null	Date	No	No
revision_reason_code	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
substitution_attribute	Null	Varchar2	No	No
substitution_context	Null	Varchar2	No	No
substitution_value	Null	Varchar2	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
operation	Null	Varchar2	No	No
from_rltd_modifier_id	Null	Number	No	No
rltd_modifier_group_no	Null	Number	No	No
product_precedence	Null	Number	No	No
break_uom_code	Null	Varchar2	No	No
break_uom_context	Null	Varchar2	No	No
break_uom_attribute	Null	Varchar2	No	No
Customer_Item_Id	Null	Number	Yes	No

PRICE_LIST_LINE_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Line_Rec_Type	Null	Record	No	No

PRICE_LIST_LINE_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
accrual_uom	Null	Varchar2	No	No
automatic	Null	Varchar2	No	No
base_uom	Null	Varchar2	No	No
generate_using_formula	Null	Varchar2	No	No
inventory_item	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
list_line_type	Null	Varchar2	No	No
modifier_level	Null	Varchar2	No	No
organization	Null	Varchar2	No	No
override	Null	Varchar2	No	No
price_break_type	Null	Varchar2	No	No
price_by_formula	Null	Varchar2	No	No
primary_uom	Null	Varchar2	No	No
print_on_invoice	Null	Varchar2	No	No
rebate_transaction_type	Null	Varchar2	No	No
related_item	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
relationship_type	Null	Varchar2	No	No
reprice	Null	Varchar2	No	No
revision_reason	Null	Varchar2	No	No

PRICE_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Line_Val_Rec_Type	Null	Record	No	No

QUALIFIERS_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_REC_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Rec_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

QUALIFIERS_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_TBL_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Tbl_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

QUALIFIERS_VAL_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Val_Rec_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

QUALIFIERS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Val_Tbl_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

Note: For setting up a Secondary Price List, create a qualifier with the following parameters:

- QUALIFIER_CONTEXT: MODLIST

- QUALIFIER_ATTRIBUTE: QUALIFIER_ATTRIBUTE4
- QUALIFIER_ATTR_VALUE: <list_header_id of the primary price list>
- LIST_HEADER_ID: <list_header_id of the secondary price list>
- COMPARISON_OPERATOR_CODE: =

PRICING_ATTR_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_REC_TYPE

Parameter	Usage	Type	Req	Drv
accumulate_flag	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
attribute_groupi ng_number	Null	Number	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
excluder_flag	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_line_id	Null	Number	No	No
pricing_attribute	Null	Varchar2	No	No
pricing_attribute _context	Null	Varchar2	No	No
pricing_attribute _id	Null	Number	No	No
pricing_attr_valu e_from	Null	Varchar2	No	No
pricing_attr_valu e_to	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
product_attribute	Null	Varchar2	Yes	No
product_attribute_context	Null	Varchar2	Yes	No
product_attribute_value	Null	Varchar2	Yes	No
product_uom_code	Null	Varchar2	Yes	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
request_id	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No
PRICE_LIST_LINE_index	Null	Number	No	No
from_rlt_modifier_id	Null	Number	No	No
comparison_operator_code	Null	Varchar2	Yes	No
product_attribute_datatype	Null	Varchar2	No	Yes

Parameter	Usage	Type	Req	Drv
pricing_attribute_datatype	Null	Varchar2	No	Yes

PRICING_ATTR_TBL_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Rec_Type	Null	Record	No	No

PRICING_ATTR_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
accumulate	Null	Varchar2	No	No
excluder	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
pricing_attribute	Null	Varchar2	No	No
product_uom	Null	Varchar2	No	No

PRICING_ATTR_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Val _Rec_Type	Null	Record	No	No

Validation of Price List Setup API

Validations:

Standard Validation

Oracle Advanced Pricing validates all required columns in the Price List Setup API.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Price List Setup API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Example of Price List Setup API

The following example scripts are located in the directory \$qp/patch/115/sql:

- QPPLXMP1.sql: This sample script inserts three price list lines into a price list with and the product information for each of the lines (product information is stored in pricing attributes table in product attribute columns). This sample price list does not have any qualifiers or price breaks or non product-information type of pricing attributes.
- QPPLXMP2.sql: This sample script inserts three price list lines into a price list and the product information for each of the lines (product information is stored in pricing attributes table in product attribute columns); one secondary price list; two price list qualifiers (only header level qualifiers supported for price lists); and two

non-product pricing attributes per price list line.

- QPPLXMP3.sql: This script inserts one price list line into a price list with product information for this line (product information is stored in pricing attributes table in product attribute columns); one non-product pricing attribute for the price list line; and one price list qualifier (only header level qualifiers supported for price lists).
- QPPLXMP4.sql: This script inserts one price list line of type PBH (price break header) into a price list and the product information for this line (product information is stored in pricing attributes table in product attribute columns); a regular pricing attribute (non product) and a price break child line. To create a price break child line, you must create a combination of a list line and a pricing attribute where the pricing attribute can only have the Volume pricing context and Pricing Attribute of Item Quantity.
- QPPLXMP5.sql: This script inserts the list price on a price list line. This script must be modified by the user such that the `qpr_list_line_tbl(K).list_line_id` column is populated with a valid `list_line_id` from the instance where this script is run.

Price List Setup Group Application Program Interface

This section explains how to use the Price List Setup Group API and how it functions in Oracle Advanced Pricing. The Price List Setup package consists of entities to set up price lists.

Functional Overview

The Price List Setup package `QP_Price_List_GRP.Process_Price_List` contains the following public record type and table of records entities:

- `Process_Price_List`: `QP_Price_List_GRP.Process_Price_List`: Takes two record types and six table types as input parameters. Use this API to insert, update, and delete price lists and to set up a price list for a given `p_PRICE_LIST_rec` record structure.

You can:

- Set up multiple price list lines by giving multiple price list line definitions in the `p_PRICE_LIST_LINE_tbl` table structure.
- Attach multiple qualifiers at the price list header level by giving multiple qualifiers in the `p_QUALIFIERS_tbl` table structure.
- Attach multiple pricing attributes to price list lines by giving the pricing attributes in the `p_PRICING_ATTR_tbl` table structure.
- `Price_List_Rec_Type`: Corresponds to the columns in the price list header tables

QP_LIST_HEADERS_B and QP_LIST_HEADERS_TL.

- Price_List_Val_Rec_Type: Attributes that store the meaning of id or code columns in the price list header table QP_LIST_HEADERS_B, for example, Currency.
- Price_List_Line_Rec_Type: Corresponds to columns in the price list line table and related modifiers tables QP_LIST_LINES and QP_RLTD_MODIFIERS.
- Price_List_Line_Tbl_Type: Table of Price_List_Line_Rec_Type.
- Price_List_Line_Val_Rec_Type: Attributes that store the meaning of id or code columns in the price list line table QP_LIST_LINES, for example, Price_By_Formula.
- Price_List_Line_Val_Tbl_Type: Table of Price_List_Line_Val_Rec_Type.
- Qualifiers_Rec_Type: Corresponds to the columns in the qualifier table QP_QUALIFIERS.
- Qualifiers_Tbl_Type: Table of Qualifiers_Rec_Type.
- Qualifiers_Val_Rec_Type: Made up of attributes that store the meaning of id or code columns in the qualifiers table QP_QUALIFIERS, for example, Qualifier_Rule.
- Qualifiers_Val_Tbl_Type: Table of Qualifiers_Val_Rec_Type.
- Pricing_Attr_Rec_Type: Corresponds to the columns in the pricing attributes table QP_PRICING_ATTRIBUTES.
- Pricing_Attr_Tbl_Type: Table of Pricing_Attr_Rec_Type.
- Pricing_Attr_Val_Rec_Type: Attributes that store the meaning of id or code columns in the pricing attributes table QP_PRICING_ATTRIBUTES, for example, Accumulate.
- Pricing_Attr_Val_Tbl_Type: Table of Pricing_Attr_Val_Rec_Type.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the Price List Group API. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

PROCESS_PRICE_LIST

The following table shows the parameters for this structure.

PROCESS_PRICE_LIST

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Number	No	No
x_msg_data	Out	Varchar2	No	No
p_PRICE_LIST_rec	In	Price_List_Rec_Type	No	No
p_PRICE_LIST_val_rec	In	Price_List_Val_Rec_Type	No	No
p_PRICE_LIST_tbl	In	Price_List_Line_tbl_type	No	No
p_PRICE_LIST_val_tbl	In	Price_List_Line_Val_tbl_type	No	No
p_QUALIFIERS_tbl	In	QP_Qualifier_Rules_PUB.Qualifiers_tbl_Type	No	No
p_QUALIFIERS_val_tbl	In	QP_Qualifier_Rules_PUB.Qualifiers_val_tbl_Type	No	No
p_PRICING_ATTR_tbl	In	Pricing_Attr_tbl_type	No	No

Parameter	Usage	Type	Req	Drv
p_PRICING_ATTR_val_tbl	In	Pricing_Attr_val_tbl_type	No	No
x_PRICE_LIST_rec	Out	Price_List_Rec_Type	No	No
x_PRICE_LIST_val_rec	Out	Price_List_Val_Rec_Type	No	No
x_PRICE_LIST_Line_tbl	Out	Price_List_Line_tbl_type	No	No
x_PRICE_LIST_Line_val_tbl	Out	Price_List_Line_Val_tbl_type	No	No
x_QUALIFIERS_tbl	Out	QP_Qualifier_Rules_PUB.Qualifiers_tbl_Type	No	No
x_QUALIFIERS_val_tbl	Out	QP_Qualifier_Rules_PUB.Qualifiers_val_tbl_Type	No	No
x_PRICING_ATTR_tbl	Out	Pricing_Attr_tbl_type	No	No
x_PRICING_ATTR_val_tbl	Out	Pricing_Attr_val_tbl_type	No	No

A key of the short names and definitions used in the API tables are provided in the following table:

Short Name Key

Short name	Definition
Drv	Derived

Short name	Definition
Req	Required Yes : This is a required parameter. No : This is an optional parameter.
N/A (no entry)	No value/not applicable

PRICE_LIST_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_REC_TYPE

Parameter	Usage	Type	Req	Drv
Attribute1	Null	Varchar2	No	No
Attribute2	Null	Varchar2	No	No
Attribute3	Null	Varchar2	No	No
Attribute4	Null	Varchar2	No	No
Attribute5	Null	Varchar2	No	No
Attribute6	Null	Varchar2	No	No
Attribute7	Null	Varchar2	No	No
Attribute8	Null	Varchar2	No	No
Attribute9	Null	Varchar2	No	No
Attribute10	Null	Varchar2	No	No
Attribute11	Null	Varchar2	No	No
Attribute12	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
Attribute13	Null	Varchar2	No	No
Attribute14	Null	Varchar2	No	No
Attribute15	Null	Varchar2	No	No
automatic_flag	Null	Varchar2	No	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
currency_code	Null	Varchar2	Yes	No
discount_lines_flag	Null	Varchar2	No	No
end_date_active	Null	Date	No	No
freight_terms_code	Null	Varchar2	No	No
gsa_indicator	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
list_header_id	Null	Number	No	No
list_type_code	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
prorate_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
rounding_factor	Null	Number	No	No
ship_method_code	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
terms_id	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No
name	Null	Varchar2	Yes	No
description	Null	Varchar2	No	No
version_no	Null	Varchar2	No	No
Active_flag	Null	Varchar2	No	No
mobile_download	Null	Varchar2	No	No
currency_header_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
list_source_code	Null	Varchar2	No	No
orig_system_header_ref	Null	Varchar2	No	No
global_flag	Null	Varchar2	No	No
org_id	Null	Number	No	No
Source_System_code	Null	Varchar2	Yes	No
Shareable_flag	Null	Varchar2	Yes	No
Sold_to_org_id	Null	Number	Yes	No

Note: On the price_list_rec, if the global_flag is passed as FND_API.G_MISS_CHAR, the API will default the global_flag as "Y." If global_flag is passed as "N" and if org_id is not null, the API will create the modifier list for that operating unit (OU). If global_flag is "N" and Org_Id is not passed, the API will derive the org_id from the org context (if set); if not, use the default OU for that responsibility. The API will validate the Org_id to check if the Org_Id corresponds to one of the OUs for that responsibility and raise an error if the org_id is not valid.

PRICE_LIST_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Rec_Type	Null	Record	No	No

PRICE_LIST_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
automatic	Null	Varchar2	No	No
Currency	Null	Varchar2	No	No
discount_lines	Null	Varchar2	No	No
freight_terms	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_type	Null	Varchar2	No	No
Prorate	Null	Varchar2	No	No
ship_method	Null	Varchar2	No	No
Terms	Null	Varchar2	No	No
currency_header	Null	Varchar2	No	No

PRICE_LIST_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Val_Record_Type	Null	Record	No	No

PRICE_LIST_LINE_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_REC_TYPE

Parameter	Usage	Type	Req	Drv
accrual_qty	Null	Number	No	No
accrual_uom_code	Null	Varchar2	No	No
arithmetic_operator	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
Attribute6	Null	Varchar2	No	No
Attribute7	Null	Varchar2	No	No
Attribute8	Null	Varchar2	No	No
Attribute9	Null	Varchar2	No	No
Attribute10	Null	Varchar2	No	No
Attribute11	Null	Varchar2	No	No
Attribute12	Null	Varchar2	No	No
Attribute13	Null	Varchar2	No	No
Attribute14	Null	Varchar2	No	No
Attribute15	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
automatic_flag	Null	Varchar2	No	No
base_qty	Null	Number	No	No
base_uom_code	Null	Varchar2	No	No
comments	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
effective_period_uom	Null	Varchar2	No	No
end_date_active	Null	Date	No	No
estim_accrual_rate	Null	Number	No	No
generate_using_formula_id	Null	Number	No	No
inventory_item_id	Null	Number	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_login	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
list_line_type_code	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
list_price	Null	Number	No	No
modifier_level_code	Null	Varchar2	No	No
number_effective_periods	Null	Number	No	No
operand	Null	Number	No	No
organization_id	Null	Number	No	No
override_flag	Null	Varchar2	No	No
percent_price	Null	Number	No	No
price_break_type_code	Null	Varchar2	No	No
price_by_formula_id	Null	Number	No	No
primary_uom_flag	Null	Varchar2	No	No
print_on_invoice_flag	Null	Varchar2	No	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
rebate_txn_type_code	Null	Varchar2	No	No
related_item_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
relationship_type_id	Null	Number	No	No
reprice_flag	Null	Varchar2	No	No
request_id	Null	Number	No	No
revision	Null	Varchar2	No	No
revision_date	Null	Date	No	No
revision_reason_code	Null	Varchar2	No	No
start_date_active	Null	Date	No	No
substitution_attribute	Null	Varchar2	No	No
substitution_context	Null	Varchar2	No	No
substitution_value	Null	Varchar2	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No
from_rltd_modifier_id	Null	Number	No	No
rltd_modifier_group_no	Null	Number	No	No
product_precedence	Null	Number	No	No

PRICE_LIST_LINE_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Line_Rec_Type	Null	Record	No	No

PRICE_LIST_LINE_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
accrual_uom	Null	Varchar2	No	No
automatic	Null	Varchar2	No	No
base_uom	Null	Varchar2	No	No
generate_using_formula	Null	Varchar2	No	No
inventory_item	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
list_line_type	Null	Varchar2	No	No
modifier_level	Null	Varchar2	No	No
organization	Null	Varchar2	No	No
override	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
price_break_type	Null	Varchar2	No	No
price_by_formula	Null	Varchar2	No	No
primary_uom	Null	Varchar2	No	No
print_on_invoice	Null	Varchar2	No	No
rebate_transaction_type	Null	Varchar2	No	No
related_item	Null	Varchar2	No	No
relationship_type	Null	Varchar2	No	No
reprice	Null	Varchar2	No	No
revision_reason	Null	Varchar2	No	No

PRICE_LIST_LINE_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICE_LIST_LINE_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Price_List_Line_Val_Rec_Type	Null	Record	No	No

QUALIFIERS_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_REC_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Rec_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

QUALIFIERS_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_TBL_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Tbl_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

QUALIFIERS_VAL_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Val_Rec_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

QUALIFIERS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
QP_Qualifier_Rules_PUB.Qualifiers_Val_Tbl_Type	Null	Record	No	No

Refer to the Qualifiers public API for the definition.

To set up a secondary price list, create a qualifier with the following parameters:

- QUALIFIER_CONTEXT: MODLIST
- QUALIFIER_ATTRIBUTE: QUALIFIER_ATTRIBUTE4
- QUALIFIER_ATTR_VALUE: <list_header_id of the primary price list>
- LIST_HEADER_ID: <list_header_id of the secondary price list>
- COMPARISON_OPERATOR_CODE: =

PRICING_ATTR_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_REC_TYPE

Parameter	Usage	Type	Req	Drv
accumulate_flag	Null	Varchar2	No	No
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
attribute_groupi ng_number	Null	Number	No	No
Context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
excluder_flag	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
list_line_id	Null	Number	No	No
pricing_attribute	Null	Varchar2	No	No
pricing_attribute_context	Null	Varchar2	No	No
pricing_attribute_id	Null	Number	No	No
pricing_attr_value_from	Null	Varchar2	No	No
pricing_attr_value_to	Null	Varchar2	No	No
product_attribute	Null	Varchar2	Yes	No
product_attribute_context	Null	Varchar2	Yes	No
product_attr_value	Null	Varchar2	Yes	No
product_uom_code	Null	Varchar2	Yes	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
request_id	Null	Number	No	No
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
Operation	Null	Varchar2	Yes	No
PRICE_LIST_LI NE_index	Null	Number	No	No
from_rltd_modif ier_id	Null	Number	No	No
comparison_ope rator_code	Null	Varchar2	Yes	No
product_attribut e_datatype	Null	Varchar2	No	Yes
pricing_attribute _datatype	Null	Varchar2	No	Yes

PRICING_ATTR_TBL_TYPE

The following table shows the parameters for this structure:

PRICING_ATTR_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Rec _Type	Null	Record	No	No

PRICING_ATTR_VAL_REC_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
Accumulate	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
Excluder	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
pricing_attribute	Null	Varchar2	No	No
product_uom	Null	Varchar2	No	No

PRICING_ATTR_VAL_TBL_TYPE

The following table shows the parameters for this structure.

PRICING_ATTR_VAL_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Pricing_Attr_Val _Rec_Type	Null	Record	No	No

Validation of Price List Group API

Validations:

Standard Validation

Oracle Advanced Pricing validates all required columns in the Price List Group API. For more information, see: *Oracle Pricing Technical Reference Manual*.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Price List Group API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Price Request Application Program Interface

This section explains how to use the Price Request API and how it functions in Oracle Pricing. The Price Request group API has procedures to be called to pass the request information to the pricing engine. This API also has the different constants that the pricing engine uses for error status, codes and so on.

The Price Request Application Program Interface (API) is a public API that allows you to get a base price and to apply price adjustments, other benefits, and charges to a transaction. Oracle Applications products request it for pricing calculations and you can request it from custom applications and legacy systems.

A pricing request consists of numerous price request lines which mirror the transaction lines of the calling application and may include a transaction header request line. Since it is PL/SQL based, the pricing request processes one pricing request per call.

To properly use the Price Request Application Program Interface, pass all lines that need prices and that the pricing engine needs to consider as part of pricing request. For example, you may have frozen the price of one order line but, if you include it in the pricing request, the pricing engine may be able to use the quantity on that line to qualify the order to receive another discount based on quantities across multiple lines.

The Price Request Application Program Interface consists of two engines:

- Search engine: Uses qualifiers and pricing attributes passed from the calling application to select the price list lines and the modifier list lines that may apply to the pricing request. As part of this process, the search engine uses rules of eligibility, incompatibility, exclusivity, and precedence.

For each pricing phase the search engine executes the following functions:

- Selects eligible price list lines and modifier list lines using predefined pricing rules.
 - Resolve incompatibilities among eligible benefits.
 - Applies the eligible benefits to the pricing request.
- Calculation engine: For each pricing request line and its associated pricing request

line details, calculates the base price, adjusted price, and extended price.

You can call one or both of the engines by setting the calculate flag on the control record.

Functional Overview

The Price Request public API QP_PREQ_PUB contains the following entities:

- **CONTROL_RECORD_TYPE:** Parameters which control the behavior of the pricing engine.
- **LINE_REC_TYPE:** A record which contains the elements in the calling application that requires a base and adjusted price. It may equate to a transaction line or transaction header record in the calling application.
- **LINE_DETAIL_REC_TYPE:** A record that contains the details of the derivation of the base and adjusted prices. Each pricing request line detail provides details for a price list line or modifier list line. The pricing engine may apply many pricing request line detail records to a pricing request.
- **QUAL_REC_TYPE:** A record that contains qualifier information. Qualifier information helps the pricing engine to determine the price list lines and modifier list lines for which a pricing request is eligible. The pricing engine returns all qualifiers that determined eligibility on the request line detail record.
- **LINE_ATTR_REC_TYPE:** A record that contains pricing attributes. Pricing attribute information helps the pricing engine to determine the price list lines and modifier list lines for which a pricing request is eligible. The calling application must load all pricing attributes into this record type that the pricing engine should use to determine if a pricing request line qualifies for a price or modifier.
- **RELATED_LINES_REC_TYPE:** A record that contains relationships between request lines and between request lines and request line details. Types of relationships are as follows:
 - **PBH_LINE:** Relates a price break header to modifier price break lines.
 - **SERVICE_LINE:** Relates an order line for a service item and its parent, the serviceable item. The pricing engine needs to know this relationship when it must price service items based on a percent of the serviceable item price.
 - **GENERATED_LINE:** Indicates the lines—both request and detail that a pricing request line or pricing request detail line created.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Price Request. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

Insert_Lines2

This API takes the request line information and does a bulk insert into the pricing temporary table which holds the request lines, qp_preq_lines_tmp. Each of the input parameter is a pl/sql table to enable the bulk insert. If an error occurs, this API returns x_return_status as FND_API.G_RET_STS_ERROR.

QP_PREQ_GRP.INSERT_LINES2

Note: The value for qp_preq_lines_tmp.price_list_header_id has precedence over the sourced price list value. For example, if a consumer application passes the price list id in qp_preq_lines_tmp.price_list_header_id, but also has a sourcing rule for the price list attribute, the value in qp_preq_lines_tmp takes precedence.

The following table shows the parameters for this structure:

QP_PREQ_GRP.INSERT_LINES2

Parameter	Usag e	Type
p_LINE_INDEX	In	QP_PREQ_GRP.PLS_INTEGER_TYPE,
p_LINE_TYPE_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_PRICING_EFFECTIVE_DATE	In	QP_PREQ_GRP.DATE_TYPE
p_ACTIVE_DATE_FIRST	In	QP_PREQ_GRP.DATE_TYPE
p_ACTIVE_DATE_FIRST_TYPE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_ACTIVE_DATE_SECOND	In	QP_PREQ_GRP.DATE_TYPE
p_ACTIVE_DATE_SECOND_TYPE	In	QP_PREQ_GRP.VARCHAR_TYPE ,
p_LINE_QUANTITY	In	QP_PREQ_GRP.NUMBER_TYPE ,

Parameter	Usage	Type
p_LINE_UOM_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_REQUEST_TYPE_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_PRICED_QUANTITY	In	QP_PREQ_GRP.NUMBER_TYPE,
p_PRICED_UOM_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_CURRENCY_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_UNIT_PRICE	In	QP_PREQ_GRP.NUMBER_TYPE,
p_PERCENT_PRICE	In	QP_PREQ_GRP.NUMBER_TYPE,
p_UOM_QUANTITY	In	QP_PREQ_GRP.NUMBER_TYPE,
p_ADJUSTED_UNIT_PRICE	In	QP_PREQ_GRP.NUMBER_TYPE,
p_UPD_ADJUSTED_UNIT_PRICE	In	QP_PREQ_GRP.NUMBER_TYPE,
p_PROCESSED_FLAG	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_PRICE_FLAG	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_LINE_ID	In	QP_PREQ_GRP.NUMBER_TYPE,
p_PROCESSING_ORDER	In	QP_PREQ_GRP.PLS_INTEGER_TYPE,
p_PRICING_STATUS_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_PRICING_STATUS_TEXT	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_ROUNDING_FLAG	In	QP_PREQ_GRP.FLAG_TYPE,
p_ROUNDING_FACTOR	In	QP_PREQ_GRP.PLS_INTEGER_TYPE,
p_QUALIFIERS_EXIST_FLAG	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_PRICING_ATTRS_EXIST_FLAG	In	QP_PREQ_GRP.VARCHAR_TYPE,

Parameter	Usage	Type
p_PRICE_LIST_ID	In	QP_PREQ_GRP.NUMBER_TYPE,
p_VALIDATED_FLAG	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_PRICE_REQUEST_CODE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_USAGE_PRICING_TYPE	In	QP_PREQ_GRP.VARCHAR_TYPE,
p_LINE_CATEGORY	In	QP_PREQ_GRP.VARCHAR_TYPE := QP_PREQ_GRP.G_LINE_CATEGORY_DEF_TBL
x_status_code	OUT	VARCHAR2
x_status_text	OUT	VARCHAR2

For information on each of these parameters, refer the LINE_REC_TYPE defined in this manual under the price_request API.

Insert_Line_Attrs2

This API takes the attribute information and does a bulk insert into the pricing temporary table which holds the request line attributes, qp_preq_line_attrs_tmp. Each of the input parameter is a pl/sql table to enable the bulk insert. In case there is an error, this API returns x_return_status as FND_API.G_RET_STS_ERROR.

QP_PREQ_GRP.INSERT_LINE_ATTRS2

Parameter	Type
p_LINE_INDEX_tbl	QP_PREQ_GRP.pls_integer_type
p_LINE_DETAIL_INDEX_tbl	QP_PREQ_GRP.pls_integer_type
p_ATTRIBUTE_LEVEL_tbl	QP_PREQ_GRP.varchar_type
p_ATTRIBUTE_TYPE_tbl	QP_PREQ_GRP.varchar_type
p_LIST_HEADER_ID_tbl	QP_PREQ_GRP.number_type

Parameter	Type
p_LIST_LINE_ID_tbl	QP_PREQ_GRP.number_type
p_CONTEXT_tbl	QP_PREQ_GRP.varchar_type
p_ATTRIBUTE_tbl	QP_PREQ_GRP.varchar_type
p_VALUE_FROM_tbl	QP_PREQ_GRP.varchar_type
p_SETUP_VALUE_FROM_tbl	QP_PREQ_GRP.varchar_type
p_VALUE_TO_tbl	QP_PREQ_GRP.varchar_type
p_SETUP_VALUE_TO_tbl	QP_PREQ_GRP.varchar_type
p_GROUPING_NUMBER_tbl	QP_PREQ_GRP.pls_integer_type
p_NO_QUALIFIERS_IN_GRP_tbl	QP_PREQ_GRP.pls_integer_type
p_COMPARISON_OPERATOR_TYPE_tbl	QP_PREQ_GRP.varchar_type
p_VALIDATED_FLAG_tbl	QP_PREQ_GRP.varchar_type
p_APPLIED_FLAG_tbl	QP_PREQ_GRP.varchar_type
p_PRICING_STATUS_CODE_tbl	QP_PREQ_GRP.varchar_type
p_PRICING_STATUS_TEXT_tbl	QP_PREQ_GRP.varchar_type
p_QUALIFIER_PRECEDENCE_tbl	QP_PREQ_GRP.pls_integer_type
p_DATATYPE_tbl	QP_PREQ_GRP.varchar_type
p_PRICING_ATTR_FLAG_tbl	QP_PREQ_GRP.varchar_type
p_QUALIFIER_TYPE_tbl	QP_PREQ_GRP.varchar_type
p_PRODUCT_UOM_CODE_TBL	QP_PREQ_GRP.varchar_type
p_EXCLUDER_FLAG_TBL	QP_PREQ_GRP.varchar_type

Parameter	Type
p_PRICING_PHASE_ID_TBL	QP_PREQ_GRP.pls_integer_type
p_INCOMPATABILITY_GRP_CODE_TBL	QP_PREQ_GRP.varchar_type
p_LINE_DETAIL_TYPE_CODE_TBL	QP_PREQ_GRP.varchar_type
p_MODIFIER_LEVEL_CODE_TBL	QP_PREQ_GRP.varchar_type
p_PRIMARY_UOM_FLAG_TBL	QP_PREQ_GRP.varchar_type
x_status_code	OUT VARCHAR2
x_status_text	OUT VARCHAR2);

For more information on each of these input parameters, refer to the line_attr_rec_type in the price_request API.

PRICE_REQUEST

The Derived value for each parameter is Null.

PRICE_REQUEST

Parameter	Usage	Type	Req
p_line_tbl	In	LINE_TBL_TYPE	Yes
p_qual_tbl	In	QUAL_TBL_TYPE	No
p_line_attr_tbl	In	LINE_ATTR_TBL_TYPE	No
p_line_detail_tbl	In	LINE_DETAIL_TBL_TYPE	No
p_line_detail_qual_tbl	In	LINE_DETAIL_QUAL_TBL_TYPE	No
p_line_detail_attr_tbl	In	LINE_DETAIL_ATTR_TBL_TYPE	No

Parameter	Usage	Type	Req
p_related_lines_tbl	In	RELATED_LINES_TB L_TYPE	No (req. for Service Line Pricing)
p_control_rec	In	CONTROL_ RECORD_TYPE	Yes
x_line_tbl	Out	LINE_TBL_TYPE	Yes
x_qual_tbl	Out	QUAL_TBL_TYPE	No
x_line_attr_tbl	Out	LINE_ATTR_ TBL_TYPE	No
x_line_detail_tbl	Out	LINE_DETAIL_ TBL_TYPE	Yes
x_line_detail_qual_tbl	Out	LINE_DETAIL_ QUAL_TBL_TYPE	No
x_line_detail_attr_tbl	Out	LINE_DETAIL_ ATTR_TBL_TYPE	No
x_related_lines_tbl	Out	RELATED_LINES_TB L_TYPE	No
x_return_status	Out	Varchar2	Yes
x_return_status_text	Out	Varchar2	Yes

PRICE_REQUEST (Overloaded)

This API has been overloaded for performance features. The request line and attributes are inserted into the pricing temporary tables prior to calling this API. Please refer to the performance features section under this API for more details regarding this call.

PRICE_REQUEST (Overloaded)

Parameter	Usage	Type	Req
p_control_rec	In	CONTROL_REC_TYPE	Yes
x_return_status	Out	Varchar2	Yes
x_return_status_text	Out	Varchar2	Yes

CONTROL_REC_TYPE

The Derived value for each parameter is Null.

CONTROL_REC_TYPE

Parameter	Type	Req	Usage
PRICING_EVENT	Varchar2	Yes	A point in the transaction life cycle of your transaction at which you wish to price it. The pricing event determines which phases the search engine processes according to the mapping in QP_EVENT_PHASES .

Parameter	Type	Req	Usage
CALCULATE_FLAG	Varchar2	Yes	<p>Use to call the engines. Allowable values are:</p> <p>QP_PREQ_GRP. G_SEARCH_ONLY: Search engine: If you do not want the engine to calculate the selling price.</p> <p>QP_PREQ_GRP.G_C ALCULATE_ONLY: Calculate engine: If you are passing the adjustment records to the engine and you want the engine to recalculate the selling price, without retrieving new adjustments.</p> <p>QP_PREQ_GRP. G_SEARCH_N_CALCULATE: Both the calculate and search engines: Regular engine call. Retrieves new adjustments and calculates the selling price</p>
SIMULATION_FLAG	Varchar2	Yes	<p>A value of Yes indicates that the call to is a pricing simulation and that the pricing engine should not make permanent record changes and neither issue or redeem coupons.</p>

Parameter	Type	Req	Usage
ROUNDING_FLAG	Varchar2	Yes	<p>Indicates whether the calculation engine should round the list price and selling price based on the price list rounding factor or the pricing request line record rounding factor. When rounding, the calculation engine rounds all intermediate subtotals for each pricing group sequence. By default, the rounding is enabled in the pricing engine unless the rounding flag is explicitly set to No.</p> <p>Allowable values are:</p> <p>Y: engine applies the rounding factor defined in the price list</p> <p>N: unrounded figures would be returned</p> <p>Q: Refer to the value of the profile QP: Selling Price Rounding Options</p>

Parameter	Type	Req	Usage
GSA_CHECK_FLAG	Varchar2	No	<p>Indicates whether the GSA Violation needs to be checked, provided the customer is not a GSA customer and the selling price falls below the price on the GSA price list.</p> <p>Allowable values are:</p> <p>Y: engine applies the rounding factor defined in the price list</p> <p>N: unrounded figures are returned</p>
TEMP_TABLE_INSE RT_FLAG	Varchar2	No	<p>Indicates if the calling application wants the engine to insert the order lines into the QP temporary table(s). The values for this are:</p> <p>Y: The pricing engine will insert the lines into the QP temporary tables</p> <p>N – This means to the engine that the order lines are already there in the temporary tables.</p>

Parameter	Type	Req	Usage
MANUAL_DISCOU NT_FLAG	Varchar2	No	<p>This flag is introduced to support new Release 11i functionality. This value is set by the calling application and the value is based on the profile QP: Return Manual Discounts.</p> <p>Indicates how pricing engine should perform incompatibility processing for manual discounts. he possible values for this profile are:</p> <p><i>Yes:</i> All the manual discounts will be returned. All the automatic discounts that get deleted as part of incompatibility processing will be returned as manual discounts.</p> <p><i>No:</i> All automatic and manual discounts will go through incompatibility processing and one of them in each incompatibility group will be returned. In this process an automatic discount might get deleted and a manual discount might get selected.</p>

Parameter	Type	Req	Usage
SOURCE_ORDER_A MOUNT_FLAG	Varchar2	No	<p>This is for internal use of the pricing engine. Indicates whether the pricing engine will source the order amount attribute.</p> <p>Allowable values are:</p> <p>Y: Indicates to the pricing engine to source the order amount. It means the calling application will provide the order amount</p> <p>N: The pricing engine should calculate the order amount</p>
CHECK_CUST_VIE W_FLAG	Varchar2	No	<p>This is for internal use only. Indicates whether the pricing engine should take a look at caller's pricing tables.</p> <p>Allowable values are:</p> <p>Y: Indicates that to access the calling applications data, the source system uses the view provided by calling application.</p> <p>N: Indicates that the calling application directly provides the data to the source system</p>
REQUEST_TYPE_CO DE	Varchar2	Yes	<p>Identifies the transaction system making the pricing request.</p>

Parameter	Type	Req	Usage
VIEW_CODE	Varchar2	No	<p>This is for internal use only. Indicates the view code for the transaction system of the caller.</p> <p>It indicates database view id, which the source system uses to access the calling application's data.</p>
USE_MULTI_CURRENCY	Varchar2	No	<p>Indicates if the calling application wants to use multi-currency.</p> <p>Valid values: 'Y' or 'N'</p> <p>Default value: 'N'</p>
USER_CONVERSION_RATE	Number	No	User defined Conversion rate, used for multi-currency
USER_CONVERSION_TYPE	Varchar2	No	User defined conversion type, used for multi-currency
FUNCTION_CURRENCY	Varchar2	No	Functional currency, used for multi-currency
FULL_PRICING_CALL	Varchar2	No	Indicates if the calling application passed only changed lines to the pricing engine or all the lines during reprice. Set to 'Y' if passed all the lines, 'N' if passed only changed lines. Refer to the changed lines feature

Parameter	Type	Req	Usage
GET_FREIGHT_FLAG	Varchar2	No	When the control record get_freight_flag is set to Y, the pricing engine will only process the phases that have freight charges existing. The pricing engine will return the modifiers (including freight charge modifiers and non-freight charge modifiers) that are qualified in these phases.
Org_Id	Number	No	Passes the Org_Id (Operating Unit) of the transaction placed. If the Org_Id is not passed, the pricing engine derives it from the Org context (if it is set), otherwise it uses the default Org_Id for the responsibility.

Note: The pricing engine will set the policy context to Single Org to the Org_Id passed on the control record. If Org_Id is not passed, the pricing engine will derive the Org_Id from the Org context if it is set; otherwise use the default Org_Id for the responsibility. If the Org_Id is not passed to the pricing engine on the control record, and the pricing engine is not able to derive an Org_Id, it will return prices from global price lists and modifiers only. If the Build_Contexts API has been called prior to calling the Price_Request API, the Org Context may have been set by the Build Contexts API.

The pricing event determines the pricing phases to be run during the call to pricing engine. If you create your own pricing phase, associate the pricing events to phases using screen setup _ event phases. Each modifier is associated to one pricing phase. Pricing engine looks only at those modifiers associated with the phases defined for the pricing event passed.

The following are some of the seeded pricing events:

Seeded Pricing Events

Pricing Event	Pricing Phase
PRICE	List Line Base Price
LINE	List Line Base Price
LINE	List Line Adjustment
LINE	Line Charges
LINE	Line Charges - Manual
BOOK	Modifiers for Book event
ORDER	All Lines Adjustments
ORDER	Header Level Adjustments
ORDER	Header Level Charges
SHIP	Line Charges
BATCH	Pricing phases in LINE+ORDER
ICBATCH	Pricing phases in LINE+ORDER. This event is specific to Inventory Intercompany Invoice Transfer Pricing

LINE_REC_TYPE

The following table shows the parameters for this structure:

LINE_REC_TYPE

Parameter	Datatype	Req	Drv	Type	Usage
REQUEST_TYPE_CODE	Varchar2	No	No	In	Identifies the transaction system making the pricing request.
PRICING_EVENT	Varchar2	No	No	N/A	Not used
HEADER_ID	Number	No	No	N/A	Not used
LINE_INDEX	Number	Yes	No	In/Out	PL/SQL unique identifier for request line.
LINE_ID	Number	No	No	In	Unique identifier of the request line in the calling application.
LINE_TYPE_CODE	Varchar2	Yes	No	In	Type of line within the request. Allowable values are: <ul style="list-style-type: none">• ORDER• LINE

Parameter	Datatype	Req	Drv	Type	Usage
PRICING_EFFECTIVE_DATE	Date	Yes	No	In/Out	Date for which the pricing engine calculates the prices.
ACTIVE_DATE_FIRST	Date	No	No	In/Out	In addition to the pricing effective date, you can specify two additional dates for the pricing engine to use to qualify pricing entities. The pricing engine compares this date against the first date range on the modifier list — QP_LIST_HEADERS.B.START_DATE_ACTIVE_FIRST and QP_LIST_HEADERS.B.END_DATE_ACTIVE_FIRST.
ACTIVE_DATE_FIRST_TYPE	Varchar2	No	No	In/Out	The date type of ACTIVE_DATE_FIRST based on lookup type EFFECTIVE_DATE_TYPES.

Parameter	Datatype	Req	Drv	Type	Usage
ACTIVE_DATE_SECOND	Date	No	No	In/Out	In addition to the pricing effective date, you can specify two additional dates for the pricing engine to use to qualify pricing entities. The pricing engine compares this date against the first date range on the modifier list — QP_LIST_HEADERS_B.START_DATE_ACTIVE_SECONDS and QP_LIST_HEADERS_B.END_DATE_ACTIVE_SECONDS.
ACTIVE_DATE_SECOND_TYPE	Varchar2	No	No	In/Out	The date type of ACTIVE_DATE_SECOND based on lookup type EFFECTIVE_DATE_TYPES.
LINE_QUANTITY	Number	Yes	No	In/Out	Pricing request line quantity.

Parameter	Datatype	Req	Drv	Type	Usage
LINE_UOM_CODE	Varchar2	Yes	No	In/Out	Pricing request line unit of measure.
UOM_QUANTITY	Number	No	No	In/Out	Unit of measure quantity, internal value used by engine to perform service pricing.
PRICED_QUANTITY	Number	No	Yes	Out	Quantity of pricing request line that pricing engine has priced.
PRICED_UOM_CODE	Varchar2	No	Yes	Out	Unit of measure in which the pricing engine priced.
CURRENCY_CODE	Varchar2	Yes	No	In/Out	Currency in which the pricing engine priced.
UNIT_PRICE	Number	No	Yes	Out	Unit price of the item that is expressed in <i>Priced UOM Code</i>

Parameter	Datatype	Req	Drv	Type	Usage
PERCENT_P RICE	Number	No	Yes	Out	Price calculated as a percentage of the price of another item.
UPDATED_A DJUSTED_U NIT_PRICE	Number	No	No	In	This is used for overriding the unit selling price of an item to apply a suitable manual adjustment. In this case, the pricing engine will apply a suitable manual adjustment after calculating the unit selling price to match the updated_adjusted_unit_price that the caller has passed in.

Parameter	Datatype	Req	Drv	Type	Usage
ADJUSTED_ UNIT_PRICE	Number	No	Yes	Out	Price per unit after the pricing engine applies discounts and surcharges. It indicates the unit price for the service item, which has the percent price.
PARENT_PRI CE	Number	No	Yes	Out	When the pricing engine determines the price of an item from the price of another item, the price of the related item. This is used only for service items and it is populated from the serviceable item.
PARENT_QU ANTITY	Number	No	Yes	Out	When the pricing engine determines the price of an item from the price of another item, the quantity of the related item.

Parameter	Datatype	Req	Drv	Type	Usage
ROUNDING_FACTOR	Number	No	No	In/Out	<p>The pricing engine will use this factor to which the selling price is rounded to when the pricing engine returns the price.</p> <p>If the caller does not pass this field, the pricing engine will populate the rounding factor from the qualified price list into this field and use it for rounding. If multi-currency is installed and used, the pricing engine will populate the selling rounding factor of the matching currency from the attached multi-currency list (currency conversion criteria) of the qualified price list into this field and</p>

Parameter	Datatype	Req	Drv	Type	Usage
					use it for rounding
PARENT_UO M_CODE	Varchar2	No	Yes	Out	When the pricing engine determines the price of an item from the price of another item, the unit of measure of the related item.
PRICING_PH ASE_ID	Number	No	No		Not used

Parameter	Datatype	Req	Drv	Type	Usage
PRICE_FLAG	Varchar2	Yes	No	In	<p>Indicates the degree to which the price is frozen. Allowable values, based on lookup type</p> <p>CALCULATE_PRICE_FLAG are:</p> <p>Y (Calculate Price): Apply all prices and modifiers to the request line.</p> <p>N (Freeze Price): Do not apply any prices or modifiers to the request line. Consider the volume of the request line when processing LINEGROUP modifiers for other lines.</p> <p>P (Partial Price): Apply prices and modifiers in phases whose freeze override flag is Y.</p>

Parameter	Datatype	Req	Drv	Type	Usage
PROCESSED_CODE	Varchar2	No	No	Out	Internal code which indicates the stage of engine processing when an error occurred.

Parameter	Datatype	Req	Drv	Type	Usage
STATUS_CO DE	Varchar2	No	No	In/Out	<p>Returned status. Allowable values are:</p> <p>N: New record created(All 'N' records are returned back from the pricing engine. These are success records)</p> <p>X: Unchanged(Default status when the line is passed to the pricing engine for processing)</p> <p>U: Updated</p> <p>IPL: Invalid price list(When passed in price list is not found , then an error is given)</p> <p>GSA: GSA violation</p> <p>FER: Error processing formula</p> <p>OER: Other error</p> <p>CALC: Error in calculation engine</p>

Parameter	Datatype	Req	Drv	Type	Usage
					UOM: Failed to price using unit of measure
					INVALID_UOM: Invalid unit of measure
					DUPLICATE_PRICE_LIST : Duplicate price list
					INVALID_UOM_CONV: Unit of measure conversion not found
					INVALID_INCOMP: Could not resolve incompatibility
					INVALID_BEST_PRICE: Could not resolve best price
STATUS_TEXT	Varchar2	No	No	Out	Returned message.

Parameter	Datatype	Req	Drv	Type	Usage
PRICE_REQ UEST_CODE	Varchar2	No	Yes	In/Out	This is unique code to identify each order line consuming a modifier. It is request_type_code concatenated to the header_id and the line_id(applicable only for order lines as line_id is null for header lines) of the order line separated .
HOLD_CODE	Varchar2	No	No	Out	This indicates that a limit having the hold_flag checked has been exceeded or adjusted and is a recommendation to the caller to put the order/line on hold.
HOLD_TEXT	Varchar2	No	No	Out	This is a detailed message corresponding to the hold_code.

Parameter	Datatype	Req	Drv	Type	Usage
USAGE_PRICING_TYPE	Varchar2	No	No	In/Out	<p>Indicates the usage pricing type.</p> <p>REGULAR – Regular pricing</p> <p>AUTHORING – Authoring call</p> <p>BILLING – Billing call</p>
LINE_CATEGORY	Varchar2	Yes	No	In/Out	<p>Indicates the line category. The applicable values:</p> <p>RETURN – Returned line</p> <p>CANCEL – Cancelled line</p> <p>NULL – Regular line</p>

Parameter	Datatype	Req	Drv	Type	Usage
CONTRACT_START_DATE	Date	No	No	In	When UOM conversion is time related and Profile QP: Time UOM Conversion is set to Oracle Contracts, order quantity by order UOM and pricing quantity by pricing UOM are calculated based on contract_start_date, contract_end_date.
CONTRACT_END_DATE	Date	No	No	In	When UOM conversion is time related and Profile QP: Time UOM Conversion is set to Oracle Contracts, order quantity by order UOM and pricing quantity by pricing UOM are calculated based on contract_start_date, contract_end_date.
LINE_UNIT_PRICE	Number	No	Yes	Out	Unit price by order UOM

Parameter	Datatype	Req	Drv	Type	Usage
CHARGE_PE RIODICITY_ CODE	Varchar2	No	No	In	Determines the time period of the charge to be calculated for an order line such as daily, monthly or one time.

Types Of Request Lines

The following describes the types of request lines:

- Regular Order Line where line_type_code = LINE.
- Summary Line where line_type_code = ORDER which is the summary line for the whole Sales Order. This line has all the order level attributes attached to it. This line is required to be passed to the pricing engine to get any Order Level Discounts based on the attributes or qualifiers that are attached to this Line.
- Lines that are generated or derived by the pricing engine , in case of discounts like Promotional Good , where in a new order line is created. This line would have the processed code as ENGINE. The relationship between the new line and the original request line is stored in the Relationship record structure RELATED_LINES_REC_TYPE. So, if the pricing engine returns a free good (PRG Modifier), it generates a new request line.

LINE_DETAIL_REC_TYPE

LINE_DETAIL_REC_TYPE

Parameter	Type	Req	Drv	Type	Usage
LINE_DETAIL_INDEX	Number	Yes	Yes	In/Out	PL/SQL unique identifier. Unique identifier of request line detail in calling application.
LINE_DETAIL_ID	Number	No	No	N/A	Not Used
LINE_DETAIL_TYPE_CODE	Varchar2	Yes	No	Out	Type of detail line.
LINE_INDEX	Number	Yes	No	In/Out	Identifier for parent request line.
LIST_HEADER_ID	Number	Yes	No	In/Out	Identifier of the list header used to create or update the pricing line.
LIST_LINE_ID	Number	Yes	No	In/Out	Identifier of the list line used to create or update the pricing line.

Parameter	Type	Req	Drv	Type	Usage
LIST_LINE_T YPE_CODE	Varchar2	Yes	No	In/Out	Line type of the list line used to update the pricing line. Possible values can be found from the lookup type LIST_LINE_T YPE_CODE from qp_lookups table.
SUBSTITUTI ON_TYPE_ CODE	Varchar2	No	No		Not used.
SUBSTITUTI ON_FROM	Varchar2	No	No		Not used.
SUBSTITUTI ON_TO	Varchar2	No	No	Out	Value for terms substitution attribute.
AUTOMATI C_FLAG	Varchar2	Yes	No	Out	Indicates if the pricing engine should automatically apply the request line detail to the request line. The engine derives the value from the list line.

Parameter	Type	Req	Drv	Type	Usage
OPERAND_ CALCULATI ON_CODE	Varchar2	Yes	No	In/Out	Type of operand. Allowable values are: Adjustment percent(for discounts) Adjustment amount (for discounts) Adjustment NewPrice(for discounts) UNIT_PRICE (for price lists) PERCENT_P RICE(for price lists)
OPERAND_ VALUE	Number	Yes	No	In/Out	Value of pricing request detail line, for example, 10 currency unit list price with 3% discount
PRICING_G ROUP_ SEQUENCE	Number	Yes	No	In/Out	Indicates the pricing bucket in which the pricing engine applied this list line.

Parameter	Type	Req	Drv	Type	Usage
PRICE_BREAK_TYPE_CODE	Varchar2	Yes	No	In/Out	Type of price break based on lookup type PRICE_BREAK_TYPE_CODE. Possible Values: POINT , RANGE , RECURRING
CREATED_FROM_LIST_TYPE_CODE	Varchar2	Yes	No	In/Out	List type used to create or update the pricing line. Possible values can be found from the lookup_type LIST_TYPE_CODE from qp_lookups table.
PRICING_PHASE_ID	Number	Yes	No	In/Out	The pricing phase which created the request line detail.
LIST_PRICE	Number	No	No	Out	Not Used

Parameter	Type	Req	Drv	Type	Usage
LINE_QUANTITY	Number	No	Yes	Out	Quantity on the price break line. Populated if the pricing engine derived the value of the request line or request line detail from a price break. A not null value indicates that this particular break line was used in the calculation.
ADJUSTMENT_AMOUNT	Number	No	No	Out	It holds the value of the bucketed adjusted amount for line types like PLL , DIS ,SUR etc. For price break(PBH) child lines , the field is populated if the pricing engine derived the value of the request line or request line detail from a price break.

Parameter	Type	Req	Drv	Type	Usage
APPLIED_FL AG	Varchar2	No	No	In/Out	<p>The lists or list lines that this pricing event or a prior pricing event applied. Allowable values are</p> <p>Yes: Applicable when the attribute context is a list or list line</p> <p>No: Applicable when the attribute context is a list or list line</p> <p>Null</p>
MODIFIER_L EVEL_ CODE	Varchar2	Yes	No	In/Out	<p>The level at which the list line qualified for the transaction. Based on lookup type MODIFIER_L EVEL_CODE.</p>

Parameter	Type	Req	Drv	Type	Usage
STATUS_CO DE	Varchar2	No	No	Out	<p>Returned status. Possible Values:</p> <p>N: New record created(All 'N' records are returned back from the pricing engine.These are success records)</p> <p>UPDATED: Indicates Success status and that the record is Updated record and is not new.</p> <p>X: Indicates Success status and that the record is Unchanged</p> <p>BACK_CALCULATION_ERROR : Indicates that there were no qualified manual overrideable adjustments to adjust the price.</p> <p>D : Deleted. This indicates Failure.</p> <p>D_PBH :</p>

Parameter	Type	Req	Drv	Type	Usage
					deleted in PBH processing
STATUS_TXT	Varchar2	No	No	Out	Returned message.
SUBSTITUTION_ATTRIBUTE	Varchar2	No	No	Out	Modifier details. The attribute in the TERMS context that the pricing engine substituted, for example, Payment Terms. Used for Term Substitution-type modifiers.
ACCRUAL_FLAG	Varchar2	No	No	In/Out	Indicates that the modifier is an accrual
LIST_LINE_NO	Varchar2	No	No	In/Out	Modifier number. This field is applicable in case of Coupon Issue kind of modifier line.

Parameter	Type	Req	Drv	Type	Usage
ESTIM_GL_VALUE	Number	No	No	Out	The discount or surcharge value of the modifier. Used to estimate the discount cost for non-monetary modifiers.
ACCRUAL_CONVERSION_RATE	Number	No	No	Out	The rate to use when converting a non-monetary accrual to a monetary value.
OVERRIDE_FLAG	Varchar2	No	No	In/Out	Indicates if a user in the calling application can override the modifier value.
PRINT_ON_INVOICE_FLAG	Varchar2	No	No	Out	Not used.
INVENTORY_ITEM_ID	Number	No	No	Out	Inventory item identifier in an item relationship. Used for list line type Item Upgrade.

Parameter	Type	Req	Drv	Type	Usage
ORGANIZATION_ID	Number	No	No	Out	Organization identifier in an item relationship. Used for list line type Item Upgrade.
RELATED_ITEM_ID	Number	No	No	Out	Related inventory item identifier in an item relationship. Used for list line type Item Upgrade.
RELATIONSHIP_TYPE_ID	Number	No	No	Out	Relationship type identifier in an item relationship. Used for list line type Item Upgrade.

Parameter	Type	Req	Drv	Type	Usage
ESTIM_ACCRUAL_RATE	Number	No	No	Out	Indicates the percentage at which to accrue or, for a coupon, the expected rate of redemption of the coupon. Liability is defined as: ACCRUAL OR COUPON VALUE * ESTIM_ACCRUAL_RATE. Default Value: 100
EXPIRATION_DATE	Date	No	No	Out	The expiration date of the accrual or coupon.
BENEFIT_PRICE_LIST_LINE_ID	Number	No	No	Out	The price list_line_id which has the list price before promotional discount. Used for Promotional Goods-type modifiers when the pricing engine creates a new transaction line.

Parameter	Type	Req	Drv	Type	Usage
RECURRING_FLAG	Varchar2	No	No		Not Used
BENEFIT_LIMIT	Number	No	No	Out	Not Used.
CHARGE_TYPE_CODE	Varchar2	No	No	In/Out	Indicates the type of charge based on lookup type FREIGHT_CHARGES_TYPE. Used for Freight/Special Charge-type modifiers.
CHARGE_SUBTYPE_CODE	Varchar2	No	No	In/Out	Indicates the type of charge based on lookup type CHARGE_TYPE_CODE.
INCLUDE_ON_RETURNS_FLAG	Varchar2	No	No	Out	Indicates whether the pricing engine should include the charge on a return transaction. Used for Freight/Special Charge-type modifiers.

Parameter	Type	Req	Drv	Type	Usage
BENEFIT_QTY	Number	No	No	Out	The accrual quantity for non-monetary accruals or, for promotional goods, item quantity.
BENEFIT_UNIT_CODE	Varchar2	No	No	Out	The accrual unit of measure for non-monetary accruals, or, for promotional goods, item unit of measure.
PRORATION_TYPE_CODE	Varchar2	No	No	Out	Not Used.
SOURCE_SYSTEM_CODE	Varchar2	No	No	Out	Not Used.
REBATE_TRANSACTION_TYPE_CODE	Varchar2	No	No	Out	Not Used.
SECONDARY_PRICE_LIST_IND	Varchar2	No	No	Out	Indicates that the pricing used a secondary price list instead of the price list that the calling application requested.

Parameter	Type	Req	Drv	Type	Usage
GROUP_VALLUE	Varchar2	No	No	Out	This is populated for modifiers with modifier level code 'Group of Lines' and arithmetic operator as 'LUMPSUM'. It indicates the quantity/amount applied for that order line to which this belongs to. It is the quantity if the pricing attribute has 'Item Quantity' and amount if the pricing attribute is 'Item Amount'.
COMMENTS	Varchar2	No	No	Out	The comments on a modifier
UPDATED_FLAG	Varchar2	No	No	In/Out	Indicates that this modifier has been overridden.
PROCESS_CODE	Varchar2	No	No	Out	Indicates the pricing engine processing code for internal use.

Parameter	Type	Req	Drv	Type	Usage
LIMIT_CODE	Varchar2	No	No	Out	<p>Indicates that the limit on this modifier has either exceeded or adjusted or consumed. Applicable values:</p> <ul style="list-style-type: none"> EXCEEDED ADJUSTED CONSUMED
LIMIT_TEXT	Varchar2	No	No	Out	The detailed message corresponding to the limit_code
FORMULA_ID	Varchar2	No	No	Out	The formula_id setup on the modifier.

Parameter	Type	Req	Drv	Type	Usage
CALCULATION_CODE	Varchar2	No	No	Out	Indicates the calculation code on the adjustment. BACK_CALCULATE : Indicates that this adjustment is a back calculated adjustment and it got applied to adjust the overridden selling price. This is related to the field updated_adjusted_unit_price on line_rec_type.
ROUNDING_FACTOR	Number	No	No	Out	Conversion rounding factor, populated when multi-currency set-up is being used
CURRENCY_DETAIL_ID	Number	No	No	Out	Unique identifier for the multi-currency detail record, populated when multi-currency set-up is being used

Parameter	Type	Req	Drv	Type	Usage
CURRENCY_HEADER_ID	Number	No	No	Out	Unique identifier for the multi-currency header record, populated when multi-currency set-up is being used
SELLING_ROUNDING_FACTOR	Number	No	No	Out	Rounding Factor, populated when multi-currency set-up is being used
ORDER_CURRENCY	Varchar2	No	No	Out	Order Currency, populated when multi-currency set-up is being used
PRICING_EFFECTIVE_DATE	Date	No	No	Out	Pricing effective date, populated when multi-currency set-up is being used
BASE_CURRENCY_CODE	Varchar2	No	No	Out	Base currency code, populated when multi-currency set-up is being used

Parameter	Type	Req	Drv	Type	Usage
Break_Uom_ code	Varchar2	No	No	Out	Unit of Measure code for the price break.
Break_Uom_ Context	Varchar2	No	No	Out	Context to define the seeded context for break UOM.
Break_Uom_ Attribute	Varchar2	No	No	Out	Attribute in which the calling application is supposed to pass the break UOM of volume attribute used for break.

Inserting Lines into Temporary Tables: The pricing engine takes in the input data, processes and inserts to the temporary table. By inserting the request lines, passed in attributes, qualifiers and modifiers directly to the temporary tables, pricing engine performance is improved. The caller can indicate to the pricing engine that the data is inserted to the temporary tables by passing a value of "No" to the TEMP_TABLE_INSERT_FLAG on the control record. The following procedures enable bulk inserts to the pricing temporary tables:

- The procedure INSERT_LINES2 in the pricing engine API QP_PREQ_GRP inserts the request lines to the temporary table. This procedure does a bulk insert to the temporary tables, so you need to load the request line details PL/SQL table of records to individual PL/SQL tables one for each column. The temporary table for request lines QP_PREQ_LINES_TMP has columns that map to the columns of the P_LINE_TBL PL/SQL table.
- The procedure INSERT_LINE_ATTRS2 in the API QP_PREQ_GRP inserts the passed in attributes. This also does a bulk insert. Each column of the P_LINE_ATTR_TBL maps to a column in the temporary table for line attributes QP_PREQ_LINE_ATTRS_TMP .
- The manual modifiers that need to be applied can be inserted into the temporary table for modifiers QP_PREQ_LDETS_TMP.

- The build sourcing API has been modified to insert into the temporary tables directly. The procedure BUILD_CONTEXTS in the QP_ATTR_MAPPING_PUB API populates the temporary tables directly. This can be called to source the necessary attributes.
- The procedure to make the pricing engine call, PRICE_REQUEST in the pricing engine API QP_PREQ_PUB, has been overloaded to pass only the control record.
- The output of the pricing engine exists in the temporary tables and the caller can process the data from the pricing temporary tables.

This is the process to insert records into the temporary tables and this will enhance the performance of the pricing engine. There is a example script to demonstrate the direct insertion. Please refer to the example scripts mentioned in this document.

Types of Request Line Details

- Price List Line (PLL): OPERAND_VALUE and OPERAND_CALCULATION_CODE columns store the price information and the type of price(UNIT_PRICE or PERCENT_PRICE).
- Discount Line (DIS): OPERAND_VALUE and OPERAND_CALCULATION_CODE columns store the price information and the type of price(AMT,%,or NEW_PRICE).
- Surcharge Line (SUR): OPERAND_VALUE and OPERAND_CALCULATION_CODE columns store the price information and the type of price(AMT,%,or NEW_PRICE).
- Other Item Discount (OID) processing: For OID processing, two request lines need to be passed to the pricing engine; for example, Buy A get 10% of B.

In the above example A and B need to be ordered on two order lines, so two request lines are passed to the pricing engine. When the engine does the OID processing, it creates a discount line for 10% on the 2nd request line, which applies the 10% discount to the 2nd request line. Also a relationship record is created in the RELATED_LINES_REC_TYPE record structure. Line Detail Index 8 is the actual Discount Line which is the OID line and Line Detail Index 9 is the actual benefit line which is 10% off line. So, the relationship is as follows:

Relationship

Line Index	Line Detail Index	Related Line Index	Related Line Detail Index	Relation Ship Type
1	8(OID Line)	2	9(10% line)	GENERATED_LINE

- Promotional Goods Discount (PRG)

PRG Processing: For example, Buy A get B for free, only Buy Item A request line needs to be passed to the pricing engine.

In the above example only request line/order line with Item A is sent to the pricing engine. Item B need not be ordered . This is the basic difference between OID and PRG. The pricing engine selects the PRG Modifier because of purchase of Item A and creates a record in the LINE_DETAIL_REC_TYPE record structure (Line Index 1-Line Detail Index 1). Then it tries to give the benefit which is a free B Item. In the process, it does the following:

1. A new request line is created in the LINE_REC_TYPE record structure.(Line Index 2).
 2. A new relationship between the Item A line and Item B line is created ,in the RELATED_LINES_REC_TYPE record structure. It is a Line-Line relationship(Line Index 1 - Line Index 2).
 3. A new Price List Line is created for the new request line in the LINE_DETAIL_REC_TYPE record structure.(Line Index 2 - Line Detail Index 3).
 4. A new adjustment line for 100% discount is created for the new request line in the LINE_DETAIL_REC_TYPE record structure.(Line Index 2 - Line Detail Index 4).
 5. A new relationship line between the original PRG Line Detail Line and the new 100% off line detail is created in the RELATED_LINES_REC_TYPE record structure .(Line Detail Index 1 - Line Detail Index 4).
 6. A new record is created in the LINE_ATTR_REC_TYPE record structure for the new item B. (Line Index 2 - PRICING_CONTEXT = 'ITEM' , PRICING_ATTRIBUTE = 'PRICING_ATTRIBUTE1', PRICING_ATTR_VALUE = 'Item B').
- Terms Substitution (TSN)

- SUBSTITUTION ATTRIBUTE
- SUBSTITUTION TO

- Freight Charge

Freight Charges Processing: All the freight charge discounts do not affect the selling price. The pricing engine calculates the adjustment amount for the freight charge discounts but, this does not effect the selling price.

- Manual Modifiers Processing

All the automatic modifiers of type Discounts and Surcharges that the user has qualified for, that are deleted as part of incompatibility resolution (due to incompatibility setup rules), are returned as manual discounts. In addition to these discounts, all the qualified manual modifiers of type Discounts and Surcharge discounts are also returned to the caller, unapplied.

Manual adjustments can be applied in two ways:

- The caller can pass the manual adjustment to the pricing engine with `Applied_Flag = Y` and `Updated_Flag = Y`. The engine will apply this manual adjustment. The manual adjustment can be overridden by passing the new operand on the `line_detail_rec_type.adjustment_amount`.

Example: A pricing call is made for three order lines with a manual adjustment of 10% to be applied to the second order line. The manual adjustment should be passed in the `line_detail_tbl` with columns `updated_flag = Y` and `applied_flag = Y` and with the `line_index` of the second order line. The pricing engine API, will calculate the adjustment amount and will apply this manual adjustment to the second order line. The `applied_flag` and `updated_flag` will be returned as Y to indicate that it has been applied.

- The selling price can be overridden by passing the new selling price in the `line_detail_rec_type.updated_adjusted_unit_price`. The engine will then pick up a suitable manual overrideable modifier that is qualified and back calculate the adjustment amount and operand to match the new selling price. In this case the pricing engine will pass back this manual modifier with `calculation_code` as `BACK_CALCULATE`, `updated_flag = Y` and `applied_flag = Y`.

Example: Three order lines are passed to the pricing engine: the unit selling price on the second order line is \$80 and the unit list price is \$100 and you want to override the selling price to \$90. In this case, you must pass 80 in the column `UPDATE_ADJUSTED_UNIT_PRICE` on the request line in the record corresponding to the second order line in the `LINE_TBL`.

Then the pricing engine selects all the qualifying manual overrideable adjustments, decides whether it needs to apply a discount or a surcharge. It prefers a manual overrideable adjustment that has been applied already. If

none has been applied already, it randomly picks up a manual adjustment, back calculates the adjustment amount (\$10 surcharge in this case), and returns it with calculation_code as BACK_CALCULATE.. If there are no qualified manual overrideable adjustments, an error status is returned on the second order line indicating that there are no manual adjustments. If an error occurs during the back calculation, the pricing engine returns an error status on the second order line. The pricing_status_code on the second order line has the error code BACK_CALCULATION_ERROR.

- Coupon Issue (CIE)

Coupon processing: For example, Buy Item A , get a Coupon for 10% of Item B.

The pricing engine gets one request line with Item A. Based on the search , the engine loads the Coupon Issue Modifier Line (CIE) in the LINE_DETAIL_REC_TYPE record structure. When processing the coupon issue, the pricing engine generates a coupon number which can be found from the LIST_LINE_NO column of the LINE_DETAIL_REC_TYPE record structure on the CIE modifier line. Also, the pricing engine creates a qualifier with this generated list_line_no for the 10% discount on Item B. That way, this discount will not be given unless this coupon number is presented the next time as a qualifier to the pricing engine.

In the above example, suppose C10B is generated as the coupon number. The next time the customer gets this Coupon and this coupon number is punched in before making a pricing engine call, then the 10% discount on Item B is given if the Customer Buys Item B.

- Price Break Header (PBH)

Price Break Processing: The price break processing is the same for both Price List Breaks and Discount/Surcharge Breaks. Price Breaks can be either POINT or RANGE breaks. This is found in the PRICE_BREAK_TYPE_CODE column in the record structure. The following is an example of how a Price Break is set up:

Price Break setup has a Price Break Parent Record which has a line type called PBH. This PBH record can have more than 1 child record that actually define the breaks. The parent and child records are contained in the LINE_DETAIL_REC_TYPE record structure.

Example

List Line Type Code	Parent/Child	Price Break From	Price Break To	Value	Break Type
PBH	Parent	--	--	--	POINT/RANGE
PLL	Child	0	100	\$100	POINT/RANGE
PLL	Child	100	200	\$90	POINT/RANGE
PLL	Child	200	300	\$80	POINT/RANGE
PLL	Child	300	--	\$70	POINT/RANGE

The Adjustment Amount and Line Quantity Fields exactly tell how and which child line has been used in deriving the break . For example, if the ordered Quantity is 350, then the record structure will have the following information:

Record Structure Information

List Line Type Code	Adjustment Amount	Line Quantity	Break Type
PBH	\$90	350	POINT
PLL	\$100	0	POINT
PLL	\$90	0	POINT
PLL	\$80	0	POINT
PLL	\$70	350	POINT
PBH	\$87.1428571	350	RANGE
PLL	\$100	100	RANGE

List Line Type Code	Adjustment Amount	Line Quantity	Break Type
PLL	\$90	100	RANGE
PLL	\$80	100	RANGE
PLL	\$70	50	RANGE

The relationship between the Parent and Child Records are stored in the RELATED_LINES_REC_TYPE record structure. The relationship type is PBH_LINE.

RELATED_LINES_REC_TYPE

Line Index	Line Detail Index	Related Line Index	Related Line Detail Index	Relation Ship Type
1	8(PBH Line Detail)	1	9(PLL Line Detail 1)	PBH_LINE
1	8	1	10(PLL Line Detail 2)	PBH_LINE
1	8	1	11(PLL Line Detail 3)	PBH_LINE
1	8	1	12(PLL Line Detail 4)	PBH_LINE

- Item Upgrade (IUE)

Important Columns:

- INVENTORY_ITEM_ID
- RELATED_ITEM_ID
- QUAL_REC_TYPE

The Required and Derived values for the following parameters are Null.

QUAL_REC_TYPE

Parameter	DataType	Type	Usage
LINE_INDEX	Number	In/Out	Unique identifier for request line or request line detail.
QUALIFIER_CONTEXT	Varchar2	In/Out	Context for qualifier, for example, Customer Hierarchy.
QUALIFIER_ATTRIBUTE	Varchar2	In/Out	Qualifier attribute, for example, QUALIFIER_ATTRIBUTE1: Customer Class.
QUALIFIER_ATTRIBUTE_VALUE_FROM	Varchar2	In/Out	Value for qualifier attribute
QUALIFIER_ATTRIBUTE_VALUE_TO	Varchar2	In/Out	Return value for qualifier attribute. Populated when the pricing engine returns details of a volume break.
COMPARISON_OPERATOR_CODE	Varchar2	Out	The pricing engine creates qualifier attributes to indicate to the calling application which qualifier attribute caused it to give a benefit, for example, Order Amount > 1000 currency units (> is the operator code).

Parameter	DataType	Type	Usage
VALIDATED_FLAG	Varchar2	In/Out	Indicates that a price list or modifier list(asked for promotion) is valid for the pricing request. Applicable to price list and modifier list qualifiers; the pricing engine assumes that other qualifiers are valid.
STATUS_CODE	Varchar2	In/Out	Return status.
STATUS_TEXT	Varchar2	Out	Return message

LINE_ATTR_REC_TYPE

The Required and Derived values for the following parameters are Null.

LINE_ATTR_REC_TYPE

Parameter	Datatype	Type	Usage
LINE_INDEX	Number	In/Out	Unique identifier for request line or request line detail.
PRICING_CONTEXT	Varchar2	In/Out	Context for a product or pricing attribute, for example, Product Hierarchy.
PRICING_ATTRIBUTE	Varchar2	In/Out	Product or pricing attribute, for example, PRICING_ATTRIBUTE11: Customer Item ID.

Parameter	Datatype	Type	Usage
PRICING_ATTR_VALUE_FROM	Varchar2	In/Out	Value for product or pricing attribute.
PRICING_ATTR_VALUE_TO	Varchar2	In/Out	Return value for Pricing attribute. Populated when the pricing engine returns details of a volume break
VALIDATED_FLAG	Varchar2	In/Out	Not used.
STATUS_CODE	Varchar2	In/Out	Return status
STATUS_TEXT	Varchar2	In/Out	Return Message

RELATED_LINES_REC_TYPE

The Required and Derived values for the following parameters are Null.

RELATED_LINES_REC_TYPE

Parameter	Datatype	Type	Usage
LINE_INDEX	Number	In/Out	PL/SQL unique identifier for request line.
LINE_DETAIL_INDEX	Number	In/Out	PL/SQL unique identifier for request detail line
RELATIONSHIP_TYPE_CODE	Varchar2	In/Out	Type of relationship between pricing lines.
RLTD_LINE_INDEX	Number	In/Out	PL/SQL identifier for related request line.

Parameter	Datatype	Type	Usage
RLTD_LINE_DETAIL_INDEX	Number	In/Out	PL/SQL unique identifier for related request detail line.

Validation of Price Request API

Validations:

Standard Validation

Oracle Pricing validates all required columns in the Price Request API. For specific information on the data implied by these columns, see: *Oracle Pricing Technical Reference Manual* for details.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Price Request API processes the rows and reports the following values for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	Null
Failure	4	actual error message

Example of Price Request Application Program Interface

The following script names are used in the examples for the Price Request Application Program Interface:

- qp_engine_testing.sql (All regular Price Lists, Discounts , Surcharges etc.)
- qp_test_service.sql (Service Item Pricing Setup)
- qp_test_oid.sql (Other Item Discount Setup)
- qp_direct_insert.sql (Direct insertion into temporary tables to improve performance)

- qp_manual_adjustments (Apply manual adjustments)
- qp_override_selling_price.sql (Override the unit selling price)

These scripts help you set up data and make a call to the pricing engine to return the price and any related discounts. The scripts are located in the following area:
\$QP_TOP/patch/115/sql.

The setup involves passing information to the pricing engine using the following:

1. **Control Record Information:** The Control Record has parameters which control the behavior of the pricing engine. Please refer to the example script for more details.
2. **Request/Order Line Information:** Request Line has information that contains the elements in the calling application that require a base and adjusted price. It may equate to a transaction line or transaction header record in the calling application. Refer to qp_engine_testing.sql for information on setting up the request line data. LINE_REC_TYPE is the record structure that needs to be completed when passing information related to a request line.
3. **Qualifiers and Pricing Attributes Information:** Qualifiers and pricing attributes information that helps the pricing engine to determine the price list lines and modifier list lines for which a price request is eligible.

Pricing Attributes and Qualifier Attributes Setup

LINE_ATTR_REC_TYPE is the record structure that needs to be filled in when passing information related to a pricing attribute. The following are the important columns:

- PRICING_CONTEXT
- PRICING_ATTRIBUTE
- PRICING_ATTR_VALUE_FROM

QUAL_REC_TYPE is the record structure that needs to be filled in when passing information related to a qualifiers. The following are the important columns:

- QUALIFIER_CONTEXT
- QUALIFIER_ATTRIBUTE
- QUALIFIER_ATTR_VALUE_FROM

Any attribute for example Item, Price List Id, Customer Name, Order Type etc. will have a mapping to give the Context, Attribute, Value_Id (will be the actual value). For more information on seeded attributes, see: *Oracle Advanced Pricing User's Guide*. Please refer to the following examples:

Pricing Attributes and Qualifier Attributes

Attribute	Attribute Name	Context	Attribute	Value_Id (id)	Value (Name)
Pricing Attribute	Inventory Item Id	ITEM	PRICING_ATTRIBUTE1	149	AS54888
Pricing Attribute	Plastics	ITEM	PRICING_ATTRIBUTE2	29	PLASTIC
Qualifier	Order Type	ORDER	QUALIFIER_ATTRIBUTE1	1000	Standard
Qualifier	Customer	CUSTOMER	QUALIFIER_ATTRIBUTE1	100	Business World
Qualifier	Price List	MODLIST	QUALIFIER_ATTRIBUTE4	1000	Corporate
Qualifier	Agreement Type	ORDER	QUALIFIER_ATTRIBUTE9	RESELLER	RESELLER
Qualifier	Order Date	ORDER	QUALIFIER_ATTRIBUTE5	2000/12/10 00:00:00	

Note: Dates and Numbers are to be passed in canonical format
Canonical Date Format: YYYY/MM/DD HH24:MM:SS Ex: select fnd_date.date_to_canonical ('01-OCT-2000') from dual; Canonical Number Format, '.' is the decimal separator. For example, select fnd_number.number_to_canonical(2000.45) from dual.

Interpreting the results from the pricing engine

- Request Line Information: The output request line record would have the unit_price and adjusted unit price information. Please refer to the example to see more details.
- Price List/Discount Related Information: The output request line detail record would have the price list line and modifier list line detail information like the unit_price value, discount percentage etc. The columns of the output record structure are described further in this document.
- Qualifiers and Pricing Attributes: The output qualifier and pricing attribute records

would have the qualifiers and pricing attributes which qualified the price list line or modifier list line. The columns of the output record structure are described further in this document.

- **Related Lines Information:** This record structure has information related to Service Lines Relationship , Price Break Lines Relationship, OID Modifier Line relationship, PRG Modifier Line Relationship and Order To Line Relationship (Type - ORDER_TO_LINE) which tells an Order Level Discount was applied to which Lines.

Pricing Data Bulk Loader API

You can use the Pricing Data Bulk Loader API to load pricing data from interface tables to the core pricing tables. Before using this feature, ensure that the interface tables contain the data to be uploaded. When you run the concurrent program QP: Bulk Import of Price List to upload the data, the Pricing Data Bulk Loader API uploads the pricing data into the pricing core tables.

Note: Currently, only the price list data is uploaded by the API.

Functional Overview

Using the Pricing Data Bulk Loader API, you can:

- Use the same interface program and tables for updating the pricing data.
- Delete or update pricing attributes, qualifiers, breaks, and so on, using the interface tables.
- Store the interface program id in the main pricing tables so that user knows which request id has updated the records.
- Optionally, load list lines even if validation of pricing attributes fails.
- Mark the record as being processed so that multiple processes do not pick up the same record.
- Process a specific set rather than the entire eligible data.
- Provide all validation error messages in the same run for unloaded records instead of returning errors one after another.

The pricing data is retrieved from four interface tables to populate the core pricing tables. The data model and validations should be considered while populating the interface tables.

Input Tables

- **QP_INTERFACE_LIST_HEADERS:** This table should be populated with the header information of the price lists to be uploaded.
- **QP_INTERFACE_LIST_LINES:** This table should be populated with the line information of the price lists to be uploaded.
- **QP_INTERFACE_PRICING_ATTRIBS:** This table should contain the Product and Pricing attribute details of the price list line, which are to be imported.
- **QP_INTERFACE_QUALIFIERS:** This table should contain the header qualifiers associated with the price lists which are to be imported.

Output table

- **QP_INTERFACE_ERRORS:** This table stores errors encountered during the bulk import process. The QP core tables are also populated during the bulk import process.

Note: For more information, see: Oracle eTRM (eTechnical Reference Manual) available on [OracleMetaLink](#).

Setting Up and Parameter Descriptions

The following parameters can be passed to the concurrent program and API:

- **Entity:** Indicates the pricing entity (for example, price list) that can be uploaded. Entity type for price lists is PRL.
- **Entity Name:** Specifies the name of the pricing entity to be processed, for example, price list name is Corporate. A null value means that all the available entities of that Entity type in the interface tables should be processed.
- **Process ID:** This corresponds to the process_id in the interface tables. The Process ID groups data in the interface table and can be specified by the user as the criteria to pick up records for processing. A null value for this parameter means all available records for processing should be considered subject to restrictions imposed by the value of parameter p_entity_name.
- **No of Threads:** Indicates the total number of child processes or threads to achieve multithreading for list lines and their child records. Default is 1.
- **Turn Debug On:** Valid values are Yes or No. If Yes, detailed debug messages are written to the concurrent program log file. As debug messages add to the performance overhead, selecting No results in quicker completion of the concurrent request.

- **Enable Duplicate Check:** Has two valid values:
 - Yes: Does a duplicate check on the entered lines and gives an error if a duplicate line is found.
 - No: Does not do a duplicate check on the entered lines and, therefore, gives better performance.

This concurrent program calls the procedure QP_Bulk_Loader_PUB.Load_Pricing_Data, which has following parameters:

Parameter	Usage	Type	Req	Drv
Err_buff	Out	Varchar2	No	No
Retcode	Out	Number	No	No
P_entity	In	Varchar2	No	No
P_entity_name	In	Varchar2	No	No
P_process_id	In	Number	No	No
P_process_type	In	Varchar2	No	No
P_no_of_threads	In	Number	No	No
P_spawned_request	In	Varchar2	No	No
P_request_id	In	Number	No	No
P_Debug_On	In	Varchar2	No	

Validation of Pricing Data Bulk Loader

Standard Validation

Some standard validations are listed below.

- UNIT price break has no BLOCK price break child lines.
- If list line type is Price List Line then values for operand, price by formula id, generate using formula id, cannot all be null.

- Recurring value is not null--when a recurring value is present, the value should be at least 1. For example, recurring value can be present when price_break_type_code is RANGE and arithmetic_operator is BLOCK_PRICE.
- If both start date and end date are filled, then start date should be before the end date.
- If list line type is Price Break Header then price break type code and RLTD_MODIFIER_GRP_CODE cannot be null.
- If block pricing and price break type code is Point then there should not be any child lines with unit price or recurring value filled.
- If the price list line and price break header is not NULL then both line and child line should have unit price or line and the child line should have block pricing. For break unit price, the parent line should be block price and break type code should be range.

All the standard price list validations are done. Some fields like SHIP_METHOD_CODE, FREIGHT_TERMS_CODE, TERMS_ID, and CURRENCY_HEADER_ID in the QP_INTERFACE_LIST_HEADERS table and PRICE_BY_FORMULA, GENERATE_USING_FORMULA fields in the QP_INTERFACE_LIST_LINES are populated with the internal code value. To ease this, one extra field for each field is provided where the description/name of the code can be populated and the code field can be left empty during the loading of the interface tables. During the loading process the appropriate code would be fetched and populated in the live tables. For more information, see: Oracle eTRM (*eTechnical Reference Manual*) available on [OracleMetaLink](#).

Other Validation

The following specific user interface (UI) validations are also handled:

- The Pricing attribute context for a PBH (price break header) child line must be VOLUME.
- For each line record inserted, there must be one pricing attribute record, with null pricing context/attribute/value.
- Application method of the PBH child line must be compatible with application method of parent PBH line.
- All the pricing attribute records of the same parent price list line must have same product context/attribute/value.

Result

Once the concurrent program runs, the following attributes in the table can be checked to find the status.

- **Process_Status_Flag:** The process status of the record. The following table outlines

the status values and their meaning:

Status Value	Meaning
P	Ready to be picked.
I	Successfully interfaced.
NULL	The record encountered errors during validation.

- **Request_Id:** Concurrent request_id which picked this record for processing.

The records in the interface tables which are successfully copied to the live tables get deleted from the interface tables. So if a record is not in the interface table after the program executes, then it indicates that the record has migrated successfully to the live tables.

Error Handling

If any validation failure occurs for the record, the process_status_flag value is set to NULL and the corresponding error message is inserted in the qp_interface_errors table along with request_id and orig_sys_reference. The error messages are also sent to the output log (when concurrent program is run).

Correct any errors in the interface tables in the QP_INTERFACE_ERROR and then reset PROCESS_STATUS_FLAG value to "P." The REQUEST_ID should also to be reset to NULL to resubmit the records for next upload. (Alternatively, you can give the Name of the price list that failed as an input so that it is not necessary to reset the values.)

Periodically, the QP_INTERFACE_ERRORS table needs to be purged of error messages from prior requests to maintain effective performance. If a child record needs to be corrected and resubmitted when the parent (and grandparent if applicable) record was successfully interfaced and deleted by the previous run of the program, you need to repopulate the parent (and grandparent if applicable) record with a process_status_flag= I in the interface table.

Examples of Pricing Data Bulk Loader API

The following table contains some example data which can be uploaded. Note that only the mandatory fields are listed.

The following script examples of the price list bulk loader feature are available from the \$QP_TOP/patch/115/sql directory:

- QPBLKEX1.sql: Script to insert price list header and price list line.
- QPBLKEX2.sql: Script to insert price list header and price list line and pricing

attributes.

- QPBLKEX3.sql: Script to insert price list header and price break line.
- QPBLKEX4.sql: Script to update price list header with qualifiers.
- QPBLKEX5.sql: Script to attach secondary price lists to the primary price list.

Example: Inserting a price list

QP_INTERFACE_LIST_HEADERS

Parameter	Value
LIST_TYPE_CODE	PRL
name	CATALOG_PRICE_LIST1
description	Catalog Price List 1
currency_code	USD
active_flag	Y
process_flag	Y
orig_sys_header_ref	CPL00001
Interface_action_code	INSERT
Language	US
source_lang	US
rounding_factor	-2
PROCESS_STATUS_FLAG	P

Example: Inserting the preceding price list as a secondary price list to the Corporate price list

QP_INTERFACE_QUALIFIERS

Parameter	Value
ORIG_SYS_QUALIFIER_REF	OSQR0001
QUALIFIER_GROUPING_NO	-1
QUALIFIER_CONTEXT	MODLIST
QUALIFIER_ATTRIBUTE_CODE	PRICE_LIST
QUALIFIER_ATTR_VAL UE_CODE	Corporate
ORIG_SYS_HEADER_REF	CPL00001
PROCESS_FLAG	Y
INTERFACE_ACTION_CODE	INSERT
COMPARISON_OPERATOR_CODE	=
QUALIFIER_PRECEDENCE	10
PROCESS_STATUS_FLAG	P

Example: Inserting a price list line

QP_INTERFACE_LIST_LINES

Parameter	Value
ORIG_SYS_LINE_REF	OSLR0001

Parameter	Value
orig_sys_header_ref	CPL00001
list_line_type_code	PLL
arithmetic_operator	UNIT_PRICE
OPERAND	999
PRIMARY_UOM_FLAG	Y
PROCESS_FLAG	Y
INTERFACE_ACTION_C ODE	INSERT
PROCESS_STATUS_FLAG	P

QP_INTERFACE_PRICING_ATTRIBS

Parameter	Value
ORIG_SYS_PRICING_AT TR_REF	OSPAR0001
ORIG_SYS_LINE_REF	OSLR0001
ORIG_SYS_HEADER_REF	CPL00001
PRODUCT_ATTRIBUTE_ CONTEXT	ITEM
PRODUCT_ATTR_CODE	INVENTORY_ITEM_ID
PRODUCT_ATTR_VAL_D ISP	AS54888
PROCESS_FLAG	Y

Parameter	Value
INTERFACE_ACTION_C ODE	INSERT
PRODUCT_UOM_CODE	Ea
PROCESS_STATUS_FLAG	P

Example: Inserting a price break header and line

QP_INTERFACE_LIST_LINES

Parameter	Value
ORIG_SYS_LINE_REF	OSLR0002
orig_sys_header_ref	CPL00001
list_line_type_code	PBH
arithmetic_operator	POINT
OPERAND	UNIT_PRICE
PROCESS_FLAG	Y
INTERFACE_ACTION_C ODE	INSERT
PROCESS_STATUS_FLAG	P

QP_INTERFACE_PRICING_ATTRIBS

Parameter	Value
ORIG_SYS_PRICING_ATT R_REF	OSPAR0002
ORIG_SYS_LINE_REF	OSLR0002

Parameter	Value
ORIG_SYS_HEADER_REF	CPL00001
PRODUCT_ATTRIBUTE_C ONTEXT	ITEM
PRODUCT_ATTR_CODE	INVENTORY_ITEM_ID
PRODUCT_ATTR_VAL_DI SP	CM13139
PROCESS_FLAG	Y
INTERFACE_ACTION_CO DE	INSERT
PRODUCT_UOM_CODE	Ea
PROCESS_STATUS_FLAG	P

QP_INTERFACE_LIST_LINES

Parameter	Value
ORIG_SYS_LINE_REF	OSLR0003
orig_sys_header_ref	CPL00001
list_line_type_code	PLL
arithmetic_operator	UNIT_PRICE
PRICE_BREAK_TYPE_CO DE	POINT
OPERAND	420
PRICE_BREAK_HEADER _REF	OSLR0002

Parameter	Value
RLTD_MODIFIER_HEADER_REF	pRICE bREAK
PROCESS_FLAG	Y
INTERFACE_ACTION_CODE	INSERT
PROCESS_STATUS_FLAG	P

QP_INTERFACE_PRICING_ATTRIBS

Parameter	Value
ORIG_SYS_PRICING_ATTR_REF	OSPAR0003
ORIG_SYS_LINE_REF	OSLR0003
ORIG_SYS_HEADER_REF	CPL00001
EXCLUDER FLAG	Y
PRODUCT_ATTRIBUTE_CONTEXT	ITEM
PRODUCT_ATTR_CODE	INVENTORY_ITEM_ID
PRODUCT_ATTR_VAL_DISP	CM13139
PRICING_ATTRIBUTE_CONTEXT	VOLUME
PRICING_ATTR_CODE	ITEM_QUANTITY
PRICING_ATTR_VALUE_FROM_DISP	25

Parameter	Value
PRICING_ATTR_VALUE_ TO_DISP	35
COMPARISON_OPERAT OR_CODE	BETWEEN
PROCESS_FLAG	Y
INTERFACE_ACTION_C ODE	INSERT
PRODUCT_UOM_CODE	Ea
PROCESS_STATUS_FLAG	P

Example: Inserting a qualifier

QP_INTERFACE_QUALIFIERS

Parameter	Value
ORIG_SYS_QUALIFIER_R EF	OSQR0002
QUALIFIER_GROUPING_ NO	-1
QUALIFIER_CONTEXT	ORDER
QUALIFIER_ATTRIBUTE _CODE	SHIPPED_FLAG
QUALIFIER_ATTR_VAL UE_CODE	NO
ORIG_SYS_HEADER_REF	CPL00001
PROCESS_FLAG	Y

Parameter	Value
INTERFACE_ACTION_C ODE	INSERT
COMPARISON_OPERAT OR_CODE	=
PROCESS_STATUS_FLAG	P

Example: Deleting a price list line

QP_INTERFACE_LIST_LINES

Parameter	Value
ORIG_SYS_LINE_REF	OSLR0001
ORIG_SYS_HEADER_REF	CPL00001
PROCESS_FLAG	Y
INTERFACE_ACTION_C ODE	DELETE
PROCESS_STATUS_FLAG	P

Related Topics

Oracle Advanced Pricing Implementation Manual, Pricing Data Bulk Loader chapter

Pricing Object Security - Check Function API

This section explains how to use the Pricing Object Security CHECK_FUNCTION API and how it functions in Oracle Advanced Pricing.

The CHECK_FUNCTION API is used to check if a specific user, logging in with a specific responsibility and within a specific operating unit, can have the functional access to a specific pricing object or not. The package QP_SECURITY contains the function CHECK_FUNCTION. In our pricing data security system, we define two levels of functional access: QP_SECU_VIEW access and QP_SECU_UPDATE access. The data model is additive, so, this API will check all existing security rules and decide to authorize the user with the functional access or deny him.

Functional Overview

This API will take the parameters and check the pricing object security rules that have been set up in the pricing data security system.

The parameter `p_function_name` is required and can take two different values:

- `QP_SECU_VIEW`
- `QP_SECU_UPDATE`

The parameter `p_instance_type` is required and can take three different values:

- `PRL`: For type of standard price list.
- `MOD`: For type of modifier list.
- `AGR`: For type of agreement price list.

The parameter `p_instance_pk1_value` is required and is the `list_header_id` from `qp_list_headers_b` for a specific pricing object.

The parameter `p_instance_pk2_value` and `p_instance_pk3_value` are not in use now. This is for future consideration if the pricing objects to be secured have composite keys.

The parameter `p_user_name`, `p_resp_id` and `p_org_id` are optional. If they are not passed in, `CHECK_FUNCTION` will take the values from the current user and the corresponding settings for responsibility and operating unit.

This API will return one of the following three values:

- `T`: The user has the asking-for functional access to the pricing object.
- `F`: The user has been denied with the access.
- `E-<error message>`: Error occurred within the API.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Pricing Object Security Check API. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

CHECK_FUNCTION

The following table shows the parameters for this structure.

CHECK_FUNCTION

Parameter	Usage	Type	Req	Drv
p_function_name	In	Varchar2	Yes	No
p_instance_type	In	Varchar2	Yes	No
p_instance_pk1	In	Number	Yes	No
p_instance_pk2	In	Number	No	No
p_instance_pk3	In	Number	No	No
p_user_name	In	Varchar2	No	Yes
p_resp_id	In	Number	No	Yes
p_org_id	In	Number	No	Yes
Function return value	Out	Varchar2	Yes	No

A key of the short names and definitions used in the API tables are provided in the following table:

Short Names Key

Short name	Definition
Drv	Derived
Req	Required Yes : This is a required parameter. No : This is an optional parameter.
N/A (no entry)	No value/not applicable

Validation of Pricing Object Security API

Standard Validation

Oracle Advanced Pricing validates all required columns in the CHECK_FUNCTION API.

Other Validation

None

Error Handling

If any validation fails, the API will return error to the calling module.

Error Handling

Condition	Function Return Value
Success	T or F
Failure	E-error message

Pricing Attributes Application Program Interface

Functional Overview

The Pricing Attributes API (QP_ATTRIBUTES_PUB) is a Business Object API that is based on the following tables:

- QP_PRC_CONTEXTS_B/TL
- QP_SEGMENTS_B/TL

This API does all the transaction processing for all the data that is inserted, updated, or deleted from contexts and attributes. The Business API Model is shown below. The Object Name is Attributes and the object code is ATR.

Procedure PROCESS_ATTRIBUTES

The following table shows the parameters for this structure. This procedure is used to add, update, or delete contexts and attributes.

PROCESS_ATTRIBUTES

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No
x_return_status	Out	Number	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Varchar2	No	No
p_CON_rec	In	Con_Rec_Type	No	No
p_CON_val_rec	In	Con_Val_Rec_Type	No	No
p_SEG_tbl	In	Con_Tbl_Type	No	No
p_SEG_val_tbl	In	Con_Val_Tbl_Type	No	No
x_CON_rec	Out	Con_Rec_Type	No	No
x_CON_val_rec	Out	Con_Val_Rec_Type	No	No
x_SEG_tbl	Out	Con_Tbl_Type	No	No
x_SEG_val_tbl	Out	Con_Val_Tbl_Type	No	No

p_api_version_number

This the version number of the API.

P_CON_REC

The following table shows the parameters for this structure.

P_CON_REC

Parameter	Usage	Type	Req	Drv
Prc_context_code	Null	Varchar2	Yes	No
Prc_context_id	Null	Number	Yes	No
Prc_context_type	Null	Varchar2	Yes	No
Program_application_id	Null	Number	No	No
Program_id	Null	Number	No	No
Program_update_date	Null	Date	No	No
Seeded_description	Null	Varchar2	No	No
Seeded_flag	Null	Varchar2	Yes	No
Seeded_prc_context_name	Null	Varchar2	No	No
User_description	Null	Varchar2	No	No
User_prc_context_name	Null	Varchar2	No	No
Return_status	Null	Varchar2	No	No
Db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No

P_CON_VAL_REC

The following table shows the parameters for this structure.

P_CON_VAL_REC

Parameter	Usage	Type	Req	Drv
Enabled	Null	Varchar2	Yes	No
Prc_context	Null	Varchar2	Yes	No
Seeded	Null	Varchar2	Yes	No

P_SEG_TBL

The following table shows the parameters for this structure.

P_SEG_TBL

Parameter	Usage	Type	Req	Drv
P_Seg_Tbl	Null	TABLE OF Seg_Rec_Type		

P_SEG_VAL_TBL

The following table shows the parameters for this structure.

P_SEG_VAL_TBL

Parameter	Usage	Type	Req	Drv
P_Seg_Val_Tbl	Null	TABLE OF Seg_Val_Rec_Type		

X_CON_REC

The following table shows the parameters for this structure.

X_CON_REC

Parameter	Usage	Type	Req	Drv
Prc_context_code	Null	Varchar2	No	No
Prc_context_id	Null	Number	No	No
Prc_context_type	Null	Varchar2	No	No
Program_application_id	Null	Number	No	No
Program_id	Null	Number	No	No
Program_update_date	Null	Date	No	No
Seeded_description	Null	Varchar2	No	No
Seeded_flag	Null	Varchar2	No	No
Seeded_prc_context_name	Null	Varchar2	No	No
User_description	Null	Varchar2	No	No
User_prc_context_name	Null	Varchar2	No	No
Return_status	Null	Varchar2	No	No
Db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No

X_CON_VAL_REC

The following table shows the parameters for this structure.

X_CON_VAL_REC

Parameter	Usage	Type	Req	Drv
Enabled	Null	Varchar2	No	No
Prc_context	Null	Varchar2	No	No
Seeded	Null	Varchar2	No	No

X_SEG_TBL

The following table shows the parameters for this structure.

X_SEG_TBL

Parameter	Usage	Type	Req	Drv
-	Null	TABLE OF Seg_Rec_Type	No	No

X_SEG_VAL_TBL

The following table shows the parameters for this structure.

X_SEG_VAL_TBL

Parameter	Usage	Type	Req	Drv
-	Null	TABLE OF Seg_Val_Rec_Type	No	No

Procedure LOCK_ATTRIBUTES

The following table shows the parameters for this structure. Users can use this procedure to lock a context row and its attributes from getting updated by another user concurrently.

LOCK_ATTRIBUTES

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	Yes	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
x_return_status	Out	Number	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Varchar2	No	No
p_CON_rec	In	Con_Rec_Type	No	No
p_CON_val_rec	In	Con_Val_Rec_Type	No	No
p_SEG_tbl	In	Con_Tbl_Type	No	No
p_SEG_val_tbl	In	Con_Val_Tbl_Type	No	No
x_CON_rec	Out	Con_Rec_Type	No	No
x_CON_val_rec	Out	Con_Val_Rec_Type	No	No
x_SEG_tbl	Out	Con_Tbl_Type	No	No
x_SEG_val_tbl	Out	Con_Val_Tbl_Type	No	No

p_api_version_number

This the version number of the API.

P_CON_REC

The following table shows the parameters for this structure.

P_CON_REC

Parameter	Usage	Type	Req	Drv
Prc_context_code	Null	Varchar2	Yes	No
Prc_context_id	Null	Number	Yes	No
Prc_context_type	Null	Varchar2	Yes	No
Program_application_id	Null	Number	No	No
Program_id	Null	Number	No	No
Program_update_date	Null	Date	No	No
Seeded_description	Null	Varchar2	No	No
Seeded_flag	Null	Varchar2	Yes	No
Seeded_prc_context_name	Null	Varchar2	No	No
User_description	Null	Varchar2	No	No
User_prc_context_name	Null	Varchar2	No	No
Return_status	Null	Varchar2	No	No
Db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No

P_CON_VAL_REC

The following table shows the parameters for this structure.

P_CON_VAL_REC

Parameter	Usage	Type	Req	Drv
Enabled	Null	Varchar2	Yes	No
Prc_context	Null	Varchar2	Yes	No
Seeded	Null	Varchar2	Yes	No

The following table shows the parameters for these table structures:

Parameters

Table name	Usage	Type
P_SEG_TBL	Null	TABLE OF Seg_Rec_Type
P_SEG_VAL_TBL	Null	TABLE OF Seg_Val_Rec_Type

X_CON_REC

The following table shows the parameters for this structure.

X_CON_REC

Parameter	Usage	Type	Req	Drv
Prc_context_code	Null	Varchar2	No	No
Prc_context_id	Null	Number	No	No
Prc_context_type	Null	Varchar2	No	No
Program_application_id	Null	Number	No	No
Program_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
Program_update_date	Null	Date	No	No
Seeded_description	Null	Varchar2	No	No
Seeded_flag	Null	Varchar2	No	No
Seeded_prc_context_name	Null	Varchar2	No	No
User_description	Null	Varchar2	No	No
User_prc_context_name	Null	Varchar2	No	No
Return_status	Null	Varchar2	No	No
Db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No

X_CON_VAL_REC

The following table shows the parameters for this structure.

X_CON_VAL_REC

Parameter	Usage	Type	Req	Drv
Enabled	Null	Varchar2	No	No
Prc_context	Null	Varchar2	No	No
Seeded	Null	Varchar2	No	No

X_SEG_TBL

The following table shows the parameters for this structure.

X_SEG_TBL

Parameter	Usage	Type	Req	Drv
X_Seg_Tbl	Null	TABLE OF Seg_Rec_Type		

X_SEG_VAL_TBL

The following table shows the parameters for this structure.

X_SEG_VAL_TBL

Parameter	Usage	Type	Req	Drv
X_Seg_Val_Tbl	Null	TABLE OF Seg_Val_Rec_Type		

Procedure GET_ATTRIBUTES

The following table shows the parameters for this structure. This procedure fetches the context and all its attributes, when prc_context_id is provided.

GET_ATTRIBUTES

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	Yes	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
x_return_status	Out	Number	No	No
x_msg_count	Out	Varchar2	No	No
x_msg_data	Out	Varchar2	No	No
p_prc_context_id	In	Number	No	No

Parameter	Usage	Type	Req	Drv
p_prc_context	In	Varchar2	No	No
x_CON_rec	Out	Con_Rec_Type	No	No
x_CON_val_rec	Out	Con_Val_Rec_Type	No	No
x_SEG_tbl	Out	Con_Tbl_Type	No	No
x_SEG_val_tbl	Out	Con_Val_Tbl_Type	No	No

p_api_version_number

This the version number of the API.

p_prc_context_id

This the primary key for a given context. The API will fetch all the context and all the attributes for this value.

p_prc_context

This the unique code for a given context. The API will fetch all the context and all the attributes for this value.

X_CON_REC

The following table shows the parameters for this structure.

X_CON_REC

Parameter	Usage	Type	Req	Drv
Prc_context_code	Null	Varchar2	No	No
Prc_context_id	Null	Number	No	No
Prc_context_type	Null	Varchar2	No	No
Program_application_id	Null	Number	No	No

Parameter	Usage	Type	Req	Drv
Program_id	Null	Number	No	No
Program_update_date	Null	Date	No	No
Seeded_description	Null	Varchar2	No	No
Seeded_flag	Null	Varchar2	No	No
Seeded_prc_context_name	Null	Varchar2	No	No
User_description	Null	Varchar2	No	No
User_prc_context_name	Null	Varchar2	No	No
Return_status	Null	Varchar2	No	No
Db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	No	No

X_CON_VAL_REC

The following table shows the parameters for this structure.

X_CON_VAL_REC

Parameter	Usage	Type	Req	Drv
Enabled	Null	Varchar2	No	No
Prc_context	Null	Varchar2	No	No
Seeded	Null	Varchar2	No	No

X_SEG_TBL

The following table shows the parameters for this structure.

X_SEG_TBL

Parameter	Usage	Type	Req	Drv
X_Seg_Tbl	Null	TABLE OF Seg_Rec_Type		

X_SEG_VAL_TBL

The following table shows the parameters for this structure.

X_SEG_VAL_TBL

Parameter	Usage	Type	Req	Drv
X_Seg_Val_Tbl	Null	TABLE OF Seg_Val_Rec_Type		

Attribute Mapping Application Program Interface

Functional Overview

The build_contexts API calls QP_BUILD_SOURCING_PVT, which contains sourcing calls to build attribute mapping rules for attributes. This API is called from the Oracle Order Management (OM) pricing integration. OM passes the request_type_code and pricing_type and this API builds the attribute mapping rules of the attributes used in the setup that can be sourced at run time. Attribute mapping rules thus built are returned in the PL/SQL table structures x_price_contexts_result_tbl and x_qual_contexts_result_tbl.

Setting Up and Parameter Descriptions

The following table describes all parameters used by the Attribute Mapping API (QP_ATTR_MAPPING_PUB). All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

QP_ATTR_MAPPING_PUB

Parameter	Usage	Type	Req	Drv
p_request_type_code	In	Varchar2	Yes	No
p_pricing_type	In	Varchar2	Yes	No
P_org_id	In	Number	No	Yes
x_price_contexts_result_tbl	Out	CONTEXTS_RESULT_TBL_TYPE	No	No
x_qual_contexts_result_tbl	Out	CONTEXTS_RESULT_TBL_TYPE	No	No

p_request_type_code

An allowable value from the QP_PRICING_REQ_SOURCES table. This is different from source system code; groups of source system codes form a request type.

For example, CRM request type code can have the I-marketing and I-store source system codes associated with it (using QP_PRICING_REQ_SOURCES). When you call Build_Contexts with the CRM request type, the process sources all contexts for both I-marketing and I-store source systems.

p_pricing_type

Indicates the sourcing rules to use when building the attributes. Allowable values are:

- L (Line): The process builds the pricing contexts based on the rules associated with the line pricing attributes entity and builds the qualifier contexts based on the rules associated with the line qualifier attribute
- H (Header): The process builds the pricing contexts based on the rules associated with the header pricing attributes entity and builds the qualifier contexts based on the rules associated with the header qualifier attributes.

p_org_id

Indicates the Org_id (Operating Unit) of the transaction. This parameter is added for MOAC compliance. The Build_context API can source the attributes for that OU context. In case the p_org_id is not passed, the API would derive the value from the OU context if set, and if OU context is not set, the API will use the Default OU for the responsibility.

Note: The Build_Contexts API will set the policy_context to single Org to the Org_Id passed to the API. If Org_Id was not passed to the Build_Contexts API , the API will check if Org Context has been set; if Org Context is not set, the policy context will be set to Single Org with the default Org for that responsibility.

CONTEXTS_RESULT_TBL_TYPE

The following table shows the parameters for this structure.

CONTEXTS_RESULT_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Contexts_Result _Tbl_Type	Out	Contexts_Result _Rec_Type	No	NO

CONTEXTS_RESULT_REC_TYPE

The following table shows the parameters for this structure.

CONTEXTS_RESULT_REC_TYPE

Parameter	Usage	Type	Req	Drv
context_name	Null	Varchar2	No	No
attribute_name	Null	Varchar2	No	No
Attribute_value	Null	Varchar2	No	No

Procedure BUILD_CONTEXTS (overloaded)

This API is called from the Order Management (OM) pricing integration in the performant code path. OM passes request_type_code and pricing_type and this API builds the attribute mapping rules of the attributes used in the setup that can be sourced at run time. Attribute mapping rules thus built are inserted into the pricing temporary table QP_PREQ_LINE_ATTRS_TMP for the passed in line_index.

The following table shows the parameters for this structure.

BUILD_CONTEXTS (overloaded)

Parameter	Usage	Type	Req	Drv
p_request_type_code	In	Varchar2	Yes	No
p_line_index	In	Number	Yes	No
p_pricing_type_code	In	Varchar2	Yes	No
P_org_id	In	Number	No	Yes

p_request_type_code

An allowable value from the QP_PRICING_REQ_SOURCES table. This is different from source system code; groups of source system codes form a request type.

For example, CRM request type code can have the I-marketing and I-store source system codes associated with it (using QP_PRICING_REQ_SOURCES). When you call Build_Contexts with the CRM request type, the process sources all contexts for both I-marketing and I-store source systems.

P_line_index

Unique identifier for line in the table qp_preq_lines_tmp.

p_pricing_type

Indicates the sourcing rules to use when building the attributes. Allowable values are:

- L (Line): The process builds the pricing contexts based on the rules associated with the line pricing attributes entity and builds the qualifier contexts based on the rules associated with the line qualifier attribute
- H (Header): The process builds the pricing contexts based on the rules associated with the header pricing attributes entity and builds the qualifier contexts based on the rules associated with the header qualifier attributes.

p_org_id

Indicates the Org_id (Operating Unit) of the transaction. This parameter is added for the MOAC compliance. The Build_context API can source the attributes for that OU context. In case the p_org_id is not passed, the API would derive the value from the OU context if set, and if OU context is not set, the API will use the Default OU for the responsibility.

Note: The Build_Contexts API will set the policy_context to single Org to the Org_Id passed to the API. If Org_Id was not passed to the Build_Contexts API , the API will check if Org Context has been set, if Org Context is not set, it will set the policy context to Single Org with the default Org for that responsibility.

Procedure Get_User_Item_Pricing_Contexts

This API returns the pricing contexts whose sourcing method is USER ENTERED. This API is called from the OM pricing integration. OM passes request_type_code and this API returns the pricing contexts whose sourcing method is USER ENTERED in the PLSQL table structure p_user_attris_tbl.

The following table shows the parameters for this structure.

Get_User_Item_Pricing_Contexts

Parameter	Usage	Type	Req	Drv
p_request_type_code	In	Varchar2	Yes	No
p_user_attris_tbl	Out	USER_ATTRIBUTE_TBL_TYPE	No	No

p_request_type_code

An allowable value from the QP_PRICING_REQ_SOURCES table. This is different from source system code; groups of source system codes form a request type.

For example, CRM request type code can have the I-marketing and I-store source system codes associated with it (using QP_PRICING_REQ_SOURCES). When you call Build_Contexts with the CRM request type, the process sources all contexts for both I-marketing and I-store source systems.

USER_ATTRIBUTE_TBL_TYPE

The following table shows the parameters for this structure.

USER_ATTRIBUTE_TBL_TYPE

Parameter	Usage	Type	Req	Drv
User_Attribute_ Tbl_Type	Out	User_Attribute_ Rec_Type	No	No

USER_ATTRIBUTE_REC_TYPE

The following table shows the parameters for this structure.

USER_ATTRIBUTE_REC_TYPE

Parameter	Usage	Type	Req	Drv
context_name	Null	Varchar2	No	No
attribute_name	Null	Varchar2	No	No

Function Is_attribute_used

This API is used to find out whether pricing attribute is used in the pricing setup. This API is called by the pricing engine. The pricing engine passes context and pricing attribute to this function and this API returns the attribute used flag (Y/N).

The following table shows the parameters for this structure.

Is_attribute_used

Parameter	Usage	Type	Req	Drv
p_attribute_context	In	Varchar2	No	No
p_attribute_code	In	Varchar2	No	No

p_attribute_context

A context name set up in the context definition table, for example, item, customer, or volume.

p_attribute_code

Name of the attribute for the sourcing rule, for example, PRICING_ATTRIBUTE1,

PRICING_ATTRIBUTE2, or QUALIFIER_ATTRIBUTE1.

Validation of Attribute Mapping API

Validations.

Standard Validation

Oracle Advanced Pricing validates all required columns in the Attribute Mapping API.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Attribute Mapping API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	Null
Failure	4	actual error message

Qualifiers Application Program Interface

This section explains how to use the Qualifiers API and how it functions in Oracle Advanced Pricing. The Qualifiers package consists of entities to set up qualifiers.

Functional Overview

The Qualifiers package QP_QUALIFIERS_RULES_PUB contains the following APIs and record type definitions:

- Qualifier_Rules_Rec_Type
- Qualifier_Rules_Tbl_Type
- Qualifier_Rules_Val_Rec_Type
- Qualifier_Rules_Val_Tbl_Type
- Qualifiers_Rec_Type
- Qualifiers_Tbl_Type

- `Qualifiers_Val_Rec_Type`
- `Qualifiers_Val_Tbl_Type`
- `Process_Qualifier_Rules`: Creates, updates, and deletes pricing qualifier rules and pricing qualifiers belonging to those rules.
- `Lock_Qualifier_Rules`: Locks table rows.
- `Get_Qualifier_Rules`: Retrieves the qualifier rule and qualifiers for a given qualifier rule ID or qualifier rule name.
- `Copy_Qualifier_Rules`: Creates a new qualifier rule with the name in `p_to_qualifier_rule`. Also, copies the qualifiers belonging to the qualifier rule specified by the qualifier rule in `p_qualifier_rule` or the qualifier rule ID in `p_qualifier_rule_id` into a new qualifier rule. Returns qualifier rule ID for the new qualifier rule.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Qualifiers. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow.

PROCESS_QUALIFIER_RULES

The following table shows the parameters for this structure.

PROCESS_QUALIFIER_RULES

Parameter	Usage	Type	Req	Drv
<code>p_api_version_number</code>	In	Number	No	No
<code>p_init_msg_list</code>	In	Varchar2	No	No
<code>p_return_values</code>	In	Varchar2	No	No
<code>p_commit</code>	In	Varchar2	No	No
<code>x_return_status</code>	Out	Varchar2	No	No
<code>x_msg_count</code>	Out	Number	No	No

Parameter	Usage	Type	Req	Drv
x_msg_data	Out	Varchar2	No	No
p_QUALIFIER_RULES_rec	In	Qualifier_Rules_Rec_Type	No	No
p_QUALIFIER_RULES_val_rec	In	Qualifier_Rules_Val_Rec_Type	No	No
p_QUALIFIERS_tbl	In	Qualifiers_Tbl_Type	No	No
p_QUALIFIERS_val_tbl	In	Qualifiers_Val_Tbl_Type	No	No
x_QUALIFIER_RULES_rec	Out	Qualifier_Rules_Rec_Type	No	No
x_QUALIFIER_RULES_val_rec	Out	Qualifier_Rules_Val_Rec_Type	No	No
x_QUALIFIERS_tbl	Out	Qualifiers_Tbl_Type	No	No
x_QUALIFIERS_val_tbl	Out	Qualifiers_Val_Tbl_Type	No	No

p_init_msg_list

Default Value: FND_API.G_FALSE

p_return_values

Default Value: FND_API.G_FALSE

p_commit

Default Value: FND_API.G_FALSE

p_QUALIFIER_RULES_rec

In this procedure, P_QUALIFIER_RULES_rec.operation states the operation that the process should perform. Allowable values of P_QUALIFIER_RULES_rec.operation are:

- QP_GLOBALS.G_OPR_CREATE
- QP_GLOBALS.G_OPR_DELETE

- QP_GLOBALS.G_OPR_UPDATE
- QP_GLOBALS.G_OPR_LOCK
- QP_GLOBALS.G_OPR_NONE

Default Value: G_MISS_QUALIFIER_RULES_REC

p_QUALIFIER_RULES_val_rec

Default Value: G_MISS_QUALIFIER_RULES_VAL_REC

p_QUALIFIERS_tbl

Default Value: G_MISS_QUALIFIERS_TBL

p_QUALIFIERS_val_tbl

Default Value: G_MISS_QUALIFIERS_VAL_TBL

QUALIFIER_RULES_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIER_RULES_REC_TYPE

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
creation_date	Null	Date	No	No
description	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_logi n	Null	Number	No	No
name	Null	Varchar2	Yes	No
program_applica tion_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update _date	Null	Date	No	No
qualifier_rule_id	Null	Number	No	No
request_id	Null	Number	No	No
return_status	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No

attribute1-15

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

description

Default Value: FND_API.G_MISS_CHAR

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

name

Default Value: FND_API.G_MISS_CHAR

program_application_id

Default Value: FND_API.G_MISS_NUM

program_id

Default Value: FND_API.G_MISS_NUM

program_update_date

Default Value: FND_API.G_MISS_DATE

qualifier_rule_id

Default Value: Comes from the sequence QP_QUALIFIER_RULES_S

request_id

Default Value: FND_API.G_MISS_NUM

return_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

QUALIFIER_RULES_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIER_RULES_TBL_TYPE

Parameter	Usage	Type	Req	Drv
Qualifier_Rules_Rec_Type	Null	Record	No	No

QUALIFIER_RULES_VAL_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIER_RULES_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
qualifier_rule	Null	Varchar2	No	No

qualifier_rule

Default Value: FND_API.G_MISS_CHAR

QUALIFIER_RULES_VAL_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIER_RULES_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Optional	Qualifier_Rules_ Val_Rec_Type	Null	Record	No

QUALIFIERS_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_REC_TYPE Parameters

Parameter	Usage	Type	Req	Drv
attribute1	Null	Varchar2	No	No
attribute2	Null	Varchar2	No	No
attribute3	Null	Varchar2	No	No
attribute4	Null	Varchar2	No	No
attribute5	Null	Varchar2	No	No
attribute6	Null	Varchar2	No	No
attribute7	Null	Varchar2	No	No
attribute8	Null	Varchar2	No	No
attribute9	Null	Varchar2	No	No
attribute10	Null	Varchar2	No	No
attribute11	Null	Varchar2	No	No
attribute12	Null	Varchar2	No	No
attribute13	Null	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
attribute14	Null	Varchar2	No	No
attribute15	Null	Varchar2	No	No
comparison_operator_code	Null	Varchar2	Yes	No
context	Null	Varchar2	No	No
created_by	Null	Number	No	No
created_from_rule_id	Null	Number	No	No
creation_date	Null	Date	No	No
end_date_active	Null	Date	No	No
excluder_flag	Null	Varchar2	No	No
last_updated_by	Null	Number	No	No
last_update_date	Null	Date	No	No
last_update_login	Null	Number	No	No
list_header_id	Null	Number	No	No
list_line_id	Null	Number	No	No
program_application_id	Null	Number	No	No
program_id	Null	Number	No	No
program_update_date	Null	Date	No	No
qualifier_attribute	Null	Varchar2	Yes	No

Parameter	Usage	Type	Req	Drv
qualifier_attr_value	Null	Varchar2	Yes	No
qualifier_attr_value_to	Null	Varchar2	No	No
qualifier_context	Null	Varchar2	Yes	No
qualifier_datatype	Null	Varchar2	No	Yes
qualifier_grouping_no	Null	Number	No	No
qualifier_id	Null	Number	No	No
qualifier_precedence	Null	Number	No	No
qualifier_rule_id	Null	Number	No	No
request_id	Null	Number	No	No
start_date_active	Null	Date	No	No
qual_attr_value_from_number	Null	Number	No	Yes
qual_attr_value_to_number	Null	Number	No	Yes
active_flag	Null	Varchar2	No	No
search_ind	Null	Number	No	Yes
qualifier_group_cnt	Null	Number	No	Yes
header_qual_explicit_flag	Null	Varchar2	No	Yes

Parameter	Usage	Type	Req	Drv
distinct_row_count	Null	Number	No	Yes
return_status	Null	Varchar2	No	No
db_flag	Null	Varchar2	No	No
operation	Null	Varchar2	Yes	No
qualify_hier_descendent_flag	Null	Varchar2	No	No

attribute1–15

Default Value: FND_API.G_MISS_CHAR

comparison_operator_code

Default Value: FND_API.G_MISS_CHAR

context

Default Value: FND_API.G_MISS_CHAR

created_by

Default Value: FND_API.G_MISS_NUM

created_from_rule_id

Default Value: FND_API.G_MISS_NUM

creation_date

Default Value: FND_API.G_MISS_DATE

end_date_active

Default Value: FND_API.G_MISS_DATE

excluder_flag

Default Value: 'N'

last_updated_by

Default Value: FND_API.G_MISS_NUM

last_update_date

Default Value: FND_API.G_MISS_DATE

last_update_login

Default Value: FND_API.G_MISS_NUM

list_header_id

Default Value: FND_API.G_MISS_NUM

list_line_id

Default Value: FND_API.G_MISS_NUM

program_application_id

Default Value: FND_API.G_MISS_NUM

program_id

Default Value: FND_API.G_MISS_NUM

program_update_date

Default Value: FND_API.G_MISS_DATE

qualifier_attribute

Default Value: FND_API.G_MISS_CHAR

qualifier_attr_value

Default Value: FND_API.G_MISS_CHAR

qualifier_attr_value_to

Default Value: FND_API.G_MISS_CHAR

qualifier_context

Default Value: FND_API.G_MISS_CHAR

qualifier_datatype

Default Value: FND_API.G_MISS_CHAR

qualifier_grouping_no

Default Value: FND_API.G_MISS_NUM

qualifier_id

Default Value: From sequence QP_QUALIFIERS_S

qualifier_precedence

Default Value: FND_API.G_MISS_NUM

qualifier_rule_id

Default Value: FND_API.G_MISS_NUM

request_id

Default Value: FND_API.G_MISS_NUM

start_date_active

Default Value: FND_API.G_MISS_DATE

qual_attr_value_from_number

Default Value: FND_API.G_MISS_NUM

qual_attr_value_to_number

Default Value: FND_API.G_MISS_NUM

active_flag

Default Value: FND_API.G_MISS_CHAR

search_ind

Default Value: FND_API.G_MISS_NUM

qualifier_group_cnt

Default Value: FND_API.G_MISS_NUM

header_qual_exists_flag

Default Value: FND_API.G_MISS_CHAR

distinct_row_count

Default Value: FND_API.G_MISS_NUM

return_status

Default Value: FND_API.G_MISS_CHAR

db_flag

Default Value: FND_API.G_MISS_CHAR

operation

Default Value: FND_API.G_MISS_CHAR

qualify_hier_descendent_flag

Default Value: FND_API.G_MISS_CHAR

QUALIFIERS_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Qualifiers_Rec_Type	Null	Record	No	No

QUALIFIER_VAL_REC_TYPE

The following table shows the parameters for this structure.

QUALIFIER_VAL_REC_TYPE

Parameter	Usage	Type	Req	Drv
created_from_rule	Null	Varchar2	No	No
list_header	Null	Varchar2	No	No
list_line	Null	Varchar2	No	No
qualifier_rule	Null	Varchar2	No	No
qualifier_attribute_desc	Null	Varchar2	No	No
qualifier_attr_value_desc	Null	Varchar2	No	No
qualifier_attr_value_to_desc	Null	Varchar2	No	No

created_from_rule

Default Value: FND_API.G_MISS_CHAR

list_header

Default Value: FND_API.G_MISS_CHAR

list_line

Default Value: FND_API.G_MISS_CHAR

qualifier_rule

Default Value: FND_API.G_MISS_CHAR

qualifier_attribute_desc

Default Value: FND_API.G_MISS_CHAR

qualifier_attr_value_desc

Default Value: FND_API.G_MISS_CHAR

qualifier_rule_value_to_desc

Default Value: FND_API.G_MISS_CHAR

QUALIFIERS_VAL_TBL_TYPE

The following table shows the parameters for this structure.

QUALIFIERS_VAL_TBL_TYPE Parameters

Parameter	Usage	Type	Req	Drv
Qualifiers_Val_Record_Type	Null	Record	No	No

LOCK_QUALIFIER_RULES

The following table shows the parameters for this structure.

LOCK_QUALIFIER_RULES Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Number	No	No
x_msg_data	Out	Varchar2	No	No
p_QUALIFIER_RULES_rec	In	Qualifier_Rules_Rec_Type	No	No
p_QUALIFIER_RULES_val_rec	In	Qualifier_Rules_Val_Rec_Type	No	No
p_QUALIFIERS_tbl	In	Qualifiers_Tbl_Type	No	No
p_QUALIFIERS_val_tbl	In	Qualifiers_Val_Tbl_Type	No	No

Parameter	Usage	Type	Req	Drv
x_QUALIFIER_RULES_rec	Out	Qualifier_Rules_Rec_Type	No	No
x_QUALIFIER_RULES_val_rec	Out	Qualifier_Rules_Val_Rec_Type	No	No
x_QUALIFIERS_tbl	Out	Qualifiers_Tbl_Type	No	No
x_QUALIFIERS_val_tbl	Out	Qualifiers_Val_Tbl_Type	No	No

p_init_msg_list

Default Value: FND_API.G_FALSE

p_return_values

Default Value: FND_API.G_FALSE

p_QUALIFIER_RULES_rec

Default Value: G_MISS_QUALIFIER_RULES_REC

p_QUALIFIER_RULES_val_rec

Default Value: G_MISS_QUALIFIER_RULES_VAL_REC

p_QUALIFIERS_tbl

Default Value: G_MISS_QUALIFIERS_TBL

p_QUALIFIERS_val_tbl

Default Value: G_MISS_QUALIFIERS_VAL_TBL

GET_QUALIFIER_RULES

The following table shows the parameters for this structure.

GET_QUALIFIER_RULES Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No

Parameter	Usage	Type	Req	Drv
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Number	No	No
x_msg_data	Out	Varchar2	No	No
p_qualifier_rule_id	In	Number	No	No
p_qualifier_rule	In	Varchar2	No	No
x_QUALIFIER_RULES_rec	Out	Qualifier_Rules_Rec_Type	No	No
x_QUALIFIER_RULES_val_rec	Out	Qualifier_Rules_Val_Rec_Type	No	No
x_QUALIFIERS_tbl	Out	Qualifiers_Tbl_Type	No	No
x_QUALIFIERS_val_tbl	Out	Qualifiers_Val_Tbl_Type	No	No

p_init_msg_list

Default Value: FND_API.G_FALSE

p_return_values

Default Value: FND_API.G_FALSE

p_qualifier_rule_id

Default Value: FND_API.G_MISS_NUM

p_qualifier_rule

Default Value: FND_API.G_MISS_CHAR

COPY_QUALIFIER_RULES

The following table shows the parameters for this structure.

COPY_QUALIFIER_RULES Parameters

Parameter	Usage	Type	Req	Drv
p_api_version_number	In	Number	No	No
p_init_msg_list	In	Varchar2	No	No
p_return_values	In	Varchar2	No	No
p_commit	In	Varchar2	No	No
x_return_status	Out	Varchar2	No	No
x_msg_count	Out	Number	No	No
x_msg_data	Out	Varchar2	No	No
p_qualifier_rule_id	In	Number	No	No
p_qualifier_rule	In	Varchar2	No	No
p_to_qualifier_rule	In	Varchar2	No	No
p_to_description	In	Varchar2	No	No
x_qualifier_rule_id	Out	Number	No	No

p_init_msg_list

Default Value: FND_API.G_FALSE

p_return_values

Default Value: FND_API.G_FALSE

p_commit

Default Value: FND_API.G_FALSE

p_qualifier_rule_id

Default Value: FND_API.G_MISS_NUM

p_qualifier_rule

Default Value: FND_API.G_MISS_CHAR

p_to_description

Default Value: FND_API.G_MISS_CHAR

Validation of Qualifiers API

Validations:

Standard Validation

Oracle Advanced Pricing validates all required columns in the Qualifiers API.

Other Validation

None

Error Handling

If any validation fails, the API will return error status to the calling module. The Qualifiers API processes the rows and reports the values in the following table for every record.

Error Handling

Condition	PROCESS_STATUS	ERROR_MESSAGE
Success	5	null
Failure	4	actual error message

Example of Qualifiers API

The following example script is located in the directory \$qp/patch/115/sql:

- QPQFXMP1.sql: This sample script inserts a qualifier Rule and a qualifier.

Reverse Limits Application Program Interface

This section explains how to use the Reverse Limits API and how it functions in Oracle Advanced Pricing.

The Reverse Limits API is a procedure that must be called directly by Oracle Order Management (OM) to reverse any promotional limits that were consumed by an order line when the line is either cancelled, amended or returned and no pricing engine call is made. The package QP_UTIL_PUB contains the procedure Reverse_Limits.

Note: The follow is not supported by this API:

- When the API is called with p_action_code = 'AMEND', an amended quantity that is greater than the originally ordered or consumed quantity is not supported.
- Reversal of order level discounts for an order line cancellation, amendment, or return is not supported.

Functional Overview

When an item is ordered in OM, a call is made to the pricing engine to price the item and to determine the promotions that are to be given away for the ordered item. If promotional limits have been set up for the promotions, then the promotions that are given away are consumed from this limit.

However, when the order line is either cancelled, amended or returned without making a pricing engine call (since there was no need to calculate the price of return), the promotional limits that were consumed do not get reversed.

To reverse the consumed limit when such a return, amend or cancellation occurs, OM must call this API with the right parameters to increment the available limit balance(s).

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the public Reverse Limits API. All of the inbound and outbound parameters are listed. Additional information on these parameters may follow. The input parameters must be passed to the Reverse Limits API by OM and the output parameters are passed to OM.

REVERSE_LIMITS

The following table shows the parameters for this structure:

Reverse_Limits Parameters

Parameter	Usage	Type	Req	Description
p_action_code	In	Varchar2	Yes	Valid values are RETURN,CANCEL and AMEND

Parameter	Usage	Type	Req	Description
p_cons_price_request_code	In	Varchar2	Yes	<p>Price_Request_Code of the last, successfully priced order_line that consumed the promotional limit:</p> <ul style="list-style-type: none"> • If the p_action_code is RETURN, then the price_request_code of the order line that is referenced by the returned line must be passed to this parameter. • If the p_action_code is CANCEL, then to this parameter must be passed the price_request_code of the order line/order that is being cancelled. • If the p_action_code is AMEND, then to this parameter

Parameter	Usage	Type	Req	Description
				must be passed the price_reques t_code of the order line that is being amended

Parameter	Usage	Type	Req	Description
p_orig_ordered_qty	In	Number	Yes1	<p>Priced_quantity of the last successfully priced order line that consumed the promotional limit:</p> <ul style="list-style-type: none"> • If the p_action_code is 'RETURN', then to this parameter must be passed the priced quantity of the order line that is referenced by the returned line. • If the p_action_code is AMEND, then to this parameter must be passed the originally priced quantity of the order line that is being amended.

Parameter	Usage	Type	Req	Description
p_amended_qty	In	Number	Yes2	The new quantity to which the original ordered or priced quantity has been amended to. The Amended Qty can be less than or equal to the originally priced qty but not greater.
p_ret_price_request_code	In	Varchar2	Yes3	The price request code of the return line.
P_returned_qty	In	Number	Yes4	The returned quantity (the quantity on the return line).
x_return_status	Out	Varchar2	Yes	See Error Handling for possible values returned.
x_return_message	Out	Varchar2	Yes	Suitable Error Message if the API returns an error status.

The following table describes the notations listed in the preceding table:

Notations

Note	Description
1	When p_action_code is RETURN or AMEND

Note	Description
2	When p_action_code is AMEND
3	When p_action_code is RETURN
4	When p_action_code is RETURN

Validation of Reverse Limits API

Validations:

Standard Validation

The calling application is responsible for passing the correct parameters to the Reverse Limits API.

Other Validation

None

Error Handling

If any exception occurs in the Reverse Limits API, it should return the error status to the calling module. The Order Management application must call the Reverse Limits API while processing Cancellations, Amendments or Returned Order Lines, when the pricing engine is not called.

Error Handling

Condition	x_return_status	x_return_message
Success	FND_API.G_RET_STS_SUCCESS	null
Expected Error	FND_API.G_RET_STS_ERROR	actual error message
Unexpected Error	FND_API.G_RET_STS_UNEXP_ERROR	actual error message

Note: The Package Specification and Body files are QPXRTCNS.pls and QPXRTCNB.pls and are available under the source control directory \$qp/patch/115/sql.

Additional Notes

- When the API is called with p_action_code='AMEND', the amended quantity can less than or equal to the originally ordered or consumed quantity but not greater than the originally priced quantity. If the amended quantity is greater, then the promotional limit may potentially be exceeded, in which case, this API (which is independent of the Limits Engine) cannot perform the logic that the Pricing Limits Engine would have performed under such circumstances. (Also, Oracle Order Management handles limit reversal for under-shipped or over-shipped quantities).
- When an order line is cancelled, amended or returned, this API may be used to only reverse limits on order-line level Discounts but not order-level discounts.
- Order Cancellations are supported by this API. To achieve this, the calling application (OM) must call this API for each order line on the order to be cancelled and once with the Order's price_request_code. Number of times to call the API = no. of order lines on the order to be cancelled + 1.
- The limit amount to be reversed/amended is calculated as a prorated value of the originally consumed amount.

Example of Reverse Limits API

Example 1

The limit amount to be reversed/amended is calculated as a prorated value of the originally consumed amount as in the following examples:

1. If an order line item has a unit price of \$100 and the original priced/ordered quantity is 10 then the gross revenue is \$1000. If the order line is eligible for a lumpsum discount of \$20, then the actual revenue is \$1000 - \$20 = \$980 which is a selling price of \$98 (\$980/10) per unit of the item effectively.

If the priced quantity is amended to 5 from 10, then since selling price must be fixed at \$98 (since there's no repricing), the new total should be $5 \times \$98 = \490 which is effectively a lumpsum discount of \$10 for an order line of qty 5 with unit price = \$10 (for example, $5 \times \$10 - \10).

2. With the pro-rated approach, the new benefit consumption against the limit would be calculated as follows:

$(\text{amended_qty} / \text{original_ordered_qty}) * \text{original consumed amount}$

$(5/10) * \$20$ (lumpsum benefit) = \$10 (new lumpsum) which is the same as the benefit(discount) calculated in a) above.

So pro-rating the original consumption correctly evaluates the new consumption against a limit when the quantity is amended.

Example 2

1. If an order line item has a unit price of \$100 and the original priced/ordered quantity is 10 then the gross revenue is \$1000. If the order line is eligible for a 30%

discount (for example, $30/100 \times \$1000 = \300), then the actual revenue is $\$1000 - \$300 = \$700$ which is a selling price of \$70 ($\$700/10$) per unit of the item effectively.

If the priced quantity is amended to 6 from 10, then the selling price must be fixed at \$70 (since there is no repricing), and the new total should be $6 \times \$70 = \420 . This is effectively a discount of \$180 ($6 \times \$100 - \420) which is an 18% percent discount consumption ($\$180/\1000)

2. With the pro-rated approach, the new benefit consumption against the limit would be:

$(\text{amended_qty} / \text{original_ordered_qty}) * \text{original consumed amount}$

$(6/10) * 30 (\text{discount}\%) = 18 (\text{new discount}\%)$ which is the same as the benefit (discount %) calculated in a) above.

So pro-rating the original consumption correctly evaluates the new consumption against a limit when the quantity is amended.

Round Price Application Program Interface

This section explains how to use the round_price API and how it functions in Oracle Advanced Pricing.

The round_price API is used to round the price using the rounding factor. The package QP_UTIL_PUB contains the procedure round_price.

Functional Overview

This API is called by pricing engine as well as directly from OM.

When it is called from pricing engine, the pricing engine passes the rounding factor and the operand (i.e. the price of an item) and so this API just uses the round() function and return the rounded operand.

But when this API is called directly from OM (while re-pricing an item), it passes the price_list_id and the operand. In this case, this API finds the rounding factor associated with price list, round the passed operand and then return the rounded operand.

The parameter p_operand_type can take 3 different values—A, S, or R, with the default value S. When R is passed for p_operand_type then rounding_factor is returned in the out parameter p_rounded_operand, else rounded value (after applying rounding factor) of p_operand is returned.

The rounded operand / rounding factor is returned only if one of the following conditions is met :

- p_operand_type = A and profile "QP: Selling Price Rounding Options" = 'ROUND_ADJ'

- p_operand_type = S and profile "QP: Selling Price Rounding Options" = 'ROUND_ADJ' or 'NO_ROUND_ADJ'
- p_operand_type = R
- profile OE_UNIT_PRICE_ROUNDING = Y

When profile options QP: Multi Currency Installed and QP: Multi Currency Usage are set to Y (Yes), then the rounding factor is retrieved by joining the tables qp_list_headers_b and qp_currency_details for the passed values of price list id, currency code and effective date; otherwise, the rounding factor is retrieved from table qp_list_headers_b for the price list id (regardless of currency or effective date).

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the round_price API. All of the inbound and outbound parameters are listed.

round_price Parameters

Parameter	Usage	Type	Req	Drv
p_operand	IN	Number	Yes, if the p_operand_type is not 'R'	No
p_rounding_fact or	IN	Number	No	No
p_use_multi_cur rency	IN	Varchar2	No	No
p_price_list_id	IN	Number	No	No
p_currency_code	IN	Varchar2	No	No
p_pricing_effecti ve_date	IN	Date	No	No
x_rounded_oper and	OUT	Number	No	No
x_status_code	OUT	Varchar2	No	No

Parameter	Usage	Type	Req	Drv
p_operand_type	IN	Varchar2	No	No

Validation of Round Price API

Validations:

Standard Validation

The caller is responsible for passing the right parameters to Round_Price.

For specific information on the data implied by these columns, See *Oracle Pricing Technical Reference Manual* for details.

Other Validation

None.

Error Handling

If any exception occurs in the Round_Price API, it returns appropriate status code as below.

Status Code

Condition	x_status_code
Success	FND_API.G_RET_STS_SUCCE
Expected Error	FND_API.G_RET_STS_ERROR
Unexpected Error	FND_API.G_RET_STS_UNEXP_ERROR

Note: The Package Specification and Body files are QPXRTCNS.pls and QPXRTCNB.pls and are available under the source control directory \$QP_TOP/patch/115/sql.

Validate Price List with Currency Code Application Program Interface

This section explains how to use the Validate Price List with Currency Code API (Validate_Price_list_Curr_code) and how it functions in Oracle Advanced Pricing.

The Validate Price List with Currency Code API is used to validate a given price list with a currency code.

Functional Overview

When an order is processed using a particular price list and currency code, the combination of price list and currency code must be verified to ensure it is correct for the pricing effective date. If the pricing effective date is not passed by the calling application, the current date defaults. If the profile options QP: Multi Currency Installed and QP: Multi Currency Usage are Y (Yes) then validation is done by joining the tables qp_list_headers_b and qp_currency_details else only table qp_list_headers_b is used.

Setting Up and Parameter Descriptions

The following chart describes all parameters used by the Validate Price List with Currency Code API. All of the inbound and outbound parameters are listed.

Validate_Price_list_Curr_code parameters

Parameter	Usage	Type	Req	Drv
l_price_list_id	IN	Number	Yes	No
l_currency_code	IN	Varchar2	Yes	No
l_pricing_effective_date	IN	Date	No	No
l_validate_result	OUT	Varchar2	No	No

Validation of Validate Price List with Currency Code API

Validations:

Standard Validation

The caller is responsible for passing the correct parameters to Validate_Price_list_Curr_code.

Other Validation

None.

Error Handling

If any exception occurs in the Validate_Price_list_Curr_code API, it returns the value "N" (No) in outbound parameter l_validate_result.

Note: The package specification and body files are QPXRTCNS.pls and QPXRTCNB.pls and are available under the source control directory

\$QP_TOP/patch/115/sql.

Oracle Advanced Pricing XML Transactions

Get Catalog Transaction

This section explains how to use the Get Catalog XML transaction (GET_CATALOG) and how it functions in Oracle Advanced Pricing. This XML transaction is based on the XML messaging standards defined by the Open Applications Group (OAG).

Get Catalog is an inbound message used between a seller and a buyer to generate price books via XML messaging. In a typical scenario, the buyer sends a GET_CATALOG message to the seller, and the seller then returns a SYNC_CATALOG message containing the price book details. This process can be set up to automatically request and receive generated price books detailing the pricing of a particular set of products. The following table provides general information on the GET_CATALOG message.

Message Map Name	QP_CATGI_OAG72_IN
Direction	Inbound
(Internal) Transaction Type	QP
(Internal) Transaction Subtype	CATGI
External Transaction Type	CATALOG
External Transaction Subtype	GET
DTD Directory	xml/oag72

Message Map Name	QP_CATGI_OAG72_IN
Map Directory	patch/115/xml/US
Message Maps XGM File Name	QP_CATGI_OAG72_IN.xgm
Standard	OAG
Release	7.2
Format	DTD
DTD Name	129_get_catalog_002.dtd

Setting up and Parameter Descriptions

The following table describes all inbound parameters used by GET_CATALOG. The abbreviations used in this table are described below:

Abbreviation	Meaning
HB	QP_PB_INPUT_HEADERS_B
HTL	QP_PB_INPUT_HEADERS_TL
L	QP_PB_INPUT_LINES

OAG Element	Description/Comment	Oracle QP Table Column*	Required
<CNTROLAR EA>	The fields included in this area provide information about the XML document such as BSR, SENDER and DATETIME (described below).	N/A	Yes
<BSR>	Shows the Business Service Request name per OAG.	N/A	Yes

OAG Element	Description/Comment	Oracle QP Table Column*	Required
<VERB/>	Value is GET.	Oracle XML Gateway tables	Yes
<NOUN/>	Value is CATALOG.	Oracle XML Gateway tables	Yes
<REVISION/>	Value is 002.	Oracle XML Gateway tables	Yes
</BSR>	--	--	Yes
<SENDER>	Provides information on the system that sends the document.	N/A	Yes
<LOGICALID/>	Sender system identifier.	Oracle XML Gateway tables	Yes
<COMPONENT/>	Sender application name.	Oracle XML Gateway tables	Yes
<TASK/>	Event or Action. Value is CATGL.	Oracle XML Gateway tables	Yes
<REFERENCE ID/>	Unique reference ID for this doc.	Oracle XML Gateway tables	Yes
<CONFIRMATION/>	Confirmation when doc is received. Value is '0' means none required.	Oracle XML Gateway tables	Yes
<LANGUAGE/>	ISO language in which the text fields are transmitted, for example, US.	Oracle XML Gateway tables	Yes

OAG Element	Description/Comment	Oracle QP Table Column*	Required
<CODEPAGE/>	Character set used in this XML doc, for example, UTF8.	Oracle XML Gateway tables	Yes
<AUTHID/>	System ID of sender. Value is "APPS".	Oracle XML Gateway tables	Yes
</SENDER>	--	--	Yes
<DATETIME/>	Creation date and time of the XML document in OAG date format.	Oracle XML Gateway tables	Yes
</CNTROLA REA>	--	N/A	Yes
<DATAAREA>	The fields included in this area provide information about the data included in the XML document.	N/A	Yes
<GET_CATALOG>	Exactly one instance is required within a message.	N/A	Yes
<CATALOG>	The fields included in this area provide price book input information. Exactly one instance is required within a message.	N/A	Yes
<DATETIME/>	Pricing effective date and time in OAG date format.	HB.EFFECTIVE_DATE	Yes
<CATALGNAME/>	Name of the price book.	HTL.PRICE_BOOK_NAME	Yes
<USERAREA>	The fields included in this area are provided by Oracle.	N/A	Yes
<CURRENCY CODE/>	Currency code.	HB.CURRENCY_CODE	If PBTTYPECODE is "F"

OAG Element	Description/Comment	Oracle QP Table Column*	Required
<ITEMQUANTITY/>	Item quantity to be used for all items in the price book.	HB.ITEM_QUANTITY	No, defaulted
<PUBLISHEXISTINGPBFLAG/>	Allows requestor to request an existing price book identified by the price book name (CATALGNAME) and the pricing effective date (DATETIME). Valid values are "Y" and "N."	HB.PUBLISH_EXISTING_PB_FLAG	No, defaulted
<OVERWRITEEXISTINGPBFFLAG/>	Allows requestor to overwrite an existing price book. Valid Values are "Y" and "N."	HB.OVERWRITE_EXISTING_PB_FLAG	No, defaulted
<PBTYPECODE/>	Price book type. Valid values are <i>F</i> and <i>D</i> .	HB.PRICE_BOOK_TYPE_CODE	No, defaulted
<LIMITPRODUCTSBYCODE/>	Product filter type. Valid values are ITEM, ITEM_CATEGORY, ALL_ITEMS, and PRICE_LIST.	HB.LIMIT_PRODUCTS_BY	If PBTYPECODE is "F"
<PRICEBASEDONCODE/>	Entity type for which the pricing is based on, namely the pricing basis. Valid values are PRICE_LIST, AGREEMENT, and BSA.	HB.PRICE_BASED_ON	If PBTYPECODE is "F"
<PRICELIST/> or <PRICELISTID/>	Price list name or ID referred to by either LIMITPRODUCTSBYCODE and/or PRICEBASEDONCODE.	HB.PL_AGR_BSA_ID	If LIMITPRODUCTSBYCODE or PRICEBASEDONCODE is "PRICE_LIST"
<AGREEMENT/> or <AGREEMENTID/>	Agreement name or ID referred to by PRICEBASEDONCODE.	HB.PL_AGR_BSA_ID	If PRICEBASEDONCODE is "AGREEMENT"

OAG Element	Description/Comment	Oracle QP Table Column*	Required
<BSA/> or <BSAID/>	Sales agreement name or ID referred to by PRICEBASEDONCODE.	HB.PL_AGR_ BSA_ID	If PRICEBASED ONCODE is " BSA"
<ITEMX/> or <ITEMID/>	Internal item number or ID referred to by LIMITPRODUCTSBYCODE.	HB.PRODUC T_ATTR_VAL UE	If LIMITPRODU CTSBYCODE is "ITEM"
<ITEMCATEG ORY/> or <ITEMCATEG ORYID/>	Internal item category name or ID referred to by LIMITPRODUCTSBYCODE.	HB.PRODUC T_ATTR_VAL UE	If LIMITPRODU CTSBYCODE is " ITEM_CATEG ORY"
<GENDATET IME>	The fields included in this area describe when the price book will be generated.	N/A	No
<GENTIMEC ODE/>	Dictates whether the price book will be generated immediately or at a specific time. Valid values are IMMEDIATE and SCHEDULE.	HB.GENERAT ION_TIME_C ODE	No, defaulted
<DATETIME/ >	Price book generation date and time in OAG date format. Used when GENTIMECODE is SCHEDULE.	HB.GEN_SCH EDULE_DAT E	If GENTIMECO DE is " SCHEDULE"
</GENDATE TIME>	--	N/A	No
<PBATTR>	The fields included in this area describe a price book input attribute.		No
<CTXT/> or <CTXTCODE/ >	Context name such as Terms or code such as TERMS.	L.CONTEXT	If PBATTR is specified

OAG Element	Description/Comment	Oracle QP Table Column*	Required
<ATTR/> or <ATTRCODE/ >	Attribute name such as Payment Terms or code such as QUALIFIER_ATTRIBUTE1.	L.ATTRIBUTE	If PBATTR is specified
<ATTRVALUE/> or <ATTRVALUECODE/>	Attribute value name such as "2 Net" or code "1145."	L.ATTRIBUTE_VALUE	If PBATTR is specified
<ATTRTYPE/>	Attribute type. Valid values are PRICING_ATTRIBUTE and QUALIFIER.	L.ATTRIBUTE_TYPE	If PBATTR is specified
</PBATTR>	--	N/A	No
</USERAREA>	--	N/A	Yes
<PARTNER>	The fields included in this area describe the customer for which the price book is created.		No
<USERAREA>	The fields included in this area are provided by Oracle.	N/A	No
<CUSTACCT/>	Internal customer account ID.	HB.CUST_AC COUNT_ID	No, defaulted.
</USERAREA>	--	N/A	No
</PARTNER>	--	N/A	No
</CATALOG>	--	N/A	Yes
</GET_CATALOG>	--	N/A	Yes
</DATAAREA>	--	N/A	Yes

The following table lists the defaulted table column values.

Oracle QP Table Column	Value
HB.REQUEST_ORIGIN ATION_CODE	XML
HB.CUSTOMER_CON TEXT	CUSTOMER
HB.CUSTOMER_ATTR IBUTE	QUALIFIER_ATTRIBUTE2
HB.GENERATION_TI ME_CODE	IMMEDIATE
HB.ITEM_QUANTITY	1
HB.OVERWRITE_EXIS TING_PB_FLAG	N
HB.PRICE_BOOK_TYP E_CODE	F
HB.PRODUCT_CONTE XT	ITEM
HB.PUBLISH_EXISTIN G_PB_FLAG	N
HB.DLV_XML_FLAG	Y
HB.CUST_ACCOUNT_ ID	If there is exactly one customer account associated with the customer, then this customer account will be used.

The following table lists the derived table column values.

Oracle QP Table Column	Derivation
HB.CREATED_BY	FND_GLOBAL.USER_ID
HB.CREATION_DATE	SYSDATE

Oracle QP Table Column	Derivation
HB.LAST_UPDATED_BY	FND_GLOBAL.USER_ID
HB.LAST_UPDATE_DATE	SYSDATE
HB.LAST_UPDATE_LOGIN	FND_GLOBAL.LOGIN_ID
HB.CUSTOMER_ATTR_VALUE	Customer defined in trading partner setup
HB.DLV_XML_SITE_ID	Party site ID defined in trading partner setup
HB.ORG_ID	Provided by XML Gateway.
HB.PB_INPUT_HEADER_ID	Next value from sequence QP_PB_INPUT_HEADERS_B_S
HB.PRICE_BASED_ON	"PRICE_LIST" if B.LIMIT_PRODUCTS_BY is "PRICE_LIST"
HB.PRICING_PERSPECTIVE_CODE	Value from profile QP: External Default Pricing Perspective
HB.REQUEST_TYPE_CODE	Value from profile QP: Pricing Perspective Request Type
HB.PRODUCT_ATTRIBUTE	Assigns appropriate attribute based on LIMITPRODUCTSBYCODE.
HB.PRODUCT_ATTR_VALUE	Assigns appropriate attribute value based on LIMITPRODUCTSBYCODE, ITEMX, ITEMID, ITEMCATEGORY, ITEMCATEGORYID
HB.PL_AGR_BSA_ID	Assigns appropriate list ID based on PRICEBASEDONCODE, PRICELIST, PRICELISTID, AGREEMENT, AGREEMENTID, BSA, BSAID.

Oracle QP Table Column	Derivation
TL.PL_AGR_BSA_NAME	Assigns appropriate list name based on PRICEBASEDONCODE, PRICELIST, PRICELISTID, AGREEMENT, AGREEMENTID, BSA, BSAID.
L.CREATED_BY	FND_GLOBAL.USER_ID
L.CREATION_DATE	SYSDATE
L.LAST_UPDATED_BY	FND_GLOBAL.USER_ID
L.LAST_UPDATE_DATE	SYSDATE
L.LAST_UPDATE_LOGIN	FND_GLOBAL.LOGIN_ID
L.PB_INPUT_HEADER_ID	HB.PB_INPUT_HEADER_ID
L.PB_INPUT_LINE_ID	Next value from sequence QP_PB_INPUT_LINES_S
L.CONTEXT	Corresponding context code in the case where CTXT is provided.
L.ATTRIBUTE	Corresponding attribute code in the case where ATTR is provided.

The following code shows the structure of the GET_CATALOG message. Although not all tags will be necessary to construct a valid message, you must preserve the order of the tag structure.

```

<GET_CATALOG_002>
  <CNTROLAREA>
    <BSR>
      <VERB value="GET"/>
      <NOUN value="CATALOG"/>
      <REVISION value="002"/>
    </BSR>
    <SENDER>
      <LOGICALID/>
      <COMPONENT/>
      <TASK/>
      <REFERENCEID/>
      <CONFIRMATION/>
      <LANGUAGE/>
      <CODEPAGE/>
      <AUTHID/>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR/>
      <MONTH/>
      <DAY/>
      <HOUR/>
      <MINUTE/>
      <SECOND/>
      <SUBSECOND/>
      <TIMEZONE/>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <GET_CATALOG>
      <CATALOG returndata="1">
        <DATETIME qualifier="EFFECTIVE">
          <YEAR/>
          <MONTH/>
          <DAY/>
          <HOUR/>
          <MINUTE/>
          <SECOND/>
          <SUBSECOND/>
          <TIMEZONE/>
        </DATETIME>
        <CATALGNAME/>
        <USERAREA>
          <CURRENCYCODE/>
          <ITEMQUANTITY/>
          <PUBLISHEXISTINGPBFLAG/>
          <OVERWRITEEXISTINGPBFLAG/>
          <PBTYPECODE/>
          <LIMITPRODUCTSBYCODE/>
          <PRICEBASEDONCODE/>
          <PRICELIST/>
          <PRICELISTID/>
          <AGREEMENT/>
          <AGREEMENTID/>
          <BSA/>
          <BSAID/>
          <ITEM/>
          <ITEMX/>
          <ITEMID/>
          <ITEMCATEGORY/>
          <ITEMCATEGORYID/>

```

```

<GENDATETIME>
  <GENTIMECODE/>
  <DATETIME qualifier="GENERATION">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
  </DATETIME>
</GENDATETIME>
<PBATTR>
  <CTXT/>
  <CTXTCODE/>
  <ATTR/>
  <ATTRCODE/>
  <ATTRVALUE/>
  <ATTRVALUECODE/>
  <ATTRTYPE/>
</PBATTR>
</USERAREA>
<PARTNER>
  <USERAREA>
    <CUSTACCT\>
  </USERAREA>
</PARTNER>
</CATALOG>
</GET_CATALOG>
</DATAAREA>
</GET_CATALOG_002>

```

Trading Partner Setup

The GET_CATALOG transaction uses Oracle XML Gateway to facilitate the incoming messages. To use this transaction, you must define a trading partner and enable the GET_CATALOG message for this partner in the Oracle XML Gateway Define Trading Partner Setup form. The relevant message for this transaction has transaction type QP and transaction subtype CATGI. For more information, see the *Oracle XML Gateway User's Guide*.

Validation of Get Catalog Transaction

Standard Validation

Oracle XML Gateway first validates the GET_CATALOG message structure, and ensures that the payload follows the DTD as specified by OAG. Oracle Advanced Pricing then validates all required table columns using the same validation in the Get Catalog transaction as that of the Create Publish Price Book API. The Get Catalog transaction is available only to external users and not to internal users. For more information, see validation section for Create Publish Price Book API.

Error Handling

If any validation fails, the error will be returned to Oracle Workflow. An e-mail notification will be sent to both the Trading Partner and System Administrator contact containing the error code and message. Users can also view error messages using the

Oracle XML Gateway Transaction Monitor and/or Oracle Workflow Status Monitor. For more information, please refer to the documentation for these modules. Note that when appropriate, error messages may refer to the table column names, and not the corresponding tag names.

Examples of the Get Catalog Transaction

Request a New Price Book

The following example describes a standard request for a new price book with the following characteristics:

- The price book is named *2006 - AS54999* effective 10/04/2006.
- The price book contains only item *AS54999*.
- The price book is based on price list *Computer Desktops and Accessories*.
- The price book is scheduled to be generated 10/04/2005.
- The order category is passed with value ORDER.
- The payment terms is passed with value 2 *Net*. Since the attribute code is known, the requestor has specified the ATTRCODE as QUALIFIER_ATTRIBUTE1.

```

<GET_CATALOG_002>
  <CNTROLAREA>
    <BSR>
      <VERB value="GET"/>
      <NOUN value="CATALOG"/>
      <REVISION value="002"/>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>QP</COMPONENT>
      <TASK>CATGI</TASK>
      <REFERENCEID>112233</REFERENCEID>
      <CONFIRMATION>1</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2000</YEAR>
      <MONTH>10</MONTH>
      <DAY>15</DAY>
      <HOUR>16</HOUR>
      <MINUTE>59</MINUTE>
      <SECOND>59</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>-0500</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <GET_CATALOG>
      <CATALOG returndata="1">
        <DATETIME qualifier="EFFECTIVE">
          <YEAR>2006</YEAR>
          <MONTH>10</MONTH>
          <DAY>04</DAY>
          <HOUR>15</HOUR>
          <MINUTE>30</MINUTE>
          <SECOND>00</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>-0000</TIMEZONE>
        </DATETIME>
        <CATALGNAME>2006 - AS54999</CATALGNAME>
        <USERAREA>
          <CURRENCYCODE>USD</CURRENCYCODE>
          <ITEMQUANTITY>5</ITEMQUANTITY>
          <PUBLISHEXISTINGPBFLAG>N</PUBLISHEXISTINGPBFLAG>
          <OVERWRITEEXISTINGPBFLAG>N</OVERWRITEEXISTINGPBFLAG>
          <PBTYPECODE>F</PBTYPECODE>
          <LIMITPRODUCTSBYCODE>ITEM</LIMITPRODUCTSBYCODE>
          <PRICEBASEDONCODE>PRICE_LIST</PRICEBASEDONCODE>
          <PRICELIST>Computer Desktops and Accessories</PRICELIST>
          <ITEMX>AS54999</ITEMX>
          <GENDATETIME>
            <GENTIMECODE>SCHEDULE</GENTIMECODE>
            <DATETIME qualifier="GENERATION">
              <YEAR>2005</YEAR>
              <MONTH>10</MONTH>
              <DAY>04</DAY>
              <HOUR>16</HOUR>
              <MINUTE>10</MINUTE>
              <SECOND>00</SECOND>
            </DATETIME>
          </GENDATETIME>
        </USERAREA>
      </CATALOG>
    </GET_CATALOG>
  </DATAAREA>
</GET_CATALOG_002>

```



```

<SUBSECOND>0000</SUBSECOND>
  <TIMEZONE>-0000</TIMEZONE>
  </DATETIME>
</GENDATETIME>
<PBATTR>
  <CTXTCODE>ORDER</CTXTCODE>
  <ATTR>Order Category</ATTR>
  <ATTRVALUE>ORDER</ATTRVALUE>
  <ATTRTYPE>QUALIFIER</ATTRTYPE>
</PBATTR>
<PBATTR>
  <CTXTCODE>TERMS</CTXTCODE>
  <ATTRCODE>QUALIFIER_ATTRIBUTE1</ATTRCODE>
  <ATTRVALUE>2 Net</ATTRVALUE>
  <ATTRTYPE>QUALIFIER</ATTRTYPE>
</PBATTR>
</USERAREA>
</CATALOG>
</GET_CATALOG>
</DATAAREA>
</GET_CATALOG_002>

```

Republish a Price Book

The example below displays a standard request to re-send an existing price book. The desired price book is identified by the price book type, the price book name, and the customer (which is derived from the trading partner setup).

```

<GET_CATALOG_002>
  <CNTROLAREA>
    <BSR>
      <VERB value="GET"/>
      <NOUN value="CATALOG"/>
      <REVISION value="002"/>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>QP</COMPONENT>
      <TASK>CATGI</TASK>
      <REFERENCEID>112233</REFERENCEID>
      <CONFIRMATION>1</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2000</YEAR>
      <MONTH>10</MONTH>
      <DAY>15</DAY>
      <HOUR>16</HOUR>
      <MINUTE>59</MINUTE>
      <SECOND>59</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>-0500</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <GET_CATALOG>
      <CATALOG returndata="1">
        <CATALGNAME>2006 - AS54999</CATALGNAME>
        <USERAREA>
          <PUBLISHEXISTINGPBFLAG>Y</PUBLISHEXISTINGPBFLAG>
          <OVERWRITEEXISTINGPBFLAG>N</OVERWRITEEXISTINGPBFLAG>
          <PBTYPECODE>F</PBTYPECODE>
        </USERAREA>
      </CATALOG>
    </GET_CATALOG>
  </DATAAREA>
</GET_CATALOG_002>

```

Request a Delta Price Book

This payload example describes a standard request for a delta price book. Note that the full price book, from which the delta is based, is identified by the price book type, the price book name, and the customer (which is derived from the trading partner setup).

Note: The Effective Date that is used in a Delta Price Book request can be different from (typically later than) the pricing effective date of the corresponding full price book with the same price book name and customer.

```

<GET_CATALOG_002>
  <CNTROLAREA>
    <BSR>
      <VERB value="GET"/>
      <NOUN value="CATALOG"/>
      <REVISION value="002"/>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>QP</COMPONENT>
      <TASK>CATGI</TASK>
      <REFERENCEID>112233</REFERENCEID>
      <CONFIRMATION>1</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2000</YEAR>
      <MONTH>10</MONTH>
      <DAY>15</DAY>
      <HOUR>16</HOUR>
      <MINUTE>59</MINUTE>
      <SECOND>59</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>-0500</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <GET_CATALOG>
      <CATALOG returndata="1">
        <DATETIME qualifier="EFFECTIVE">
          <YEAR>2006</YEAR>
          <MONTH>10</MONTH>
          <DAY>04</DAY>
          <HOUR>15</HOUR>
          <MINUTE>30</MINUTE>
          <SECOND>00</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>-0000</TIMEZONE>
        </DATETIME>
        <CATALGNAME>2006 - AS54999</CATALGNAME>
        <USERAREA>
          <PUBLISHEXISTINGPBFLAG>N</PUBLISHEXISTINGPBFLAG>
          <OVERWRITEEXISTINGPBFLAG>N</OVERWRITEEXISTINGPBFLAG>
          <PBTYPECODE>D</PBTYPECODE>
        </USERAREA>
      </CATALOG>
    </GET_CATALOG>
  </DATAAREA>
</GET_CATALOG_002>

```

Sync Catalog Transaction

This section explains how to use the Sync Catalog XML transaction (SYNC_CATALOG) and how it functions in Oracle Advanced Pricing. This XML transaction is based on XML messaging standards defined by the Open Applications Group (OAG).

SYNC_CATALOG is an outbound message used between seller and buyer to send price

books via XML messaging. This message is generated and sent in response to a GET_CATALOG message (described in the preceding section). The following table provides general information on the SYNC_CATALOG message:

Message Map Name	QP_CATSO_OAG72_OUT
Direction	Outbound
(Internal) Transaction Type	QP
(Internal) Transaction Subtype	CATSO
External Transaction Type	CATALOG
External Transaction Subtype	SYNC
DTD Directory	xml/oag72
Map Directory	patch/115/xml/US
Message Maps XGM File Name	QP_CATSO_OAG72_OUT.xgm
Standard	OAG
Release	7.2
Format	DTD
DTD Name	128_sync_catalog_003.dtd

Setting Up and Parameter Descriptions

The parameters table describes all outbound parameters used by SYNC_CATALOG. The following abbreviations are used in the table:

Abbreviation	Meaning
IHV	QP_PB_INPUT_HEADERS_V
IHTL	QP_PB_INPUT_HEADERS_TL

Abbreviation	Meaning
ILV	QP_PB_INPUT_LINES_V
PBHV	QP_PRICE_BOOK_HEADERS_V
PBMV	QP_PRICE_BOOK_MESSAGES_V
PBLV	QP_PRICE_BOOK_LINES_V
PBAV	QP_PRICE_BOOK_ATTRIBUTES_V
PBLDV	QP_PRICE_BOOK_LINE_DETAILS_V
PBBLV	QP_PRICE_BOOK_BREAK_LINES_V

The following table lists the outbound parameters used by SYNC_CATALOG:

Outbound parameters in SYNC_CATALOG

OAG Element	Description/Comment	Oracle QP Table.View Column*
<CNTROLAREA>	The fields included in this area provide information about the XML document such as BSR, SENDER, and DATETIME described below.	N/A
<BSR>	Shows the Business Service Request name per OAGI.	N/A
<VERB/>	Value is SYNC	Oracle XML Gateway
<NOUN/>	Value is CATALOG	Oracle XML Gateway
<REVISION/>	Value is 003	Oracle XML Gateway
</BSR>	--	--

OAG Element	Description/Comment	Oracle QP Table.View Column*
<SENDER>	Provides information on the system that sends the document.	N/A
<LOGICALID/>	Sender system identifier.	Oracle XML Gateway
<COMPONENT/>	Sender application name.	Oracle XML Gateway
<TASK/>	Event or Action. Value is CATSO.	Oracle XML Gateway
<REFERENCEID/>	Unique reference ID for this doc.	Oracle XML Gateway
<CONFIRMATION/>	Confirmation when doc is received. Value is 0 means none required.	Oracle XML Gateway
<LANGUAGE/>	ISO language in which the text fields are transmitted, for example, US.	Oracle XML Gateway
<CODEPAGE/>	Character set used in this XML doc, for example, UTF8.	Oracle XML Gateway
<AUTHID/>	System ID of sender. Value is APPS.	Oracle XML Gateway
</SENDER>	--	--
<DATETIME/>	Creation date and time of the XML document in OAG date format.	Oracle XML Gateway
</CNTROLAREA>	--	N/A
<DATAAREA>	The fields included in this area provide information about the data included in the XML document.	N/A

OAG Element	Description/Comment	Oracle QP Table.View Column*
<SYNC_CATALOG>	Exactly one instance is required within a message.	N/A
<CATALOG>	The fields included in this area provide price book information. Exactly one instance is required within a message.	N/A
<DATETIME/>	Pricing effective date and time in OAG date format.	PBHV.EFFECTIVE_DATE
<CATALGNAME/>	Name of the price book.	PBHV.PRICE_BOOK_NAME
<USERAREA>	The fields included in this area are provided by Oracle.	N/A
<PBTYPECODE/>	Price book type. Valid values are F and D.	PBHV.PRICE_BOOK_TYPE_CODE
<CURRENCYCODE/>	Currency code	PBHV.CURRENCY_CODE
<ITEMQUANTITY/>	Item quantity to be used for all items in the price book.	PBHV.ITEM_QUANTITY
<LIMITPRODUCTSBYCODE/>	Product filter type. Valid values are ITEM, ITEM_CATEGORY, ALL_ITEMS, and PRICE_LIST.	IHV.LIMIT_PRODUCTS_BY
<ITEM/>	Internal item name referred to by LIMITPRODUCTSBYCODE..	IHV.PRODUCT_NAME
<ITEMCATEGORY/>	Internal item category name referred to by LIMITPRODUCTSBYCODE.	IHV.PRODUCT_NAME
<PRICELIST/>	Price list name referred to by either LIMITPRODUCTSBYCODE and/orEBASEDONCODE.	IHTL.PL_AGR_BSA_NAME

OAG Element	Description/Comment	Oracle QP Table.View Column*
<PRICEBASEDONCODE/>	Entity type for which the pricing is based on, namely the pricing basis. Valid values are PRICE_LIST, AGREEMENT, and BSA.	IHV.PRICE_BASED_ON
<AGREEMENT/>	Agreement name referred to by PRICEBASEDONCODE.	IHTL.PL_AGR_BSA_NAME
<BSA/>	Sales Agreement Number referred to by PRICEBASEDONCODE.	IHTL.PL_AGR_BSA_NAME
<PBATTR>	The fields included in this area describe a price book input attribute.	N/A
<CTXT/>	Context name of attribute.	ILV.CONTEXT
<ATTR/>	Attribute name	ILV.ATTRIBUTE
<ATTRVALUE/>	Attribute value name such as "2 Net."	ILV.ATTRIBUTE_VALUE
<ATTRTYPE/>	Attribute type. Valid values are PRICING_ATTRIBUTE and QUALIFIER_ATTRIBUTE.	ILV.ATTRIBUTE_TYPE
</PBATTR>	--	N/A
<PBHEADERMSG>	The fields included in this area describe a price book header error message.	N/A
<TYPE/>	Message type	PBMV.MESSAGE_TYPE
<CODE/>	Message code	PBMV.MESSAGE_CODE
<TEXT/>	Message text	PBMV.MESSAGE_TEXT
</PBHEADERMSG>	--	N/A

OAG Element	Description/Comment	Oracle QP Table.View Column*
<PARTNER>	The fields included in this area describe the customer for which the price book has been created.	N/A
<NAME/>	Customer name	PBHV.CUSTOMER_NAME
<PARTNRID/>	Customer ID	PBHV.CUSTOMER_ID
<USERAREA>	The fields included in this area are provided by Oracle.	N/A
<CUSTACCT/>	Internal customer account ID	PBHV.CUST_ACCOUNT_ID
</USERAREA>	--	--
</PARTNER>	--	N/A
<CATITEM>	The fields included in this area describe a price book line.	N/A
<ITEM/>	Internal item number	PBLV.ITEM_NUMBER
<SYNCIND/>	Sync action indicator	PBLV.SYNC_ACTION_CODE
<DESCRIPTN/>	Internal item description	PBLV.DESCRPTION
<USERAREA>	The fields included in this area are provided by Oracle.	N/A
<LISTPRICE/>	List price	PBLV.LIST_PRICE
<UOMCODE/>	UOM code	PBLV.PRODUCT_UOM_CODE
<NETPRICE/>	Net price	PBLV.NET_PRICE
<LINESTATUSCODE/>	Line status code	PBLV.LINE_STATUS_CODE
<ITEMCAT>	The fields included in this area describe a category in which the item belongs.	N/A

OAG Element	Description/Comment	Oracle QP Table.View Column*
<ITEMCATEGORY/>	Item category name	PBAV.ATTRIBUTE_VALUE_NAME
</ITEMCAT>	--	N/A
<PBLINMSG>	The fields included in this area describe a price book line error message.	N/A
<TYPE/>	Message type	PBMV.MESSAGE_TYPE
<CODE/>	Message code	PBMV.MESSAGE_CODE
<TEXT/>	Message text	PBMV.MESSAGE_TEXT
</PBLINMSG>	--	N/A
<PBLINEDetail>	The fields included in this area describe a price book line detail (such as price list lines and modifiers).	N/A
<DETAILTYPE/>	Line type	PBLDV.LIST_LINE_TYPE
<BREAKTYPE/>	Price break type	PBLDV.PRICE_BREAK_TYPE
<LISTPRICE/>	List price	PBLDV.LIST_PRICE
<LISTNAME/>	Name of either price list or modifier list	PBLDV.LIST_NAME
<MODLINENO/>	Modifier number	PBLDV.LIST_LINE_NO
<MODOPERAND/>	Modifier operand	PBLDV.MODIFIER_OPERAND
<MODAPPMETHOD/>	Modifier application method	PBLDV.MODIFIER_APPLICATION_METHOD
<ADJAMT/>	Modifier adjustment amount	PBLDV.ADJUSTMENT_AMOUNT

OAG Element	Description/Comment	Oracle QP Table.View Column*
<CNETPRICE/>	Cumulative net price	PBLDV.ADJUSTED_NET_PRICE
<PBLINEATTR>	The fields included in this area describe an attribute attached to the price list line.	N/A
<CTXT/>	Context name of attribute	PBAV.CONTEXT_NAME
<ATTR/>	Attribute name	PBAV.ATTRIBUTE_NAME
<ATTRVALFROM/>	Attribute FROM value Note: This does not correspond to the meaning or description of the value, but rather the value or ID.	PBAV.ATTRIBUTE_VALUE_NAME
<ATTRVALTO/>	Attribute TO value	PBAV.ATTRIBUTE_VALUE_TO_NAME
<ATTRTYPE/>	Attribute type. Valid value is PRICING_ATTRIBUTE.	PBAV.PRICING_PROD_ATTR_DATATYPE
</PBLINEATTR>	--	N/A
<PBLINEBRK>	The fields included in this area describe the price breaks defined for the price book line detail.	N/A
<CTXT/>	Context name of attribute	PBBLV.CONTEXT_NAME
<ATTR/>	Attribute name	PBBLV.ATTRIBUTE_NAME

OAG Element	Description/Comment	Oracle QP Table.View Column*
<ATTRVALFROM/>	Attribute FROM value Note: This does not correspond to the meaning or description of the value, but rather the value or ID.	PBBLV.ATTRIBUTE_VALUE_NAME
<ATTRVALTO/>	Attribute TO value	PBBLV.ATTRIBUTE_VALUE_TO_NAME
<ATTRTYPE/>	Attribute type. Valid value is PRICING_ATTRIBUTE.	PBBLV.PRICING_ATTRIBUTE_DATATYPE
<OPERAND/>	Price break line operand	PBBLV.OPERAND
</PBLINEBRK>	--	N/A
</PBLINEDetail>	--	N/A
</USERAREA>	--	N/A
</CATITEM>	--	--
</CATALOG>	--	N/A
</GET_CATALOG>	--	N/A
</DATAAREA>	--	N/A

The following displays the structure of the SYNC_CATALOG message:

```

<SYNC_CATALOG_003>
  <CNTROLAREA>
    <BSR>
      <VERB value="SYNC"/>
      <NOUN value="CATALOG"/>
      <REVISION value="003"/>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>QP</COMPONENT>
      <TASK>CATSO</TASK>
      <REFERENCEID/>
      <CONFIRMATION/>
      <LANGUAGE/>
      <CODEPAGE/>
      <AUTHID/>
    </SENDER>
    <DATETIME qualifier="CREATION" type="T" index="1">
      <YEAR>2005</YEAR>
      <MONTH>12</MONTH>
      <DAY>13</DAY>
      <HOUR>16</HOUR>
      <MINUTE>35</MINUTE>
      <SECOND>49</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SYNC_CATALOG>
      <CATALOG>
        <DATETIME qualifier="EFFECTIVE" type="T" index="1">
          <YEAR/>
          <MONTH/>
          <DAY/>
          <HOUR/>
          <MINUTE/>
          <SECOND/>
          <SUBSECOND/>
          <TIMEZONE/>
        </DATETIME>
        <CATALGNAME/>
        <SYNCIND/>
        <CATALOGRV/>
        <CLSSSCHMID/>
        <DESCRIPTN/>
        <USERAREA>
          <PBTYPECODE/>
          <CURRENCYCODE/>
          <ITEMQUANTITY/>
          <LIMITPRODUCTSBYCODE/>
          <ITEM/>
          <ITEMCATEGORY/>
          <PRICELIST/>
          <PRICEBASEDONCODE/>
          <AGREEMENT/>
          <BSA/>
          <PBATTR>
            <CTXT/>
            <ATTR/>
            <ATTRVALUE/>
          </PBATTR>
        </USERAREA>
      </CATALOG>
    </SYNC_CATALOG>
  </DATAAREA>
</SYNC_CATALOG_003>

```

```

<ATTRTYPE/>
    </PBATTR>
    <PBHEADERMSG>
        <TYPE/>
        <CODE/>
        <TEXT/>
    </PBHEADERMSG>
</USERAREA>
<PARTNER>
    <NAME index="1"/>
    <PARTNRID/>
    <PARTNRTYPE/>
    <PARTNRIDX/>
    <USERAREA>
        <CUSTACCT/>
    </USERAREA>
</PARTNER>
<CATITEM>
    <ITEM/>
    <SYNCIND/>
    <DESCRIPTN/>
    <USERAREA>
        <LISTPRICE/>
        <UOMCODE/>
        <NETPRICE/>
        <LINESTATUSCODE/>
        <ITEMCAT>
            <ITEMCATEGORY/>
        </ITEMCAT>
    <PBLINMSG>
        <TYPE/>
        <CODE/>
        <TEXT/>
    </PBLINMSG>
    <PBLINEDetail>
        <DETAILTYPE/>
        <BREAKTYPE/>
        <LISTPRICE/>
        <LISTNAME/>
        <MODLINENO/>
        <MODOPERAND/>
        <MODAPPMETHOD/>
        <ADJAMT/>
        <CNETPRICE/>
        <PBLINEATTR>
            <CTXT>
            <ATTR>
                <ATTRVALFROM>
                <ATTRVALTO>
                <ATTRTYPE>
            </PBLINEATTR>
        <PBLINEBRK>
            <CTXT/>
            <ATTR/>
                <ATTRVALFROM/>
                <ATTRVALTO/>
                <ATTRTYPE/>
                <OPERAND/>
            </PBLINEBRK>
        </PBLINEDetail>
    </USERAREA>

```

```
</CATITEM>
  </CATALOG>
</GET_CATALOG>
</DATAAREA>
</SYNC_CATALOG_003>
```

Trading Partner Setup

The Sync Catalog transaction uses Oracle XML Gateway to send the outgoing messages. To use this transaction, you must define a trading partner and enable the SYNC_CATALOG message for this partner in the Oracle XML Gateway Define Trading Partner Setup window. The relevant message for this transaction has transaction type QP and transaction subtype CATSO. For more information, see the *Oracle XML Gateway User's Guide*.

Validation of Sync Catalog Transaction

Standard Validation

No validation is required because there are no input parameters to SYNC_CATALOG. Oracle Advanced Pricing generates the SYNC_CATALOG message and delivers the message via the Oracle XML Gateway.

Error Handling

Although no validation is required, you can view any unexpected error messages using the Oracle XML Gateway Transaction Monitor and/or Oracle Workflow Status Monitor. For more information, see the available documentation for these modules.

Examples of Sync Catalog Transaction

The following examples show a full price book and a delta price book.

Full Price Book

The following example describes a generated full price book with the following characteristics:

- The price book is named "2006 - AS54999," effective 10/04/2006.
- The price book contains only item "AS54999."
- The price book is based on price list "Computer Desktops and Accessories."
- The item "AS54999" has a list price of \$100 and a net price of \$75.
- The pricing details include:
 - a price list line from "Computer Desktops and Accessories" with list price \$100.
 - a modifier from "F06-1" yielding a \$10 discount.
 - a modifier from "F06-1" yielding a 15% discount.

```

<SYNC_CATALOG_003>
  <CNTROLAREA>
    <BSR>
      <VERB value="SYNC"/>
      <NOUN value="CATALOG"/>
      <REVISION value="003"/>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>QP</COMPONENT>
      <TASK>CATSO</TASK>
      <REFERENCEID/>
      <CONFIRMATION>0</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION" type="T" index="1">
      <YEAR>2006</YEAR>
      <MONTH>02</MONTH>
      <DAY>24</DAY>
      <HOUR>10</HOUR>
      <MINUTE>04</MINUTE>
      <SECOND>15</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SYNC_CATALOG>
      <CATALOG>
        <DATETIME qualifier="EFFECTIVE" type="T" index="1">
          <YEAR>2006</YEAR>
          <MONTH>10</MONTH>
          <DAY>04</DAY>
          <HOUR>15</HOUR>
          <MINUTE>30</MINUTE>
          <SECOND>00</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+0000</TIMEZONE>
        </DATETIME>
        <CATALGNAME>2006 - AS54999</CATALGNAME>
        <SYNCIND/>
        <CATALOGRV/>
        <CLSSSCHMID/>
        <DESCRIPTN/>
        <USERAREA>
          <PBTYPECODE>F</PBTYPECODE>
          <CURRENCYCODE>USD</CURRENCYCODE>
          <ITEMQUANTITY>5</ITEMQUANTITY>
          <LIMITPRODUCTSBYCODE>ITEM</LIMITPRODUCTSBYCODE>
          <ITEM>AS54999</ITEM>
          <ITEMCATEGORY/>
          <PRICELIST>Computer Desktops and Accessories</PRICELIST>
          <PRICEBASEDONCODE>PRICE_LIST</PRICEBASEDONCODE>
          <AGREEMENT/>
          <BSA/>
          <PBATTR>
            <CTXT>Order</CTXT>
            <ATTR>Order Category</ATTR>
            <ATTRVALUE>ORDER</ATTRVALUE>
          </PBATTR>
        </USERAREA>
      </CATALOG>
    </SYNC_CATALOG>
  </DATAAREA>
</SYNC_CATALOG_003>

```



```

<ATTRTYPE>QUALIFIER</ATTRTYPE>
  </PBATTR>
  <PBATTR>
    <CTXT>Terms</CTXT>
    <ATTR>Payment Terms</ATTR>
    <ATTRVALUE>2 Net</ATTRVALUE>
    <ATTRTYPE>QUALIFIER</ATTRTYPE>
  </PBATTR>
</USERAREA>
<PARTNER>
  <NAME index="1">Business World</NAME>
  <PARTNRID>4429</PARTNRID>
  <PARTNRTYPE/>
  <PARTNRIDX/>
  <USERAREA>
    <CUSTACCT/>
  </USERAREA>
</PARTNER>
<CATITEM>
  <ITEM>2155</ITEM>
  <SYNCIND/>
  <DESCRIPTN>Sentinel Standard Desktop-Rugged</DESCRIPTN>
  <USERAREA>
    <LISTPRICE>100</LISTPRICE>
    <UOMCODE>Ea</UOMCODE>
    <NETPRICE>75</NETPRICE>
    <LINESTATUSCODE/>
    <ITEMCAT>
      <ITEMCATEGORY>MISC.MISC.</ITEMCATEGORY>
    </ITEMCAT>
    <ITEMCAT>
      <ITEMCATEGORY>COMPUTER.DESKTOP</ITEMCATEGORY>
    </ITEMCAT>
    <ITEMCAT>
      <ITEMCATEGORY>Computer/Hardware/Desktop</ITEMCATEGORY>
    </ITEMCAT>
    <PBLINEDetail>
      <DETAILTYPE>Price List Line</DETAILTYPE>
      <BREAKTYPE/>
      <LISTPRICE>100</LISTPRICE>
      <LISTNAME>Computer Desktops and Accessories</LISTNAME>
      <MODLINENO/>
      <MODOPERAND/>
      <MODAPPMETHOD>UNIT_PRICE</MODAPPMETHOD>
      <ADJAMT/>
      <CNETPRICE>100</CNETPRICE>
    </PBLINEDetail>
    <PBLINEDetail>
      <DETAILTYPE>Discount</DETAILTYPE>
      <BREAKTYPE/>
      <LISTPRICE/>
      <LISTNAME>F06-1</LISTNAME>
      <MODLINENO>609637</MODLINENO>
      <MODOPERAND>10</MODOPERAND>
      <MODAPPMETHOD>AMT</MODAPPMETHOD>
      <ADJAMT>-10</ADJAMT>
      <CNETPRICE>90</CNETPRICE>
    </PBLINEDetail>
    <PBLINEDetail>
      <DETAILTYPE>Discount</DETAILTYPE>
      <BREAKTYPE/>

```

```

<LISTPRICE/>
    <LISTNAME>F06-1</LISTNAME>
    <MODLINENO>607635</MODLINENO>
    <MODOPERAND>15</MODOPERAND>
    <MODAPPMETHOD>%</MODAPPMETHOD>
    <ADJAMT>-15</ADJAMT>
    <CNETPRICE>75</CNETPRICE>
  </PBLINEDetail>
</USERAREA>
</CATITEM>
</CATALOG>
</SYNC_CATALOG>
</DATAAREA>
</SYNC_CATALOG_003>

```

Delta Price Book

The following example shows a generated delta price book with the following characteristics:

- The delta price book is named *2006 - AS54999* effective 01/04/2007.
- The delta price book contains only one item AS54999 whose price has changed when compared with the full price book.
- The item AS54999 has a list price of \$100 and a net price of \$65.
- The pricing details include:
 - a price list line from Computer Desktops and Accessories with list price \$100
 - a modifier from *F06-1* yielding a \$10 discount
 - a modifier from *F06-1* yielding a 15% discount
 - a modifier from *W06-1* yielding a 10% discount

```

<SYNC_CATALOG_003>
  <CNTROLAREA>
    <BSR>
      <VERB value="SYNC"/>
      <NOUN value="CATALOG"/>
      <REVISION value="003"/>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>QP</COMPONENT>
      <TASK>CATSO</TASK>
      <REFERENCEID/>
      <CONFIRMATION>0</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION" type="T" index="1">
      <YEAR>2006</YEAR>
      <MONTH>02</MONTH>
      <DAY>24</DAY>
      <HOUR>15</HOUR>
      <MINUTE>50</MINUTE>
      <SECOND>23</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SYNC_CATALOG>
      <CATALOG>
        <DATETIME qualifier="EFFECTIVE" type="T" index="1">
          <YEAR>2007</YEAR>
          <MONTH>01</MONTH>
          <DAY>04</DAY>
          <HOUR>15</HOUR>
          <MINUTE>30</MINUTE>
          <SECOND>00</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+0000</TIMEZONE>
        </DATETIME>
        <CATALGNAME>2006 - AS54999</CATALGNAME>
        <SYNCIND/>
        <CATALOGRV/>
        <CLSSSCHMID/>
        <DESCRIPTN/>
        <USERAREA>
          <PBTYPECODE>D</PBTYPECODE>
          <CURRENCYCODE>USD</CURRENCYCODE>
          <ITEMQUANTITY>5</ITEMQUANTITY>
          <LIMITPRODUCTSBYCODE>ITEM</LIMITPRODUCTSBYCODE>
          <ITEM>AS54999</ITEM>
          <ITEMCATEGORY/>
          <PRICELIST>Computer Desktops and Accessories</PRICELIST>
          <PRICEBASEDONCODE>PRICE_LIST</PRICEBASEDONCODE>
          <AGREEMENT/>
          <BSA/>
          <PBATTR>
            <CTXT>Order</CTXT>
            <ATTR>Order Category</ATTR>
            <ATTRVALUE>ORDER</ATTRVALUE>
          </PBATTR>
        </USERAREA>
      </CATALOG>
    </SYNC_CATALOG>
  </DATAAREA>
</SYNC_CATALOG_003>

```

```

<ATTRTYPE>QUALIFIER</ATTRTYPE>
  </PBATTR>
  <PBATTR>
    <CTXT>Terms</CTXT>
    <ATTR>Payment Terms</ATTR>
    <ATTRVALUE>2 Net</ATTRVALUE>
    <ATTRTYPE>QUALIFIER</ATTRTYPE>
  </PBATTR>
</USERAREA>
<PARTNER>
  <NAME index="1">Business World</NAME>
  <PARTNRID>4429</PARTNRID>
  <PARTNRTYPE/>
  <PARTNRIDX/>
  <USERAREA>
    <CUSTACCT/>
  </USERAREA>
</PARTNER>
<CATITEM>
  <ITEM>2155</ITEM>
  <SYNCIND/>
<DESCRIPTN>Sentinel Standard Desktop
- Rugged</DESCRIPTN>
<USERAREA>
  <LISTPRICE>100</LISTPRICE>
  <UOMCODE>Ea</UOMCODE>
  <NETPRICE>65</NETPRICE>
  <LINESTATUSCODE/>
  <ITEMCAT>
    <ITEMCATEGORY>MISC.MISC.</ITEMCATEGORY>
  </ITEMCAT>
  <ITEMCAT>
    <ITEMCATEGORY>COMPUTER.DESKTOP</ITEMCATEGORY>
  </ITEMCAT>
  <ITEMCAT>
    <ITEMCATEGORY>Computer/Hardware/
    Desktop</ITEMCATEGORY>
  </ITEMCAT>
  <PBLINEDetail>
    <DETAILTYPE>Price List Line</DETAILTYPE>
    <BREAKTYPE/>
    <LISTPRICE>100</LISTPRICE>
    <LISTNAME>Computer Desktops and
    Accessories </LISTNAME>
    <MODLINENO/>
    <MODOPERAND/>
    <MODAPPMETHOD>UNIT_PRICE</MODAPPMETHOD>
    <ADJAMT/>
    <CNETPRICE>100</CNETPRICE>
  </PBLINEDetail>
  <PBLINEDetail>
    <DETAILTYPE>Discount</DETAILTYPE>
    <BREAKTYPE/>
    <LISTPRICE/>
    <LISTNAME>F06-1</LISTNAME>
    <MODLINENO>609637</MODLINENO>
    <MODOPERAND>10</MODOPERAND>
    <MODAPPMETHOD>AMT</MODAPPMETHOD>
    <ADJAMT>-10</ADJAMT>
    <CNETPRICE>90</CNETPRICE>
  </PBLINEDetail>

```

```

<PBLINEDetail>
    <DetailType>Discount</DetailType>
    <BreakType/>
    <ListPrice/>
    <ListName>W06-1</ListName>
    <ModLineNo>610643</ModLineNo>
    <ModOperand>10</ModOperand>
    <ModAppMethod>%</ModAppMethod>
    <AdjAmt>-10</AdjAmt>
    <NetPrice>80</NetPrice>
</PBLINEDetail>
<PBLINEDetail>
    <DetailType>Discount</DetailType>
    <BreakType/>
    <ListPrice/>
    <ListName>F06-1</ListName>
    <ModLineNo>607635</ModLineNo>
    <ModOperand>15</ModOperand>
    <ModAppMethod>%</ModAppMethod>
    <AdjAmt>-15</AdjAmt>
    <NetPrice>65</NetPrice>
</PBLINEDetail>
</UserArea>
</CatItem>
</Catalog>
</Sync_Catalog>
</DataArea>
</Sync_Catalog_003>

```

Oracle Order Management XML Transactions: Implementation Considerations

This chapter covers the following topics:

- Process_PO
- Acknowledge_PO
- Show_SalesOrder
- Change_SalesOrder
- Change_PO
- Cancel_PO
- Open Interface Tracking
- Integration Event – oracle.apps.ont.oi.xml_int.status

Process_PO

Setup: Customer Setup

Using the Oracle Order Management Super User responsibility, Customer Standard window (Customers-> Standard), query a customer for whom o create XML orders. On one of the customer's addresses, create a primary **Sold To** site usage and specify an EDI Location (for example Q7Q-1A1). Save your work. When provided on the inbound XML document, this information will be used to denote a particular customer site and also to determine the destination of the acknowledgment.

Customer Addresses - Computer Service and Rentals, 1006

Country: **United States** Site Number: **1034**

Address: **301 Summit Hill Drive**

Alternate Name: City: **Chattanooga** State: **TN**

Postal Code: **37401** Province: County: **Hamilton**

EDI Location: **Q7Q-1A1** ☐ Identifying Address ☒ Active

Addressee:

Business Purposes | Characteristics | Communication | Contacts | Contacts : Roles | Bank Accounts

Usage	Location	Bill To Location	Primary	Active
Bill To	Chattanooga (OPS)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ship To	Chattanooga (OPS)	Chattanooga (OPS)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sold To			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>

New Open

The following table lists the setup steps for Process_PO:

Process_PO Setup Steps

Step	Description
1	Define Transactions
2	Define Trading Partner/Hub
3	Define Code Conversion
4	Set Up the Oracle Workflow Business Event System

Define Transactions

Use Oracle XML Gateway to define a cross reference between the Oracle transaction name and the external transaction name. The external transaction name will be based on what is meaningful per the XML standard used by the recipient. The external transaction name will appear on the message envelope to support message transport.

Define Transactions

Party Type: **Customer**

Transaction Type: **ONT**

Transaction Subtype: **POI**

Transaction Description: **Purchase Order Inbound**

— External Transactions —

Standard Code	Direction	External Transaction Type	External Transaction Subtype	Queue
OAG	IN	PO	PROCESS	APPS.ECX_IN_OAG_Q

Define Trading Partner/Hub

e-Business can be conducted directly with a business partner commonly known as a trading partner or by means of a hub such as Oracle Exchange where many buyers and sellers converge to conduct electronic commerce.

With Oracle XML Gateway, you can define the hub or the actual business partner as a trading partner. If you define the hub as the trading partner, you can identify all the buyers and sellers who are conducting business on the hub as trading partners to the hub.

Included in the trading partner/hub definition are the following:

- Trading partner/hub name
- Message enabled
- Message confirmation requested
- Message map to use for message creation or consumption
- E-mail address of trading partner contact to notify for data errors
- Trading partner specific code conversion values
- Transport protocol - SMTP, HTTP, and HTTPS with credential and username/password as necessary

The Trading Partner Setup window is used to:

- Enable messages for the trading partner by identifying the internal and external transaction type and transaction subtype codes, and the XML standard associated with the message.
- Access the Trading Partner Code Conversion window.
- Select a message map for the trading partner.
- Identify the communications protocol and address for a message. Optionally, the user can be selected from a hub.

To set up a trading partner:

1. Navigate to the Define Trading Partner Setup form from the XML Gateway Responsibility by selecting Setup > Define Trading Partners.

Transaction Type	Transaction SubType	Standard Code	External Transaction Type	External Transaction SubType	Direction Map	Connection/Hub	Protocol Type
ECX	CBODO	OAG	BOD	CONFIRM	OUT	ECX_CBODO	SMTP
ONT	POI	OAG	PO	PROCESS	IN	ONT_3A4R_O	
ONT	CPO	OAG	PO	CANCEL	IN	ONT_3A9R_O	
ONT	POA	OAG	PO	ACKNOWLED	OUT	ONT_3A4A_O	SMTP
ONT	SSO	OAG	SALESORDE	SHOW	OUT	ONT_3A6_OA	SMTP

2. Enter Customer in the Trading Partner Type field.
3. Select values in the Trading Partner and the Trading Partner Site fields.
4. Select ONT in the Transaction Type field.
5. Select ONT_3A4R_OAG72_IN in the Map field.
6. Save your work.

Define Code Conversion

With Oracle XML Gateway, you can identify the cross reference of an Oracle code to something that is meaningful to the recipient or vice versa. Common examples of Oracle e-Business Suite codes requiring code conversion are units of measure and currency code.

Code conversion values can be defined to be applied universally for all trading partners and all messages. Additionally, code conversion values can be defined for a specific XML standard or specific to a trading partner.

The Oracle XML Gateway code conversion function provides a method to cross-reference the codes defined in Oracle Applications to codes used by trading partners, the XML standard, or other standard codes in the transactions.

The Code Conversion Standard code determines to which transaction standard these values are applied. A standard code of UNIVERSAL applies these values across all standards. In this case use the Code Conversion window from the Navigator Setup menu.

The image shows a window titled "Standard Code Conversion". It contains two main sections:

Category Code

Category Code	Description
UNIT_PRICE_BASIS	Basis of Unit Price Code
UOM	
UPC	Uniform Product Code (UPC) Code

Category Values

Standard Code	Oracle Value	Description	From Trading Partner Value	To Trading Partner Value	Standard	Data Seeded
OAG	EACH		ea	ea	<input checked="" type="checkbox"/>	<input type="checkbox"/>
OAG	GAL		gallon	gallon	<input checked="" type="checkbox"/>	<input type="checkbox"/>
OAG	PIECE		pc	pc	<input checked="" type="checkbox"/>	<input type="checkbox"/>
RosettaNet	EACH		EA	EA	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
UNIVERSAL	EACH		EA	EA	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UNIVERSAL	PIECE		PC	PC	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UNIVERSAL	SLAB		SL	SL	<input checked="" type="checkbox"/>	<input type="checkbox"/>
					<input type="checkbox"/>	<input type="checkbox"/>

Buttons: Revert All, Revert

Define the Subscribing System for the Event Subscription for the Event 'oracle.apps.ont.oi.po_inbound.create'

Subscriber: System - This will usually be the local system, the system where the subscription code is to run.

Note: This data will be available as the initial setup, and comes as an XML file and will be loaded to the customer system. The user of the

system can change it to work for their system settings.

Set Up the Oracle Workflow Business Event System

Oracle XML Gateway leverages the Oracle Workflow Business Event System to publish and subscribe to application business events of interest to automatically trigger message creation or consumption.

Seeded business events and event subscriptions to send outbound or consume inbound messages are delivered for all Oracle pre-built messages.

Note: The seeded event subscriptions can be configured during implementation to perform activities to address specific business requirements.

The XML Gateway Execution Engine interfaces with Oracle Workflow to actively notify the XML Gateway system administrator regarding system or process errors, or the trading partner contact for data errors.

The XML Gateway system administrator has the option to "retry" failed outbound processes, or "reprocess" failed inbound processes.

The following function is internal to the XML Gateway Execution Engine:

Validate Trading Partner/Hub

Verify that the trading partner is defined and the required documents are enabled.

Message Set Up

To implement a message with a trading partner, use XML Gateway message set up to define the trading partner or hub, code conversion values, and internal to external transaction name cross references. In addition, you can identify the XML Gateway system administrator to notify for system or process errors.

Installation

This functionality is available as part of Oracle Order Management.

Loading of Message Map and DTD

Oracle message maps are delivered and installed as a part of Oracle Order Management file structure and schema. They are automatically loaded into the XML Gateway repository using the LoadMap program. The reports are available to verify the success of the loading process.

Enabling and Consuming the Business Event

The necessary business events and subscriptions are also delivered and installed as part

of the schema, they are loaded by the driver.

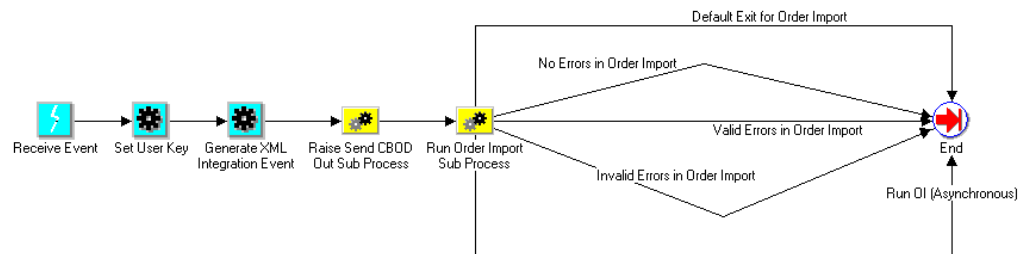
Integrations

- Oracle XML Gateway
- Oracle Workflow
- Oracle Advanced Queuing

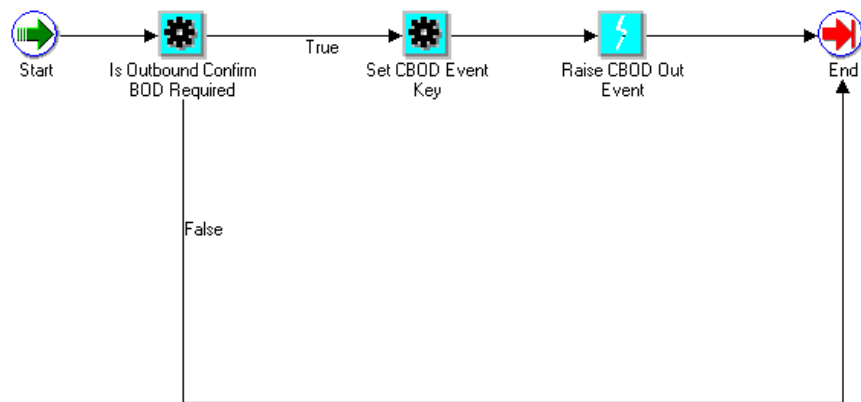
Message Map: Seeded Workflow for Inbound XML Messages

A pre-defined Workflow event is seeded to run Order Import immediately if the Profile Option OM: Run Order Import for XML is set to 'Synchronous.' This Workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the Workflow Process it will run Order Import for the XML message, that is currently being consumed. The following Workflow is designed using Workflow Builder 2.6.

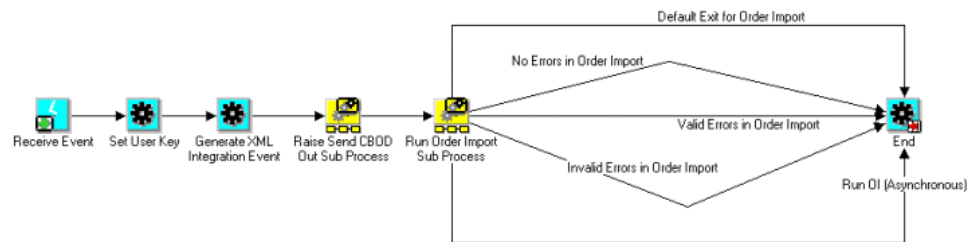
Order Import Flow - Generic



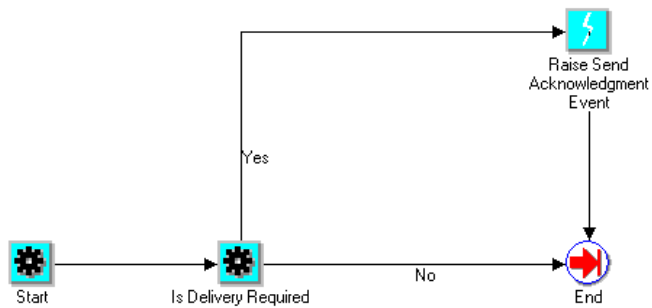
Raise Send CBOD Out Sub Process



Run Order Import Sub Process



Raise Send Acknowledgement Sub Process



Sample Business Flow for 3A4 Process PO/3A4 Acknowledge PO

Sample Business Flow for 3A4 Process PO/3A4 Acknowledge PO

Step	Buyer System E.G. Oracle Purchasing	Oracle Order Management
0		<p>Figure, "Trading Partner Setup completed for Oracle Order Management via XML Gateway forms"</p> <p>Figure, "Profile Option 'OM: Run Order Import for XML' set to 'SYNCHRONOUS' at site-level"</p>
1	Figure, "Buyer approves a Purchase Order"	
2		<p>Figure, "Order Management receives the Purchase Order electronically as a Process PO XML document"</p> <p>Figure, "Open Interface Tracking Details Main Tab"</p> <p>Figure, "Results Summary Inbound Message Details"</p>

Step	Buyer System E.G. Oracle Purchasing	Oracle Order Management
3		Figure, "Seeded inbound workflow is automatically triggered and runs Order Import"
		Figure, "Corresponding XML Sales Order is created Header"
		Figure, "Corresponding XML Sales Order is created Lines"
4		Figure, "Seeded outbound workflow is automatically triggered. This flow generates and sends Acknowledge PO to the buyer"
		Figure, "Status Diagram"
		Figure, "Transaction Monitor"
		Figure, "Results Summary"
		Figure, "Results Summary Outbound Message Details"

Trading Partner setup completed for Oracle Order Management using XML Gateway forms

Trading Partner Setup

Trading Partner Type: **Customer**

Trading Partner Name: **XMLCUST02**

Trading Partner Site: **Cyber XMLCUST02A Chattanooga TN 37401**

Company Admin Email: **rparamat@oracle.com**

Code Conversion

Trading Partner Details

Transaction Type	Transaction SubType	Standard Code	External Transaction Type	External Transaction SubType	Direction Map	Connection/Hub	Protocol Type
ECX	CBODO	OAG	BOD	CONFIRM	OUT	ECX_CBODO	SMTP
ONT	POA	OAG	PO	ACKNOWLED	OUT	ONT_3A4A_O	SMTP
ONT	CPO	OAG	PO	CANCEL	IN	ONT_3A9R_O	
ONT	CHO	OAG	PO	CHANGE	IN	ONT_3A8R_O	
ONT	POI	OAG	PO	PROCESS	IN	ONT_3A4R_O	
ONT	CSO	OAG	SALESORDE	CHANGE	OUT	ONT_3A7_OA	SMTP
ONT	SSO	OAG	SALESORDE	SHOW	OUT	ONT_3A6_OA	SMTP

Profile Option 'OM: Run Order Import for XML' set to 'SYNCHRONOUS' at site-level

System Profile Values

Profile	Site	Application	Responsibility	User
OM: Run Order Import for XML	SYNCHRONOUS			

Buyer approves a Purchase Order

Purchase Orders (Vision Operations) - [New]

PO, Rev	12122	0	Type	Standard Purchase	Created	18-JUN-2004 18:19
Supplier	ABC Order Manag		Site	SUPPLIER SITE	Contact	
Ship-To	V1- New York City		Bill-To	V1- New York City	Currency	USD
Buyer	Stock, Ms. Pat		Status	Incomplete	Total	27,696.25
Description						
P-Card						Transaction Code

[Lines](#)
[Price Reference](#)
[Reference Documents](#)
[More](#)
[Agreement](#)
[Temporary Labor](#)

Num	Rev	Job	Category	Description	UOM	Quantity
1			PRODUCTN.FIN	Dimension 4550	Each	10
2			PRODUCTN.FIN	Dimension 4550	Each	15


Item **AS54888** **Dimension 4550**

[Catalog...](#)
[Currency...](#)
[Terms](#)
[Shipments](#)
[Approve...](#)

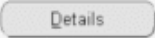
Order Management receives the Purchase Order electronically as a Process PO XML document

Open Interface Tracking

Main Others



Order Source	Order Source Reference	Customer	Customer Number
XML	PM.XPO-0617A	XMLCUST02	3908



Open Interface Tracking Details Main Tab

Open Interface Tracking Details

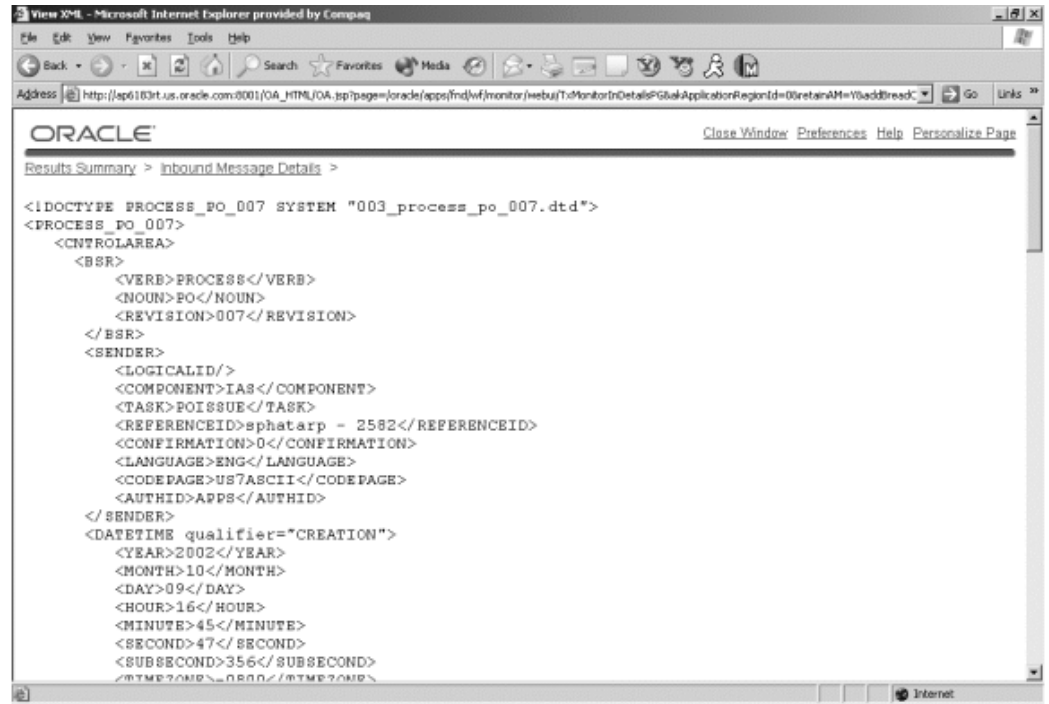
Customer Number: 3908 Order Source: XML
Customer: XMLCUST02 Message Count: 6
Order Source Reference: PM-XP0-0617A

Main Others

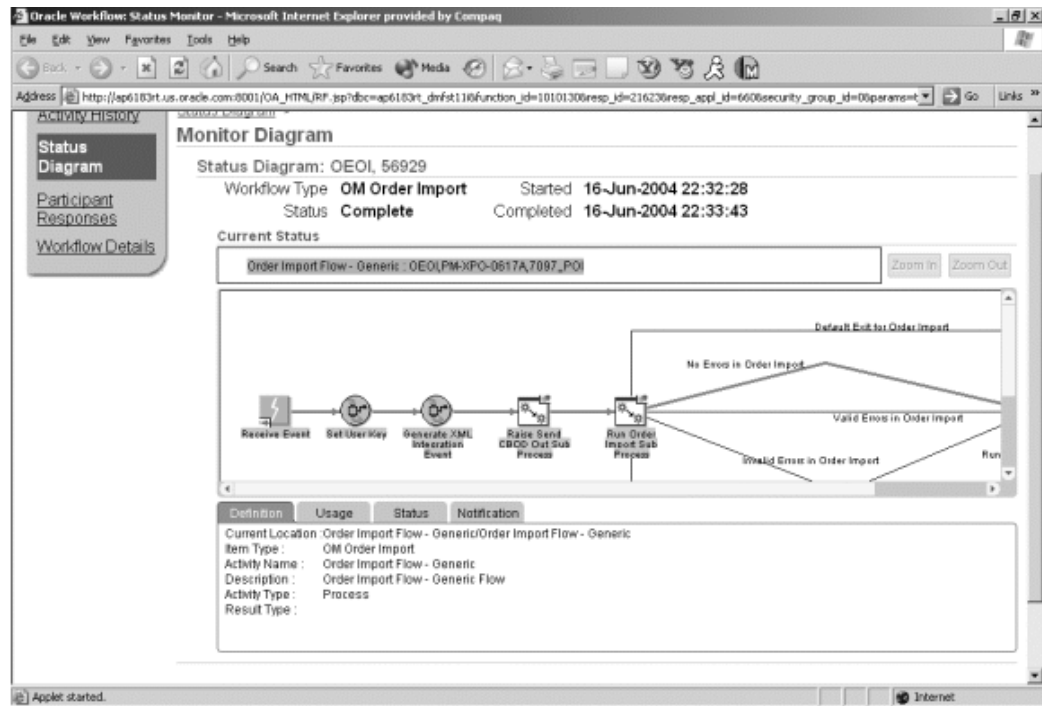
Transaction Type	Status	Message Text
Process PO	SUCCESS	Process PO - Order Import successful

WF Monitor XML Monitor Corrections

Results Summary Inbound Message Details



Seeded inbound workflow is automatically triggered and runs Order Import



Corresponding XML Sales Order is created Header

Open Interface Tracking Details

Customer Number: 3908
Customer: XMLCUST02
Order Source Reference: PM-XP0-0617A

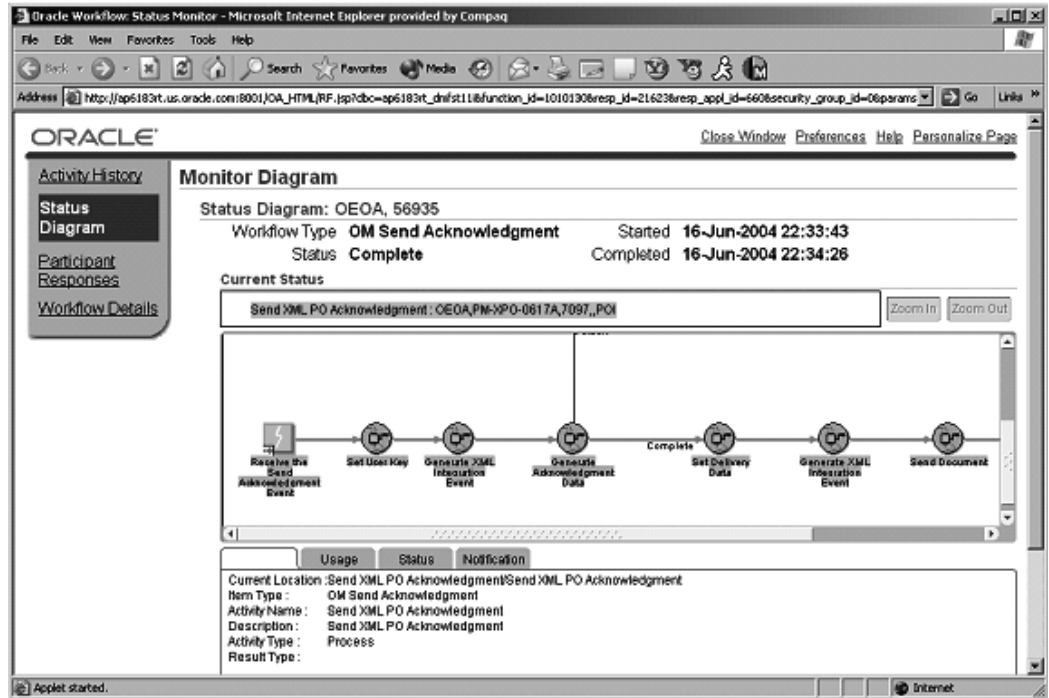
Order Source: XML
Message Count: 6

Main Others

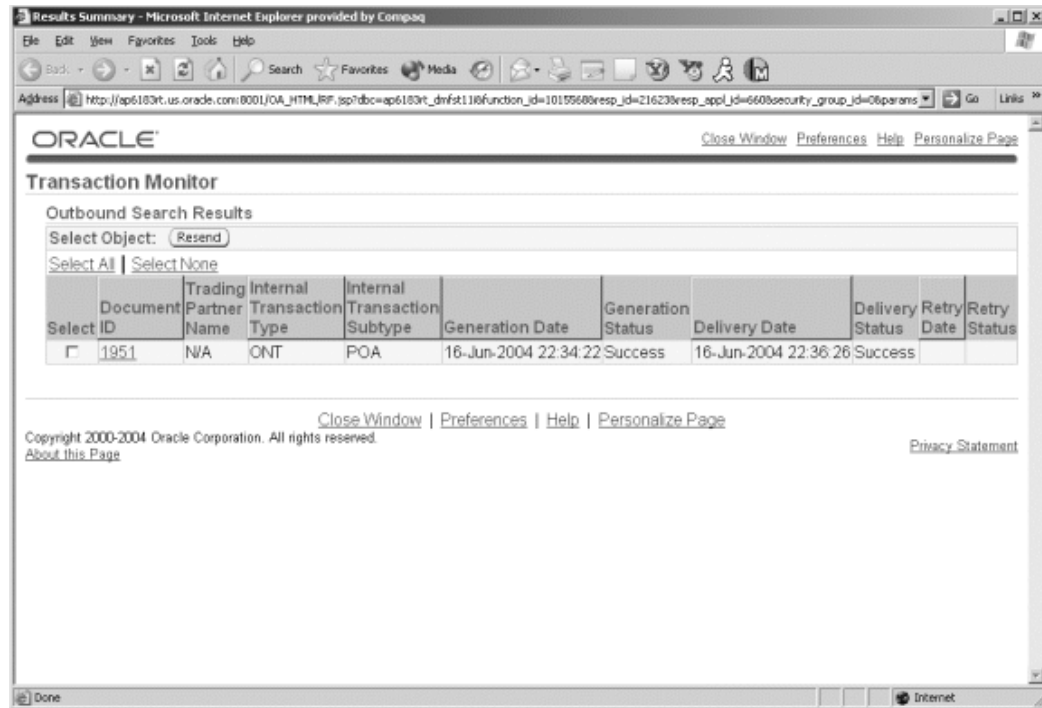
Transaction Type	Status	Message Text
Process PO	SUCCESS	Process PO - Order Import successful
Acknowledge PO	SUCCESS	Acknowledge PO message has been sent.

WF Monitor XML Monitor Corrections

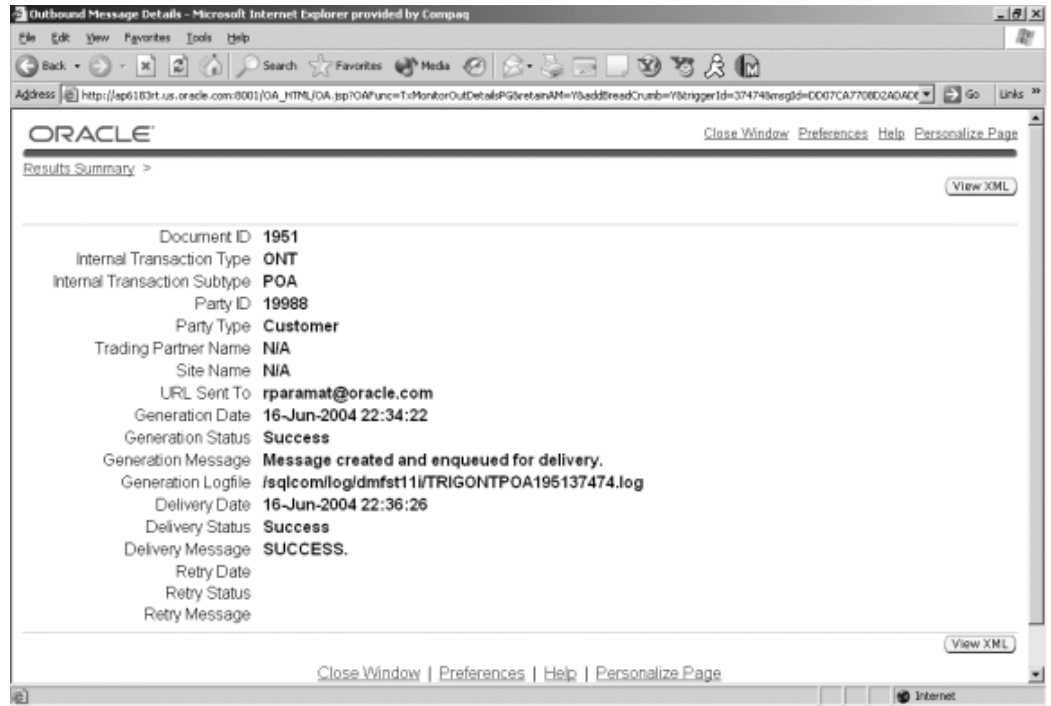
Status Diagram



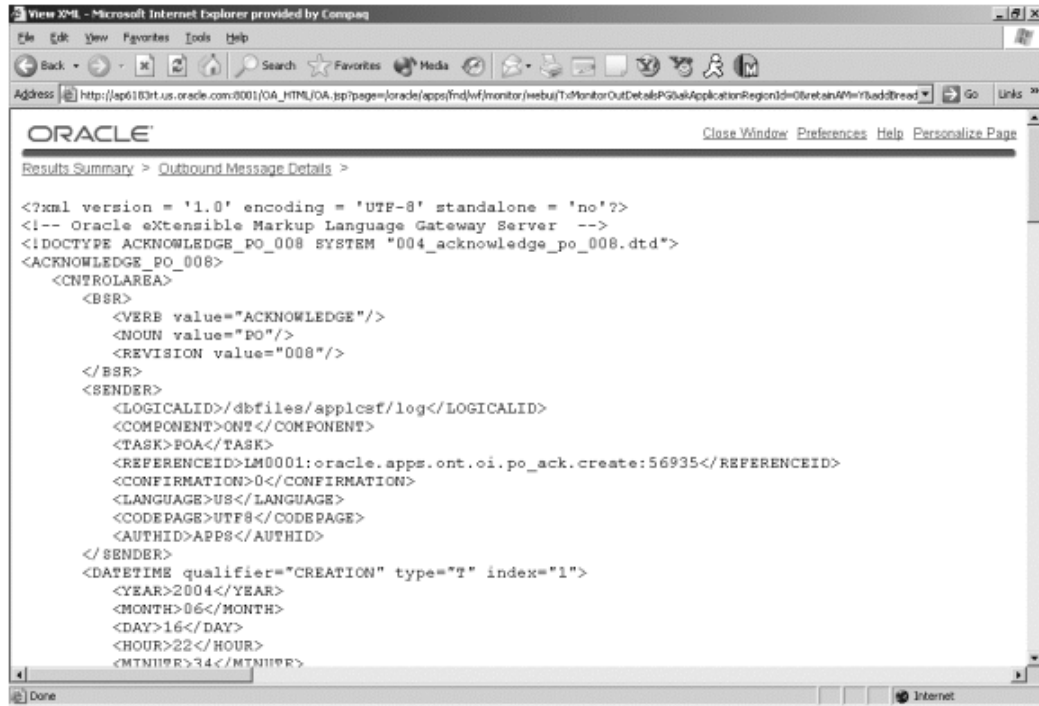
Transaction Monitor



Results Summary



Results Summary Outbound Message Details



Acknowledge_PO

Message Map

Loading of Message Map and DTD

Oracle message maps are delivered and installed as a part of the Oracle Order Management file structure and schema. They are automatically loaded into the XML Gateway repository using the LoadMap program. The reports are available to verify the success of the loading process.

Enabling and Consuming the Business Event

Create the Business Event and a Subscription for the event in the Application using the Workflow Administrator Responsibility. This event will be raised by the Post Process Action once the XML Message is consumed by Oracle XML Gateway and the data is loaded in the Open Interface Tables.

Setup: Customer Setup

Using the Oracle Order Management Super User responsibility, Customer Standard window (Customers-> Standard), query a customer to create XML orders for. On one of the customer's addresses, create a primary **Sold To** site usage and specify an EDI

Location (e.g. Q7Q-1A1). Save your work. When provided on the inbound XML document, this information will be used to denote a particular customer site and also to determine the destination of the acknowledgment.

Usage	Location	Bill To Location	Primary	Active
Bill To	Chattanooga (OPS)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ship To	Chattanooga (OPS)	Chattanooga (OPS)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sold To	Chattanooga (OPS)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Before performing these Order Management Suite XML setup steps, check the documentation for the following areas and verify that you have performed their set up:

- The Oracle Applications with which this transaction deals
- Oracle XML Gateway
- Oracle Workflow

Setup Steps

Step	Description	New Transaction Development	Transaction Implementation
1	Define Transactions	Required	Optional
2	Define Trading Partner and Hub	NA	Required

Step	Description	New Transaction Development	Transaction Implementation
3	Define Code Conversion	Required	Required
4	Setup Oracle Workflow Business Event System	NA	As Needed

Define Transactions

Use Oracle XML Gateway to define a cross reference between the Oracle transaction name and the external transaction name. The external transaction name will be based on what is meaningful per the XML standard used by the recipient. The external transaction name will appear on the message envelope to support message transport.

Define Trading Partner and Hub

e-Business may be conducted directly with a business partner commonly known as a trading partner or via a hub such as Oracle Exchange where many buyers and sellers converge to conduct electronic commerce.

With Oracle XML Gateway, you can define the hub or the actual business partner as a trading partner. If you define the hub as the trading partner, you can identify all the buyers and sellers who are conducting business on the hub as trading partners to the hub.

Included in the trading partner/hub definition are the following:

- Trading Partner/Hub name
- Message enabled
- Message confirmation requested
- Message map to use for message creation or consumption
- E-mail address of trading partner contact to notify for data errors
- Trading Partner specific code conversion values
- Transport protocol - SMTP, HTTP, or HTTPS with credential and username and password as necessary

Define Code Conversion

With Oracle XML Gateway, you may identify the cross reference of an Oracle code to something that is meaningful to the recipient or vice versa. Common examples of Oracle e-Business Suite codes requiring code conversion are units of measure and currency code.

Code conversion values may be defined to be applied universally for all trading partners and all messages. Additionally, code conversion values may be defined for a specific XML standard or specific to a trading partner.

Setup Oracle Workflow Business Event System

Oracle XML Gateway leverages the Oracle Workflow Business Event System to publish and subscribe to application business events of interest to automatically trigger message creation or consumption.

The following seeded event subscription, will listen to the event `oracle.apps.ont.oi.po_acknowledge.create` and will send the acknowledgment.

The XML Gateway Execution Engine interfaces with Oracle Workflow to actively notify the XML Gateway system administrator regarding system or process errors, or the trading partner contact for data errors.

The XML Gateway system administrator has the option to retry failed outbound processes, or reprocess failed inbound processes.

The raising of this event can be done, as part of the new workflow added for the Order Import Inbound flow. After Order Import (Run Order Import), finishes with the status complete we will Raise event, that will be received by above workflow for creating the acknowledgment.

The above workflow will be provided as standard flow (where the acknowledgment is sent when the synchronous flow of order import completes), the user can move the event and put it anywhere in their existing standard flow.

Message Set Up

To implement a message with a trading partner, use XML Gateway message set up to define the trading partner or hub, code conversion values, and internal to external transaction name cross references. In addition, you may identify the XML Gateway system administrator to notify for system or process errors.

The XML Gateway Execution Engine interfaces with Oracle Workflow to actively notify the XML Gateway system administrator regarding system or process errors, or the trading partner contact for data errors.

The XML Gateway system administrator has the option to "retry" failed outbound processes, or "reprocess" failed inbound processes.

The raising of this event can done, as part of the new workflow added for the Order Import Inbound flow. After Order Import (Run Order Import), finish with status

complete we will Raise event, which will be received by above workflow for creating the acknowledgment.

Oracle XML Gateway Details

This table lists the Oracle XML Gateway Details.

Oracle XML Gateway Details

Detail	Value
Message Map Name:	ONT_3A4A_OAG72_OUT_PO
Direction:	Outbound
(Internal) Transaction Type:	ONT
(Internal) Transaction Subtype:	POA
External Transaction Type:	PO
External Transaction Subtype:	ACKNOWLEDGE
DTD Directory:	xml/oag72
Map Directory:	patch/115/xml/US
Message Maps XGM File Name:	ONT_3A4A_OAG72_OUT_PO.xgm
Standard:	OAG
Release:	7.2
Format:	DTD
DTD Name:	004_acknowledge_po_008.dtd

USERAREAs

This table explains the data in the USERAREAs.

USERAREA Data

Within Data Type	Data Element	Content
POORDERLIN	USERITEMDESCRIPTN	User Item Description
POLINESCHD	ACKCODE	Acknowledgment Code
POLINESCHD	DATETIME	Scheduled Arrival Date
POLINESCHD	SALESORDID	Sales Order Number
POLINESCHD	SOLINENUM	Line Number

Seed Codes for the Code Conversion

Following elements need to be enabled for code conversions:

- Currency
- UOM
- POType
- AckCode
- Termid
- Line Status

Show_SalesOrder

Setup

Loading of Message Map and DTD

Oracle message maps are delivered and installed as a part of the Oracle Order Management file structure and schema. They are automatically loaded into the XML Gateway repository using the LoadMap program. The reports are available to verify the success of the loading process.

Enabling and Consuming the Business Event

Create the Business Event and a Subscription for the event in the application using the Workflow Administrator responsibility. This event will be raised by the either the WF process (WF Support), the Concurrent Program (Periodical Support), or through the Order Management internal processing (Attribute/Status Change Support).

Using the Oracle Order Management Super User responsibility, Customer Standard window (Customers-> Standard), query a customer to create XML orders for. On one of the customer's addresses, create a primary **Sold To** site usage and specify an EDI Location (e.g. Q7Q-1A1). Save your work. When provided on the inbound XML document, this information will be used to denote a particular customer site and also to determine the destination of the acknowledgment.

Usage	Location	Bill To Location	Primary	Active
Bill To	Chattanooga (OPS)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ship To	Chattanooga (OPS)	Chattanooga (OPS)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sold To			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>

Setup Steps

Step	Description
1	Define Transactions
2	Define Trading Partner/Hub
3	Define Code Conversion

Step	Description
4	Setup Oracle Workflow Business Event System
5	Setting Up Constraints

Define Transactions

Use Oracle XML Gateway to define a cross reference between the Oracle transaction name and the external transaction name. The external transaction name will be based on what is meaningful per the XML standard used by the recipient. The external transaction name will appear on the message envelope to support message transport.

Define Trading Partner/Hub

e-Business may be conducted directly with a business partner commonly known as a trading partner or via a hub such as Oracle Exchange where many buyers and sellers converge to conduct electronic commerce.

With Oracle XML Gateway, you can define the hub or the actual business partner as a trading partner. If you define the hub as the trading partner, you can identify all the buyers and sellers who are conducting business on the hub as trading partners to the hub.

Included in the trading partner/hub definition are the following:

- Trading Partner/Hub name
- Message enabled
- Message confirmation requested
- Message map to use for message creation or consumption
- E-mail address of trading partner contact to notify for data errors
- Trading Partner specific code conversion values
- Transport protocol - SMTP, HTTP, or HTTPS with credential and username/password as necessary
- Source Location Code

Define Code Conversion

With Oracle XML Gateway, you may relate the cross reference of an Oracle code to

something that is meaningful to the recipient or vice versa. Common examples of Oracle e-Business Suite codes requiring code conversion are units of measure and currency code.

Code conversion values may be defined to be applied universally for all trading partners and all messages. Additionally, code conversion values may be defined for a specific XML standard or specific to a trading partner.

Setup Oracle Workflow Business Event System

Oracle XML Gateway leverages the Oracle Workflow Business Event System to publish and subscribe to application business events of interest, to automatically trigger message creation or consumption.

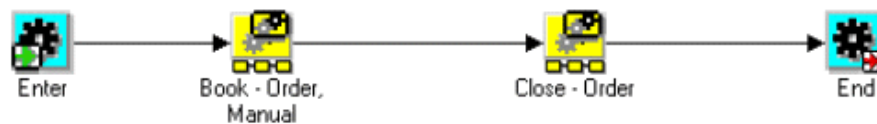
The following seeded event subscription will listen to the event `oracle.apps.ont.oi.showso.create` and will send the acknowledgment.

The XML Gateway Execution Engine interfaces with Oracle Workflow to actively notify the XML Gateway system administrator regarding system or process errors, or the trading partner contact for data errors.

The XML Gateway system administrator has the option to "retry" failed outbound processes, or "reprocess" failed inbound processes.

Workflow Support

The raising of this event can be done, as part of the any existing workflow available for Order Header (OEOH) or/and Order Line (OEOL) flow. The following example shows one of the positions where the raise event process can be inserted for sending the 3A6 in OEOH flow (after Booking), the process should be copied (or created same way) from the new workflow (OESO).



Note: This is currently only supported for the OEOH flow (Pack J onwards).

The Raise Show Sales Order Event Sub Process will check if the Sold To customer on the order is set up as a Trading Partner. If the customer is enabled for 3A6 Show SO, the sub-process will raise the event `oracle.apps.ont.oi.show_so.create` to generate and send the 3A6 Show SO XML document for XML orders.

In addition, the event `oracle.apps.ont.oi.xml_int.status` will be also be raised (for both XML and non-XML orders).

Workflow for The Raise Event

Raise Event Position	-
----------------------	---

Standard OM Workflow (OEOH/OEOL)

The workflow will continue to the point where the raise event is entered. When it hit raise event, the Show SO (3A6) with status will be send.



Validate Trading Partner/Hub

Verify that the trading partner is defined and required document are enabled.

Additional User Setup

ON Demand

ON Demand generation of the Show_SalesOrder, can be achieved by running the concurrent program on an as needed basis (this means run the concurrent program to send the Show_SalesOrder).

Attribute Change

No User setup is required for the Show_SalesOrder in event of following attributes change:

- Unit selling Price
 - Ordered Qty
 - Scheduled arrival date
 - Shipped Qty
 - Scheduled Ship Date

Any time any of these attributes change for an order (after the order has been booked) Order Management will generate the Show_Salesorder message.

Adding a new line to a booked order will also generate a Show_Salesorder message.
Splitting a line on a booked order will also generate a Show_Salesorder message.

Status Change

No User setup is required for the Show_SalesOrder in event of following status change:

- Booking the order
- Shipping
- Scheduling of lines

When an order or line status changes, the show SO is generated.

Partner Segment Information for Partner type 'Supplier' Not Mapped

The Partner segment Supplier will be derived from the setup in XML Gateway. Trading partner setup allows the setup of source location code, which is a source trading partner location code for the particular trading partner and particular transaction (in many cases this could be the location code of the supplying warehouse). This data will be picked up by Order Management to populate the Supplier location code on the outbound Show_SalesOrder message.

Make sure that while performing the trading partner setup the source location for the outbound transaction (for Order Management) is set up. This source location needs to be a valid EDI location code that will be validated by the XML Gateway and the Oracle Application Server 10g (in case of Rosettanet implementations).

See the User Procedures section for additional information about this setup.

User Procedures

Defining Trading Partners

The Trading Partner Setup window is used to:

- Enable messages for the trading partner by identifying the internal and external transaction type and transaction subtype codes, and the XML standard associated with the message.
- Access the Trading Partner Code Conversion form.
- Select a message map for the trading partner.
- Identify the communications protocol and address for a message. Optionally, the user can be selected from a hub.

To define a trading partner:

1. Navigate to the Define Trading Partner Setup window from the XML Gateway Responsibility. Setup > Define Trading Partners.

2. Enter the Trading Partner Type, Customer.
3. Select the Trading Partner and Trading Partner Site.
4. Select the transaction type as 'ONT' for this functionality.
5. Select 'ONT_3A6_OAG72_OUT_SSO' as the map.
6. Save your work.

Define Code Conversion Values

The Oracle XML Gateway code conversion function provides a method to cross-reference the codes defined in Oracle Applications to codes used by trading partners, the XML standard, or other standard codes in the transactions.

The Code Conversion Standard code determines which transaction Standard these values are applied to. A Standard Code of 'UNIVERSAL' applies these values across all Standards. In this case run the Code Conversion form from the Navigator Setup menu.

Define the Subscribing System for the event subscription for event 'oracle.apps.ont.oi.show_so.create'

Subscriber: System - This will usually be the local system, the system on which the subscription code is to run.

Define the Source Trading Partner Location Code

Setup the Source trading partner EDI Location Code. This EDI location Code will be used as the Suppliers EDI Location code which is sent as the PARTNRIDX for Partner 'Ship From.'

This is as part of the Trading Partner Setup window. For each trading partner that you have, specify the supplier location code to send on the Outbound XML transaction.

Setting Up Constraints

This details the setups required for constraint-based generation of an XML integration event and the Show SO XML document.

You can setup a user action called 'Raise Integration Event' in the Constraints window. This can be set for any attribute that is available and for any operation.

This action will raise the 'oracle.apps.ont.oi.xml_int.status' workflow business event. Interested parties can subscribe to the event in a workflow to perform any desired actions. Also, if the Customer on the order is defined as a trading partner in XML Gateway and is enabled to receive the Show SO transaction, a Show SO XML document will be generated and sent to that Customer.

You can also use the validation templates to perform any necessary validations before raising the event and sending the XML document. For example, if the event should be raised only for a specific trading partner then a validation template can be utilized to check for the customer corresponding to that trading partner.

Constraint conditions can be defined to control exactly when your constraint will be triggered. Those conditions can be set up such that multiple conditions need to be

satisfied for triggering (i.e. logical AND) or such that satisfaction of any particular condition will trigger the constraint (i.e. logical OR).

Through use of constraint-based generation, you have more flexibility to dynamically control the raising of integration events as well as the generation of the Show SO message. A detailed setup example can be found in the next section.

Please refer to *Oracle Order Management Implementation Manual* for more detailed information on how to setup constraints.

The following is an example of how to set up constraints to raise an Integration event if the salesperson on an order changes:

1. Navigate to the Processing Constraints window using Order Management Super User responsibility (Setup => Rules => Security => Processing Constraints).
2. Create the constraint. Query the constraints for the Application 'Oracle Order Management' and the Entity 'Order Header.'

Processing Constraints

Application: **Oracle Order Management**
Entity: **Order Header**

Constraints

Operation	Attribute	User Action	Applies To	System Changes	User Changes	Enabled	System
UPDATE	Salesperson	Raise Integration Event				<input checked="" type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>

Conditions **Applicable To**

Group #	Scope	Validation Entity	Record Set	Not	Validation Template	Enabled	System
1	Scope	Salesperson	Record Set	<input checked="" type="checkbox"/>	Validation Template	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
				<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>

User Message:

3. Navigate to the Constraints section of the window and enter a new constraint with the following specifications:

Operation => **'UPDATE'** (this field specifies which operation will fire your constraint... i.e. INSERT, UPDATE, or DELETE)

Attribute => **'Salesperson'** (this field restricts your constraint to fire when the operation occurs on a particular attribute)

User Action => **'Raise Integration Event'** (this field specifies what will happen if your constraint is fired)

Enabled => *checked*

Application: **Oracle Order Management**
Entity: **Order Header**

Constraints

Operation	Attribute	User Action	Applies To	System Changes	User Changes	Enabled System
Delete		Not Allowed				<input checked="" type="checkbox"/>
Delete		Not Allowed	Negotiation			<input checked="" type="checkbox"/>
Update	Salesperson	Raise Integ...				<input checked="" type="checkbox"/>

Conditions **Applicable To**

Group #	Scope	Validation Entity	Record Set	Not	Validation Template	Enabled System
1	Any			<input type="checkbox"/>		<input checked="" type="checkbox"/>
2				<input type="checkbox"/>		<input type="checkbox"/>
3				<input type="checkbox"/>		<input type="checkbox"/>

User Message:

Once this constraint has been saved, any updates to the Header-level Salesperson on an order will raise an integration event.

The constraints framework also provides the flexibility to restrict this behavior to orders that satisfy particular conditions. For instance, you could set up your constraint so you only raise integration events for Salesperson changes for the customer "Computer Service and Rentals." To do this you would need to define a condition on your constraint and, consequently, an appropriate validation template.

To create the necessary validation template:

1. Navigate to the Validation Templates window (Setup => Rules => Security => Validation Templates). Then create a new template with the following specifications:

Entity => 'Order Header'

Template Name => 'Customer Restriction Template'

Short Name => 'CUSTRES'

Description => 'Customer Restriction Template'

Validation Type => 'TBL'

Validation Templates

Application **Oracle Order Management**

Validation Templates

Entity	Template Name	Seeded?	Short Name	Description	Validation Type		
					WF	API	TBL
Order Header	Customer Restriction	<input type="checkbox"/>	CUSTRES	Customer Restriction Tem	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Validation Semantics

Column	Validation Operation	Value String
	= (Equal To)	

1. Designate the Validation Type to be 'TBL', and enter the following information in the Validation Semantics section of the window:

Column => 'Customer'

Validation Operation => '= (Equal To)'

Value String => 'Computer Service and Rentals'

Validation Templates

Application **Oracle Order Management**

Validation Templates

Entity	Template Name	Seeded?	Short Name	Description	Validation Type		
					WF	API	TBL
Order Header	Customer Restriction	<input type="checkbox"/>	CUSTRES	Customer Restriction Tem	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
		<input type="checkbox"/>			<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Validation Semantics

Column	Validation Operation	Value String
Customer	= (Equal To)	Computer Service and Rentals

1. After saving the validation template, you must run the Generate Constraints Validation Packages concurrent program to make your validation template available in the Constraints window (Setup => Rules => Security => Generate Constraints Validation Packages).

Navigator - Order Management Super User, Vision Operations (USA)

Functions Documents Processes

Setup:Rules:Security:Generate Constraints Validation Packages

- + Shipping
- + Customers
- Rules
 - Security
 - Processing Constraints
 - Validation Templates
 - Record Sets
 - Generate Constraints Validation Packages**
 - Defaulting
 - Credit
- + UOM
- + QuickCodes
- + Tax
- Profiles
- Shipping Tolerances
- + Documents
- + Transaction Types

Top Ten List

1. Sales Orders
2. Release Sales Orders
3. Shipping Transactions Form
4. ONT Transaction Types
5. Blanket Sales Agreements
6. Quick Sales Orders
7. Quote
8. Shipping Reports and Documents
9. Transact Move Orders
0. Bills of Material

Open

After the concurrent process has completed successfully, navigate back to the Constraints window and query your Salesperson constraint. To add a Condition that utilizes your newly created validation template by entering the following information:

Group => '101'

Scope => 'Any'

Validation Entity => 'Order Header'

Validation Template => 'Customer Restriction Template'

Enabled => *checked*

User Message => 'An integration event is being raised as a result of an UPDATE to Salesperson'

Processing Constraints

Application: **Oracle Order Management**
Entity: **Order Header**

Constraints

Operation	Attribute	User Action	Applies To	System Changes	User Changes	Enabled	System
Update	Salesperson	Raise Integ...				<input checked="" type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>

Conditions **Applicable To**

Group #	Scope	Validation Entity	Record Set	Not	Validation Template	Enabled	System
101	Any	Order Header	Order	<input type="checkbox"/>	Customer Restriction Template	<input checked="" type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>

User Message
An integration event is being raised as a result of an UPDATE to Salesperson

After this condition is saved, only updates to the Salesperson on orders for Computer Service and Rentals will raise an integration event. If you wanted to add another restriction, for instance that the raise only occur after booking, you would simply add another condition. A seeded validation template called 'Booked' already exists and can be used for this purpose. Note that the Group # of the additional condition should be the same as the Group # of the previously added template. This will cause the conditions to be evaluated together as an AND condition. Differing Group #s would cause the conditions to be treated separately as an OR condition.

Processing Constraints

Application: **Oracle Order Management**
Entity: **Order Header**

Constraints

Operation	Attribute	User Action	Applies To	System Changes	User Changes	Enabled	System
Update	Salesperson	Raise Integ...				<input checked="" type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>
						<input type="checkbox"/>	<input type="checkbox"/>

Conditions **Applicable To**

Group #	Scope	Validation Entity	Record Set	Not	Validation Template	Enabled	System
101	Any	Order Header	Order	<input type="checkbox"/>	Customer Restriction Templa	<input checked="" type="checkbox"/>	<input type="checkbox"/>
101	Any	Order Header	Order	<input type="checkbox"/>	Booked	<input checked="" type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>

User Message: **This is a booked order**

Once the new condition is saved, integration events will only be raised for updates to the salesperson on booked orders for Computer Service and Rentals.

Change_SalesOrder

Setup

Customer Setup

Setup Steps

Step	Description
1	Define Transaction
2	Message Setup
3	Define Trading Partner/Hub
4	Define Code Conversion Values
5	Set the Profile Options

Step	Description
6	Loading of Message Map and DTD
7	Enabling and Consuming the Business Event

Define Transaction

Use Oracle XML Gateway to define a cross reference between the Oracle transaction name and the external transaction name. The external transaction name will base on what is meaningful per the XML standard used by the recipient. The external transaction name will appear on the message envelope to support message transport.

Message Setup

To implement a message with a Trading Partner, use the XML Gateway message set up to define the Trading Partner or hub, code conversion values, and internal to external transaction name cross references. In addition, you may identify the XML Gateway system administrator to notify for system or process errors. Assign the Notify PO Update transaction to the trading partner.

Define Trading Partner/Hub

E-Business may be conducted directly with a business partner commonly known as a Trading Partner or via a hub such as Oracle Exchange where many buyers and sellers converge to conduct electronic commerce.

With Oracle XML Gateway, you can define the hub or the actual business partner as a Trading Partner. If you define the hub as the Trading Partner, you can identify all the buyers and sellers who are conducting business on the hub as Trading Partners to the hub.

Included in this definition are the following:

- Trading Partner/Hub name
- Message enabled
- Message confirmation requested
- Message map to use for message creation or consumption
- E-mail address of Trading Partner contact to notify for data errors
- Trading Partner specific code conversion values

- Transport protocol - SMTP, HTTP, HTTPS with credential and username/password as necessary

Please refer to the Oracle XML Gateway documentation for further details.

To define a trading partner:

1. Navigate to the Define Trading Partner Setup window from the XML Gateway Responsibility. Setup > Define Trading Partners.
2. Enter the Trading Partner Type, Customer.
3. Select the Trading Partner and Trading Partner Site.
4. Select the transaction type as 'ONT' for this functionality.
5. Select 'ONT_3A7R_OAG72_OUT_SO' as the map.
6. Save your work.

Define Code Conversion Values

With Oracle XML Gateway, you can create a cross reference between an Oracle code to a meaningful concept to the recipient or vice versa. Common examples are units of measure and currency code.

Code conversion values can be defined to be universal for all Trading Partners and all messages. Additionally, code conversion values may be defined for a specific XML standard or specific to a Trading Partner.

Note: If consuming 3A8 Change PO Response XML documents sent by Oracle Purchasing, you must convert the value Response to the Oracle value of Y using the seeded Code Category DOC_PURPOSE_CODE. This ensures that the column Response_Flag in OE_HEADERS_INTERFACE is populated correctly.

Set the Profile Options

All these profile options are site level.

Set the 'OM: Change SO Response Required' the default value for this profile is No.'

Set the value of the profile ONT_3A7_RESPONSE_REQUIRED to either Yes or No. Based on this profile the system will decide whether to put the order line on hold after the changes are triggered and then wait for either a 3A8 response or a 3A9.

Set the 'OM: Order Accept State for XML' profile.

Based on this profile the system decides what status is sent on the outbound Acknowledgement. The default value of this profile is Entered.

This is also governed by the value in the tag <Booked Flag> that is mapped in the inbound transactions. The value in this tag determines whether the order is Booked or not when it is created. The tag will have a value of Y or N.

Set the 'OM: Send Acknowledgment for Change PO Response' profile.

Set the value of the profile ONT_3A8_RESPONSE_ACK to either 'Yes' or 'No' (default value is 'No').

Based on this profile the system will determine whether or not an Acknowledge PO message is sent when a 3A8 Change PO response message is received. This profile does not govern the behavior Acknowledge PO for normal 3A8 Change PO (i.e. non-response) messages.

Loading of Message Map and DTD

3A7 transaction message maps are delivered and installed as part of Oracle Order Management file structure and schema. They are automatically loaded into the XML Gateway repository using the LoadMap program. The reports are available to verify the success of the loading process.

Profile Options

Profile Option Name	User Profile Name	Description
ONT_3A7_RESPONSE_REQ UIRED	OM: Change SO Response Required	To determine if the Seller should expect a response to an outbound Change SO XML message
ONT_3A8_RESPONSE_ACK	OM: Send Acknowledgment for Change PO Response	Determines whether an Acknowledgment will be sent for the Change PO Response
ONT_XML_ACCEPT_STATE	OM: Order Accept State for XML	Determines the order Accept status for outbound XML messages

The following table determines the status send on the Outbound Acknowledgment.

Status Send on the Outbound Acknowledgement

Profile	Booked Flag	Order Status	Acknowledgement Status
BOOKED	Y	Book the Order.	Accept
		Order in Booked state	Pending
		Order in Entered state	
BOOKED	N	Don't book the Order	Pending
		Order in Entered state	
ENTERED	Y	Book the Order	Accept
		Order in Booked state	'Accept'
		Order in Entered state	
ENTERED	N	Don't book the Order	Accept
		Order in Entered state	

Enabling and Consuming the Business Event

The necessary business events and subscriptions are also delivered and installed as part of the schema. They are loaded by the driver.

Data Archive and Purge Procedures

Purge Data From the Acknowledgment Tables

Acknowledgment data is deleted from the Acknowledgment tables after the extraction. Data is left in the tables in case the outbound transaction fails.

Change_PO

Setup: Customer Setup

Setup Steps

Step	Description
1	Define Transaction Type 'ONT' and Subtype 'CHO'
2	Define Trading Partners
3	Define the Subscribing System for the event subscription for event 'oracle.apps.ont.oi.po_inbound.create'
4	Message Set Up
5	Define Subscription for the Event 'oracle.apps.ont.oi.po_inbound.create'
6	Define System Profile Values

Define Transaction Type 'ONT' and Subtype 'CHO'

Use Oracle XML Gateway to define a cross reference between the Oracle transaction name and the external transaction name. The external transaction name will be based on what is meaningful per the XML standard used by the recipient. The external transaction name will appear on the message envelope to support message transport.

The transaction type ONT and subtype CHO is seeded for this functionality.

Navigate to the Define Transactions window from the XML Gateway Responsibility by selecting Setup /rt.jar:\$Workflow_JAR_file_directory/wfjava.jar: \$ Workflow_JAR_file_directory/jdbc/lib/classes111.zip:" oracle.apps.fnd.wf.WFXLoad -d use r connect_string lang objectJava

Define Trading Partners

The Trading Partner Setup window is used to:

Enable messages for the trading partner by identifying the internal and external transaction type and transaction subtype codes, and the XML standard associated with

the message.

Access the Trading Partner Code Conversion form.

Select a message map for the trading partner.

Identify the communications protocol and address for a message. Optionally, the user can be selected from a hub.

Navigate to the Define Trading Partner Setup window from the XML Gateway Responsibility by selecting Setup DB password:Port mymap.xgm DB password:Port mapname Define XML Standards.

Included in the trading partner/hub definition are the following:

- Trading Partner/Hub name
- Message confirmation requested
- Message map to use for message creation or consumption
- E-mail address of trading partner contact to notify for data errors
- Trading Partner specific code conversion values
- Transport protocol - SMTP, HTTP, or HTTPS with credential and username/password as necessary
- Enable messages for the trading partner by identifying the internal and external transaction type and transaction subtype codes, and the XML standard associated with the message.

To define a trading partner:

1. Navigate to the Define Trading Partner Setup window from the XML Gateway Responsibility. Setup > Define Trading Partners.
2. Enter the Trading Partner Type, Customer.
3. Select the Trading Partner and Trading Partner Site.
4. Select the transaction type as 'ONT' for this functionality.
5. Select 'ONT_3A8R_OAG72_IN' as the map.
6. Save your work.

Define the Subscribing System for the event subscription for event 'oracle.apps.ont.oi.po_inbound.create'

Subscriber: System - This will usually be the local system, the system on which the

subscription code is to run.

(This data will be available as the initial setup, it will be an XML file and is loaded to the customer system. The user of the system can change it to work for their system settings).

Define Subscription for the Event 'oracle.apps.ont.oi.po_inbound.create'

Create a subscription for the event 'oracle.apps.ont.oi.po_inbound.create'. The event is raised in the Post Process by the XML Gateway engine after the Inbound XML Message is consumed and the data is loaded in the OM Interface tables. This subscription will then consume this event and in turn run a pre-defined workflow.

Use the Oracle XML Workflow Loader utility to download and upload XML definitions for Business Event System objects. When you download Business Event System object definitions from a database, Oracle Workflow saves the definitions as an XML file. This XML file will be delivered to the Customer as part of the Seed Data.

On UNIX, use the following commands to run the Workflow XML Loader:

```
jre -classpath "$JREPATH>/rt.jar:$Workflow_JAR_file_directory>:  
$Workflow_JAR_file_directory>/wfjava.jar: $Workflow_JAR_file_directory>/wfapi.jar:  
$ORACLE_HOME>/jdbc/lib/classes111.zip:" oracle.apps.fnd.wf.WFXLoad -d  
user>password> connect_string> protocol> lang> output_file> object> key>Java
```

Define System Profile Values

The following table lists the System Profile options, which need to be set.

System Profile Values

Profile System Option	Description	Required	Default Value
OM: Run Order Import for XML	Determines whether Order Import will run Asynchronously or Synchronously	No	Synchronous

The directories listed in the above profiles should also be mentioned in the utl_file_dir parameter in init.ora

Please refer to the Oracle XML Gateway User's Guide for the setup information.

Message Set Up

To implement a message with a trading partner, use XML Gateway message set up to define the trading partner or hub, code conversion values, and internal to external

transaction name cross references. In addition, you may identify the XML Gateway system administrator to notify for system or process errors.

Pre-Defined Workflow Event

A pre-defined Workflow event is seeded to run Order Import immediately if the Profile Option OM: Run Order Import for XML is set to 'Synchronous.' This Workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the Workflow Process it will run Order Import for the XML Message, which is currently being consumed.

Implementation Considerations

Loading of Message Map and DTD

Oracle message maps are delivered and installed as a part of Oracle Order Management file structure and schema. They are automatically loaded into the XML Gateway repository using the LoadMap program. The reports are available to verify the success of the loading process.

Enabling and Consuming the Business Event

The necessary business events and subscriptions are also delivered and installed as part of the schema. They are loaded by the driver.

To install this functionality, the required prereq patch for the XML gateway should be installed first.

Message Map

Most data values in the message map are seeded.

Message Map 'ONT_3A8 R_OAG72_IN.xgm'

The message map 'ONT_3A8R_OAG72_IN' is created using the Oracle XML Gateway Message Designer tool.

The source DTD used is 058_cancel_po_006.dtd, revision 7.2.1 of the Open Application Group. The other associated external reference DTD files are:

oagis_domains.dtd

oagis_resources.dtd

oagis_fields.dtd

oagis_segments.dtd

oagis_extensions.dtd

oagis_entity_extensions.dtd

All the DTD's will be checked in ONT source area under \$ont/xml/oag72.

The target for the Inbound XML Message are the Order Management Open Interface tables OE_HEADERS_INTERFACE & OE_LINES_INTERFACE.

The INBOUND CHANGE_PO DTD is a three level hierarchy with Order Lines split into one or more Shipment Lines. The Order Management architecture is however a two level one with Order Header and one or more Lines. The message map will collapse the three level XML Message into a two level structure when the data is inserted in the Order Management Open Interface tables.

Both the message map created using the Message Designer and its associated DTD's are stored in the database. The following Java programs are available to Load/Delete Maps or Load/Delete DTD's into/from the XML Gateway repository. Please refer to the *Oracle XML Gateway Manual* for more information.

Load/Delete Maps, Load/Delete DTD's

The following process is used only for customizations.

```
java LoadMap DB username> DB password> Hostname>:Port>:SID> mymap.xgm>
```

Example: java oracle.apps.ecx.loader.LoadMap apps apps ap505dbs:1521:dev115
ONT_3A8R_OAG72_IN.xgm

1. java DeleteMap DB username> DB password> Hostname>:Port>:SID> mapname>

Example: java oracle.apps.ecx.loader.DeleteMap apps apps ap505dbs:1521:dev115
ONT_3A8R_OAG72_IN.xgm

The Message Map is a .xgm file which will be stored in the Order Management Source area under \$ont/patch/115/map/ ONT_3A8R_OAG72_IN.xgm

Note: Maps and DTD's must be kept in sync between the Message Designer and the XML Gateway repository. When in doubt, always reload the map and DTD as a pair.

key: H => OE_HEADERS_INTERFACE, L => OE_LINES_INTERFACE

Mapping the Document to the Order Management Table

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
<POHEADER >							
POID	H.CUSTOMER_PO_NUMBER	Y					PO_ECX_HEADER_ARCH_V.contractb PHA.SEGMENT1
	H.Creation_DATE	Y			SYSDATE	Execute Function Call	
	H.Last_UPDATED_BY	Y			FND_GLOBAL.LOGIN_ID	Execute Function Call	
	H.Last_UPDATE_DATE	Y			SYSDATE	Execute Function Call	
	H.Creation_BY	Y			FND_GLOBAL.USER_ID	Execute Function Call	

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
	H.ORDR_SOURCE_ID	Y			20	Get Pre-Defined Variable Value	
	H.XML_TRANSACTION_TYPE_CODE	Y			'CHO'	Get Pre-Defined Variable Value	
	H.XML_MESSAGE_ID	Y				Get Pre-Defined Variable Value (Internal Control Number)	
	H.OPERATION_CODE	Y			'UPDATE'		
</USERAREA>							
ORGID	H.ORG_ID					Derive Target Org Id using Sold To Org Id (if not sent)	
ORACLE.CHANGESEQUENCE	H.CHANGE_SEQUENCE						

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
ORACLE.BOOKEDFLAG	H.BOOKED_FLAG						
MSGTYPE	H.RESPONSE_FLAG						
ORACLE.SUPPLIERDOCUMENT REF	H.ORIGINAL_SYSTEM_DOCUMENT_REF						
</USERAREA>							
<PARTNER> (Supplier)							
NAME							PO_ECX_HEADER_ARCH_V.supplier_org_name PO_VENDORS.VENDOR_NAME

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
ONETIME							PO_ECX_HEADER_ARCH_V.supp_otf DECODE (PO_VENDORS.ONE_ TIME_FLAG,'N','0','1')
PARTNRID							PO_ECX_HEADER_ARCH_V.supplier_partner_id PO_VENDORS.SEGMENT1
PARTNRTYPE							
CURRENCY							PO_ECX_HEADER_ARCH_V.supp_Currency PO_VENDORS.PAYMENT_CURRENCY_CODE

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
PARTNRIDX							PO_ECX_HEADER_ARCH_V.supplier_partner_id_x NVL(PO_VENDOR_SITE_ALL_EC TE_TP_LOC ATION_CODE,PHA.VENDOR_ID
ADDRESS1							.
ADDRESS2							.
ADDRESS3							.
ADDRESS4							.
CITY							.
COUNTY							.
POSTALCODE							.
STATEPROVIN							.
</PARTNER>							.

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
<PARTNER> (SoldTo)		Y					
NAME							PO_ECX_HEADER_ARCH_V.buying_org_name HR_ALL_ORGANIZATION_UNITS_TL.NAME
ONETIME							PO_ECX_HEADER_ARCH_V.buying_org_ottf '0
PARTNRID							
PARTNRTYPE							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
CURRENCY							PO_ECX_HEADER_ARCH_V.buying_org_currency GL_SETS_OF_BOOKS.CURRENCY_CODE
PARTNRIDX	H.SOLD_TO_ORG_ID	Y		OE_XML_PROCESSTIL.GET_SOLD_TO_ORG_ID		Derive Address Id from Location Code Execute Procedure Call	PO_ECX_HEADER_ARCH_V.partner_id_x HR_LOCATIONS_ALL.ECE_TP_LOCATION_CODE
ADDRESS1							HR_ORGANIZATION_UNITS_ V.ADDRESS_LINE_1-3
ADDRESS2							
ADDRESS3							
ADDRESS4							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
CITY							HR_ORGANIZATION_UNITS_ V.TOWN_OR_CITY
COUNTRY							PO_ECX_HEADER_ARCH_V.buying_ org_country
COUNTY							
POSTALCODE							HR_ORGANIZATION_UNITS_ V.POSTAL_CODE
STATEPROVIN							HR_ORGANIZATION_UNITS_ V.REGION2
TELEPHONE							HR_ORGANIZATION_UNITS_

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
							V.TELEPHONE_NUMB ER_1-3
	/PARTNER						
	/PARTNER						
	<PARTNER> (BillTo)						
	NAME						PO_ECX_HEADERS_AR CH_V.buying_org_name HR_ALL_ORGANIZATION_UNITS _TL.NAME
	ONETIME						PO_ECX_HEADERS_AR CH_V.billto -
							org_otf '0
	PARTNRID						PO_ECX_HEADERS_AR CH_V.partner_id PHA.ORG_ID

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
PARTNRTYPE							
CURRENCY							FINANCIALS_SYSTEM_PARAMS_ ALL.PAYMENT_CURRENCY_CODE
PARTNRIDX	H.INVOICE_TO_ORG_ID			OE_XML_PROCESS_UTIL.GET_BILL_TO_ORG_ID		Derive Address Id from Location Code Execute Procedure Call	HR_LOCATIONS_ALL.ECE_TP_ LOCATION_CODE
ADDRESS1	H.INVOICE_ADDRESS1						HR_LOCATIONS_ALL.ADDRESS_ LINE_1-3
ADDRESS2	H.INVOICE_ADDRESS2						
ADDRESS3	H.INVOICE_ADDRESS3						
ADDRESS4							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
CITY	H.INVOICE_CITY						HR_LOCATIONS_ALL.TOWN_OR_CITY
COUNTRY	H.INVOICE_COUNTRY						HR_LOCATIONS_ALL.COUNTRY
COUNTY	H.INVOICE_COUNTY						
POSTALCODE	H.INVOICE_POSTALCODE						HR_LOCATIONS_ALL.POSTAL_CODE
STATEPROVIN	H.INVOICE_STATE						HR_LOCATIONS_ALL.REGION_2
TELEPHONE							HR_LOCATIONS_ALL.TELEPHONE_ NUMBER_1-3
</PARTNER>							
<PARTNER> (ShipTo)							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
NAME							
ONETIME							
PARTNRID							
PARTNRTYPE							
CURRENCY							
PARTNRIDX	H.SHIP_TO_ORG_ID			OE_XML_PROCESS_UTIL.GET_SHIP_TO_ORG_ID		Derive Address Id from Location Code Execute Procedure Call	
ADDRESS1	H.SHIP_TO_ADDRESS1						
ADDRESS2	H.SHIP_TO_ADDRESS2						
ADDRESS3	H.SHIP_TO_ADDRESS3						
ADDRESS4							

Source	Target	Re q.	Code Catego ry	Derivati on Rule	Defaul t Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
CITY	H.SHIP_TO_CITY						
COUNTY	H.SHIP_TO_COUNTY						
POSTALCODE	H.SHIP_TO_POSTAL_CODE						
STATEPROVIN	H.SHIP_TO_STATE						
</PARTNER>							
<POTERM>							
DESCRIPTN							PO_ECX_HEADER_ARCH_V.payment_terms_description AP_TERMS.DESCRPTION

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
TERMDID	H.PAYMENT_TERM		PAYMENT_TERMS				PO_ECX_HEADER_ARCH_V.paymente t_terms_name AP_TERMS.NAME
DAYSNUM							No mapping provided by PO although generated XML has these elements.
</POTERM>							
/POHEADER							
<POLINE>							
ShipTo) ="UNIT" type="T">	L.CUSTOMER_ITEM_PRICE					Convert from OAG Operating Amount	

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
VALUE	L.CUSTOMER_TEMPERATURE					Convert from OAG Operating Amount	PO_ECX_LINE_ARCHIVE_ALL.UNIT_PRICE
NUMOFDEC	L.CUSTOMER_TEMPERATURE					Convert from OAG Operating Amount	
SIGN	L.CUSTOMER_TEMPERATURE					Convert from OAG Operating Amount	
CURRENCY							PO_ECX_HEADER_ARCHIVE_V.po_currency PHA.CURRENCY_CODE
UOMVALUE							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
UOMNUMDEC							
UOM							
</OPERAMT>							
<QUANTITY qualifier="ORDERED">							
VALUE							PO_ECX_LINE_ARCHIVE_V.quantity PO_LINES_ARCHIVE_ ALL.QUANTITY
</QUANTITY>							
REASONCODE							PO_LINES_ARCHIVE_ ALL.LINE_NUM

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
POLINENUM	L.CUSTOMER_LINE_NUMBER	Y					PO_ECX_LINE_ARCH_V.line_num
	L.CREATION_DATE	Y			SYSDATE	Execute Function Call	
	L.LAST_UPDATED_BY	Y			FND_GLOBALS.GIN_ID	Execute Function Call	
	L.LAST_UPDATE_DATE	Y			SYSDATE	Execute Function Call	
	L.CREATED_BY	Y			FND_GLOBALS.USER_ID	Execute Function Call	
	L.ORDER_SOURCE_ID	Y			20	Get Pre-Defined Variable Value	
	L.XML_TRANSACTION_TYPE_CODE	Y			'CHO'	Get Pre-Defined Variable Value	

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
	L.CALCULATE_PRICE_FLAG				'Y' if L.CUSTOMER_TOLERANCE_ITEM_NAME is passed		PO_LINES_ARCHIVE_ALL_ITEM_DESCRIPTION
	DESCRIPTION						PO_ECX_LINE_ARCHIVE_V.description
	ITEMRV						PO_ECX_LINE_ARCHIVE_V.itemrv PO_LINES_ARCHIVE_ALL_ITEM_REVISION
ITEM	L.CUSTOMER_ITEM_NAME						PO_ECX_LINE_ARCHIVE_V.item MTL_SYSTEM_ITEMS_B — KFV.SEGMENT1

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
ITEMX							PO_ECX_LINE_ARCHIVE_V.itemx PO_LINES_ARCHIVE_ ALL.VENDOR_PRODUCT_NUM
NOTES index="1"	L.CHANGE_COMMENT NTS						PO_ECX_LINE_ARCHIVE_V.note_to_vendor PO_LINES_ARCHIVE_ ALL.NOTE_TO_VENDOR
<USERAREA >							
ORACLE.SUPPLIERLINE REF	H.ORIG_SYS_LINE_REF						
</USERAREA >							
<SCHEDULE >							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
<DATETIME qualifier="NE EDDELV">	L.REQUIRE EST_DATE TE					Convert from OAG Datetime	PO_ECX_LINE_LOCATION_ARCHIVE_ALL.NEED_BY_DATE
YEAR						Convert from OAG Datetime	
MONTH						Convert from OAG Datetime	
DAY						Convert from OAG Datetime	
HOUR						Convert from OAG Datetime	
MINUTE						Convert from OAG Datetime	
SECOND						Convert from OAG Datetime	
SUBSECOND						Convert from OAG Datetime	
TIMEZONE						Convert from OAG Datetime	

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
</DATETIME>							
>							
= "ORDERED"	L. ORDER_QUANTITY					Convert from OAG Quantity	PO_ECX_LINE_LOCATION_S_ARCHIVE_ALL.QUANTITY - PO_LINE_LOCATION_S_ARCHIVE_ALL.QUANTITY_CANCELLED)
>							
VALUE						Convert from OAG quantity	
NUMOFDEC						Convert from OAG quantity	
SIGN						Convert from OAG quantity	
UOM	L. ORDER_QUANTITY_UOM		UOM			Convert from OAG quantity	

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
</QUANTITY>							
PSCLINENUM	L.CUSTOMER_SHIPMENT_NUMBER	Y					
REASONCODE	L.CHANGE_REASON				'Not provided'		
<USERAREA>							
PARTNER (BillTo)							
NAME							
ONETIME							
PARTNRID						Execute Procedure Call	
PARTNRTYPE							
CURRENCY							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
PARTNRIDX	L.INVOICE_TO_ORG_ID			OE_XML_PRO_CESS_UTL.GET_BILL_TO_ORG_ID		Derive Address Id from Location Code	
ADDRESS1	L.INVOICE_ADDRESS1						
ADDRESS2	L.INVOICE_ADDRESS2						
ADDRESS3	L.INVOICE_ADDRESS3						
ADDRESS4							
CITY	L.INVOICE_CITY						
COUNTRY	L.INVOICE_COUNTRY						
COUNTY	L.INVOICE_COUNTY						

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
POSTAL CODE	L.INVOICE_POS TALCODE						
STATE PROVN	L.INVOICE_STATE						
</PARTNER>							
<PARTNER> (ShipTo)							
NAME							PO_ECX_LINE_LOC_ARCH_V.Ship to_org_name HR_ORG NIZATION _UNITS_V .NAME
ONETIME							

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
PARTNRID							PO_ECX_LINE_LOC_ARCHIVE_V.partner_id PO_LINE_LOCATIONS_ARCHIVE_ALL.ORG_ID
PARTNRTY PE							PO_ECX_LINE_LOC_ARCHIVE_V.ship_to_org_partner_type 'ShipTo'
CURRENCY							
PARTNRIDX	L.SHIP_TO_ORG_ID			OE_XML_PROCESS_UTIL.GET_SHIP_TOORG_ID		Derive Address Id from Location Code Execute Procedure Call	HR_LOCATIONS_ALL.ECE_TP_LOCATION_CODE
ADDRESS1	L.SHIP_TO_ADDRESS1						HR_LOCATIONS_ALL.ADDRESS_LINE_1,2,3
ADDRESS2	L.SHIP_TO_ADDRESS2						

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
ADDRESS3	L.SHIP_TO_ADDRESS3						
ADDRESS4							
CITY	L.SHIP_TO_CITY						HR_LOCATIONS_ALL.TOWN_OR_CITY
COUNTRY	L.SHIP_TO_COUNTRY						HR_LOCATIONS_ALL.COUNTRY
COUNTY	L.SHIP_TO_COUNTY						
POSTAL_CODE	L.SHIP_TO_POSTAL_CODE						HR_LOCATIONS_ALL.POSTAL_CODE
STATE PROVN	L.SHIP_TO_STATE						HR_LOCATIONS_ALL.REGION_2
TELEPHONE							HR_LOCATIONS_ALL.TELEPHONE_NUMBER 1,2,3

Source	Target	Req.	Code Category	Derivation Rule	Default Value	Action	PO Mapping
CHANGE_PO	OE_HEADERS_INTERFACE (H) OE_LINES_INTERFACE (L)						
</PARTNER>							
OPERATION	L.OPERATION_CODE		ACTION_CODE	OE_XM L_PRO CESS_U TIL.DE RIVE_LI NE_OP ERATIO N_COD E if not passed			
ORACLE.SUPPLIERSHIPMENTREF	L.ORIG_SYS_SHIPMENT_REF						
</USERAREA>							
</SCHEDULE>							
</POLINE>							

Sample Business Flow for 3A4 Process PO/3A4 Acknowledge PO

Sample business flow for 3A7 Change SO/3A8 Change PO response

Step	Buyer System E.G. Oracle Purchasing	Oracle Order Management
0		<p>Figure, "Trading Partner Setup completed for Oracle Order Management via XML Gateway forms"</p> <p>Figure, "Profile Option OM: Change SO Response Required set to 'Yes' at site-level"</p>
1	Figure, "Buyer approves a Purchase Order"	
2		<p>Figure, "Order Management receives the Purchase Order electronically as a Process PO XML document and a corresponding sales order is createdHeader"</p> <p>Figure, "Order Management receives the purchase order electronically as a Process PO XML document and a corresponding sales order is created Line"</p>
3		<p>Figure, "After booking the order, the seller modifies the ordered quantity on one of the lines"</p> <p>Figure, "The order line goes on hold"</p> <p>Figure, "3A7 Change SO XML document is generated and sent to the buyer"</p> <p>Figure, "Results Summary Outbound Message Details"</p>

Step	Buyer System E.G. Oracle Purchasing	Oracle Order Management
4	<p>Figure, "Buyer receives seller-initiated changes"</p> <p>Figure, "Buyer sends a 3A8 Change PO response acknowledging the changes"</p>	
5		<p>Figure, "Seller receives 3A8 Change PO response"</p> <p>Figure, "Results Summary Inbound Message Details"</p> <p>Figure, "3A8 Change PO response is processed successfully and the hold on the sales order line removed"</p>

Trading Partner Setup

Trading Partner Type: **Customer**

Trading Partner Name: **XMLCUST02**

Trading Partner Site: **Cyber XMLCUST02A Chattanooga TN 37401**

Company Admin Email: **rparamat@oracle.com**

Code Conversion

Trading Partner Details

Transaction Type	Transaction SubType	Standard Code	External Transaction Type	External Transaction SubType	Direction Map	Connection/Hub	Protocol Type
ECX	CBODO	OAG	BOD	CONFIRM	OUT	ECX_CBODO	DIRECT SMTP
ONT	POA	OAG	PO	ACKNOWLED	OUT	ONT_3A4A_O	DIRECT SMTP
ONT	CP0	OAG	PO	CANCEL	IN	ONT_3A9R_O	
ONT	CH0	OAG	PO	CHANGE	IN	ONT_3A8R_O	
ONT	POI	OAG	PO	PROCESS	IN	ONT_3A4R_O	
ONT	CS0	OAG	SALESORDE	CHANGE	OUT	ONT_3A7_OA	DIRECT SMTP
ONT	SS0	OAG	SALESORDE	SHOW	OUT	ONT_3A6_OA	DIRECT SMTP

Profile Option OM: Change SO Response Required set to 'Yes' at site-level

Personal Profile Values

Profile Name	Default Value	User Value
OM: Change SO Response Req	Yes	

Buyer approves a Purchase Order

Purchase Orders (Vision Operations) - [New]

PO, Rev	12122	0	Type	Standard Purchase	Created	18-JUN-2004 18:19
Supplier	ABC Order Manag		Site	SUPPLIER SITE	Contact	
Ship-To	V1- New York City		Bill-To	V1- New York City	Currency	USD
Buyer	Stock, Ms. Pat		Status	Incomplete	Total	27,696.25
Description						
P-Card						Transaction Code

Num	Rev	Job	Category	Description	UOM	Quantity
1			PRODUCTN.FIN	Dimension 4550	Each	10
2			PRODUCTN.FIN	Dimension 4550	Each	15

Item AS54888 Dimension 4550

Order Management receives the Purchase Order electronically as a Process PO XML document and a corresponding sales order is created.

Sales Orders (70586) - Vision operations

Order Information Line Items

Firm Order Total **17,885.61**

Main Pricing Shipping Addresses Returns Services Others

Line	Ordered Item	Qty	Blanket Number	UOM	Unit Selling Price	Request Date	Sch
1.1	AS54888	6		Ea	1,117.85	20 JUN 2004 17:30:00	21 J
2.1	AS54888	5		Ea	1,117.85	20 JUN 2004 17:30:00	21 J

Line Total **6,707.10** Line Qty **11**

Description **Dimension 4550**

Actions Related Items Configurator

Define Trading Partners

Enter Reason for ...

Reason **Administrative or procedural c**

Comments

OK Exit

The order line goes on hold.

Sales Orders (70586) - Vision operations

Order Information Line Items

Firm Order Total **18,444.53**

Main Pricing Shipping Addresses Returns Services Others

Line	Ordered Item	Qty	Request Date	Schedule Ship Date	Status	On Hold	Cascad
1.1	AS54888	6	20 JUN 2004 17:30:00	21 JUN 2004 23:59:00	Awaiting Shippin	<input checked="" type="checkbox"/>	
2.1	AS54888	5	20 JUN 2004 17:30:00	21 JUN 2004 23:59:00	Awaiting Shippin	<input type="checkbox"/>	

Line Total **6,707.10** Line Qty **6** Service Total **0.00**

Description **Dimension 4550**

Actions Related Items Configurator Availability Book Order

3A7 Change SO XML document is generated and sent to the buyer.

Open Interface Tracking Details

Customer Number: 6048
 Customer: Vision operations
 Order Source Reference: 12125

Order Source: XML
 Message Count: 4

Main Others

Transaction Type	Status	Message Text
Process PO	SUCCESS	Process PO - Order Import successful
Acknowledge PO	SUCCESS	Acknowledge PO message has been sent.
Confirm BOD	SUCCESS	Confirm BOD message has been sent.
Change SO	SUCCESS	Change SO message has been sent.

WF Monitor XML Monitor Corrections

Results Summary Outbound Message Details window.

Oracle Results Summary > Outbound Message Details

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!-- Oracle eBusiness Backup Language Gateway Screen -->
<DOCTYPE CHANGE_SALESORDER_OOS SYSTEM "os7_change_salesorder_oos.dtd">
<CHANGE_SALESORDER_OOS>
  <CTRLAREA>
    <BUS>
      <VERB value="CHANGE"/>
      <DOCS value="SALESORDER"/>
      <REVISION value="000"/>
    </BUS>
    <SENDER>
      <LOGICALID>dbflem/appscmt/logc/LOGICALID</LOGICALID>
      <COMPONENT>INT</COMPONENT>
      <TASK>CSO</TASK>
      <REFERENCEID>LMD001:oracle.apps.out.os.show_so_create12556</REFERENCEID>
      <CONFIRMATION>2</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APP2</AUTHID>
    </SENDER>
    <DATEIME qualifier="CREATION" index="1">
      <YEAR>2004</YEAR>
      <MONTH>06</MONTH>
      <DAY>21</DAY>
      <HOUR>00</HOUR>
      <MINUTE>01</MINUTE>
      <SECOND>14</SECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATEIME>
    <CTRLAREA>
      <DATAAREA>
        <CHANGE_SALESORDER>
          <CHHEADER>
            <SALESORDERID>70506</SALESORDERID>
            <SALESOOS index="1"/>
            <NOTES index="1">Accepted</NOTES>
            <POID>12125</POID>
            <POSTATUS>OPEN</POSTATUS>
            <USERAREA><ORACLE_SUPPLIERDOCREP>12125</ORACLE_SUPPLIERDOCREP></USERAREA>
          </CHHEADER>
          <PARTNER>
            <NAME index="1">Vision operations</NAME>
            <ONETIME>0</ONETIME>
            <PARTNERID>10291</PARTNERID>
          </PARTNER>
        </CHANGE_SALESORDER>
      </DATAAREA>
    </CTRLAREA>
  </CHANGE_SALESORDER_OOS>
</CHANGE_SALESORDER_OOS>
  
```

Buyer receives seller-initiated changes.

Respond to Change Request - Microsoft Internet Explorer

Address: http://ap6155rt.us.oracle.com:8001/OA_HTML/OA.jsp?page=/oracle/apps/pos/changeorder/webui/PosViewPOPG5_riv=1776CallMode=View6PoHeaderId=%7B11wWo1OutgF...

Shaker Heights, OH 44122

Buyer: **Stock, Ms. Pat**
Order Date: **24-May-2004**
Description: **SP01 for ACK testing**
Note to Supplier:
Attachments: [View](#)

Bill-To Address
90 Fifth Avenue
New York, NY 10022-3422

Ship-To Address
90 Fifth Avenue
New York, NY 10022-3422

PO Details
☐ Indicates cancellation request
☐ Indicates new value

Details	Request Date	Line /	Shipment	Item	Supplier	Promised Date	Start Date	End Date	Amount (USD)	Quantity	Price / Rate (USD)	Split	Request Reason	Distribution Details	Cancel Backing Request
Show	25-May-2004 03:23:53	2	1	OAISP Item2					600.00	60	10	No	Qty splitted		
Show	25-May-2004 03:23:53	2	4	OAISP Item2		29-May-2004 01:00:00				30	10	Yes	Qty splitted		

[View Purchase Order Details](#)

[Cancel](#) [View Change History](#) [Submit](#)

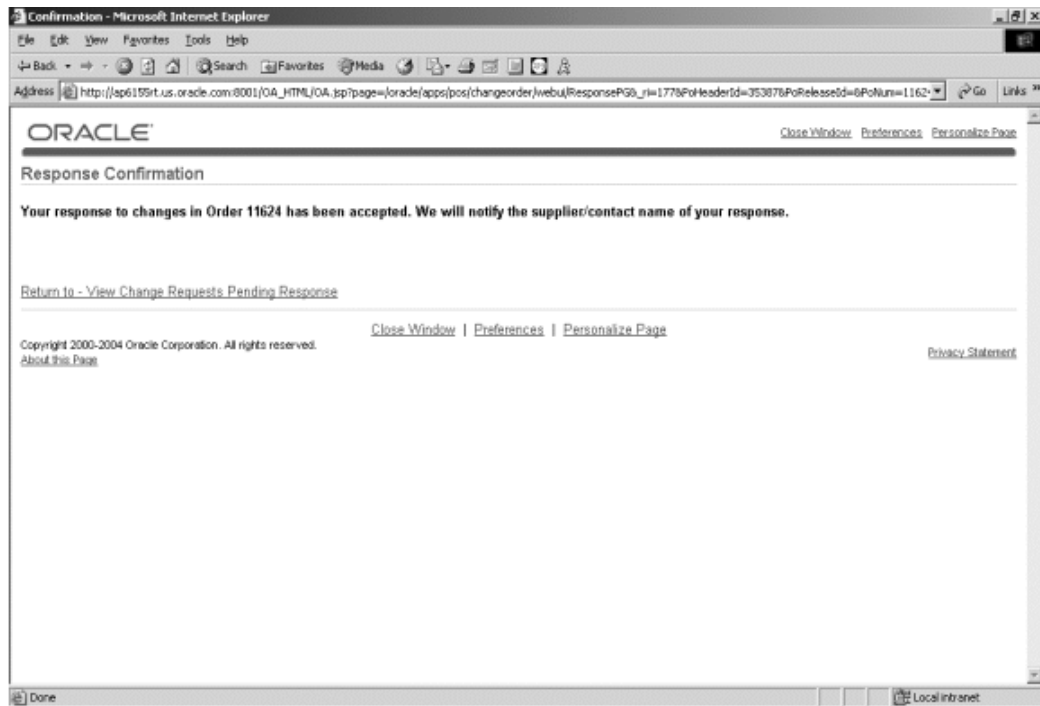
[Close Window](#) | [Preferences](#) | [Personalize Page](#)

Copyright 2000-2004 Oracle Corporation. All rights reserved.
[About this Page](#) [Privacy Statement](#)

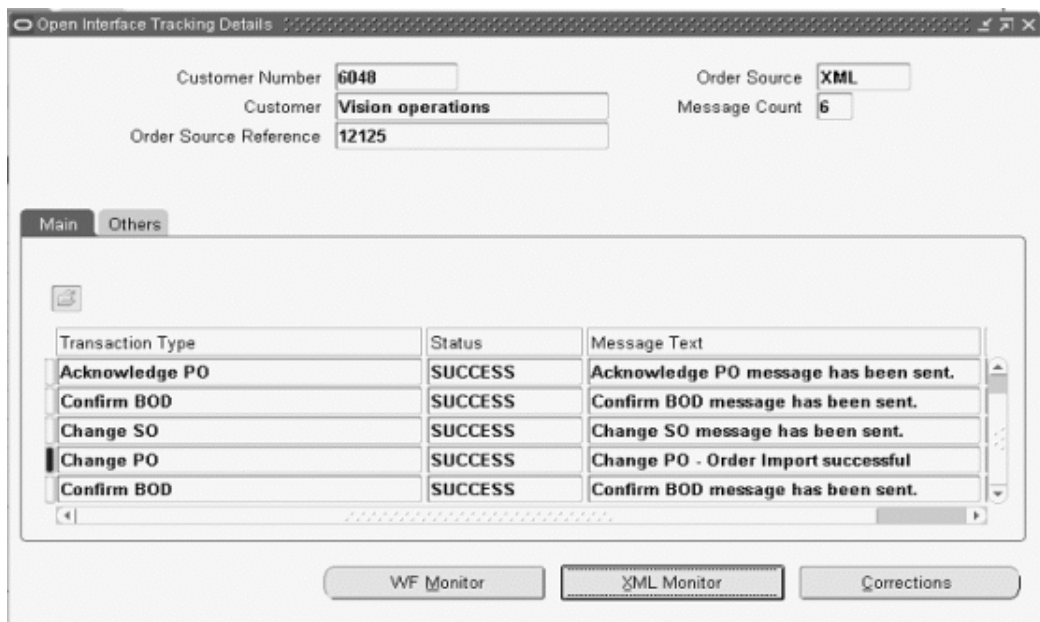
javascript:_subInitNav('DefaultFormName','http://ap6155rt.us.oracle.com:8001/OA_HTML/OALogout.jsp?closeWindow=true&menu=1')

Local intranet

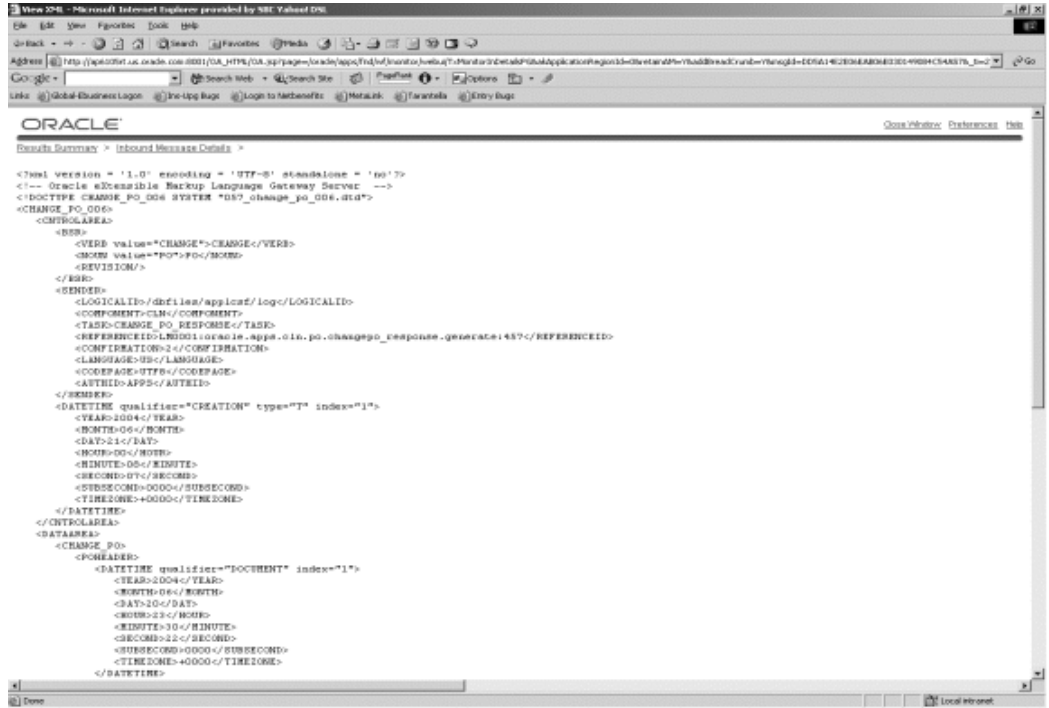
Buyer sends a 3A8 Change PO response acknowledging the changes.



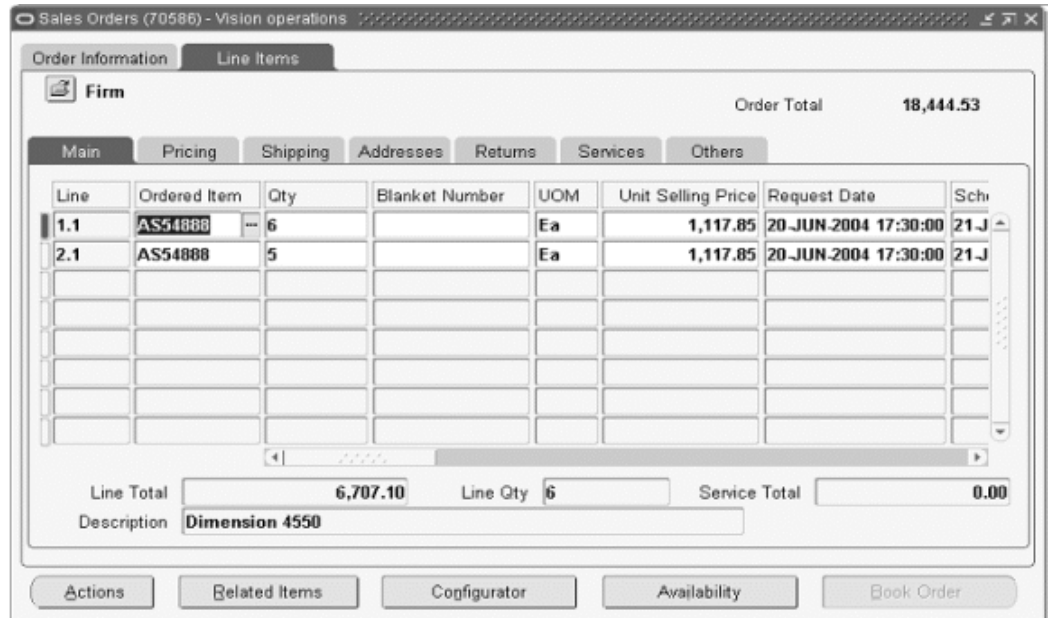
Seller receives 3A8 Change PO response.



Results Summary Inbound Message Details.



3A8 Change PO response is processed successfully and the hold on the sales order line removed.



Cancel_PO

Setup: Customer Setup

Using the Oracle Order Management Super User responsibility, Customer Standard window (Customers-> Standard), query a customer for whom to create XML orders. On one of the customer's addresses, create a primary **Sold To** site usage and specify a location (for example Q7Q-1A1). Save your work. When provided on the inbound XML document, this information will be used to denote a particular customer site and also to determine the destination of the acknowledgment.

The screenshot shows the 'Customer Standard' window for 'Computer Service and Rentals, 1006'. The address details are as follows:

Country	United States	Site Number	1034
Address	301 Summit Hill Drive		
Alternate Name		City	Chattanooga
Postal Code	37401	State	TN
EDL Location	Q7Q-1A1	Province	
Addressee		County	Hamilton

Below the address details is a table of site usages:

Usage	Location	Bill To Location	Primary	Active
Bill To	Chattanooga (OPS)		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ship To	Chattanooga (OPS)	Chattanooga (OPS)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sold To			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>

Setup Steps

Step	Description
1	Define Transactions
2	Define Trading Partners
3	Define Code Conversions

Step	Description
4	Set Up Oracle Workflow Business Event System

Define Transactions

Use Oracle XML Gateway to define a cross reference between the Oracle transaction name and the external transaction name. The external transaction name will be based on what is meaningful per the XML standard used by the recipient. The external transaction name will appear on the message envelope to support message transport.

Define Trading Partners

e-Business can be conducted directly with a business partner commonly known as a trading partner or via a hub such as Oracle Exchange where many buyers and sellers converge to transact in electronic commerce.

With Oracle XML Gateway, you can define the hub or the actual business partner as a trading partner. If you define the hub as the trading partner, you can identify all the buyers and sellers who are conducting business on the hub as trading partners to the hub.

Included in the trading partner/hub definition are the following:

- Trading Partner/Hub name
- Message confirmation requested
- Message map to use for message creation or consumption
- E-mail address of trading partner contact to notify for data errors
- Trading Partner specific code conversion values
- Transport protocol: SMTP, HTTP, or HTTPS with credential and username/password as necessary
- Enable messages for the trading partner by identifying the internal and external transaction type and transaction subtype codes, and the XML standard associated with the message.

To define a trading partner:

1. Navigate to the Define Trading Partner Setup window from the XML Gateway Responsibility. Setup > Define Trading Partners.

2. Enter the Trading Partner Type, Customer.
3. Select the Trading Partner and Trading Partner Site.
4. Select the transaction type as 'ONT' for this functionality.
5. Select 'ONT_3A9R_OAG72_IN' as the map.
6. Save your work.

Define Code Conversions

The Oracle XML Gateway code conversion function provides a method to cross-reference the codes defined in Oracle Applications to codes used by trading partners, the XML standard, or other standard codes in the transactions.

The Code Conversion Standard code determines which transaction Standard these values are applied to. A Standard Code of UNIVERSAL applies these values across all Standards. Here, run the Code Conversion form from the Navigator Setup menu.

Set Up Oracle Workflow Business Event System

Oracle XML Gateway leverages the Oracle Workflow Business Event System to publish and subscribe to application business events of interest to automatically trigger message creation or consumption.

Seeded business events and event subscriptions to send outbound or consume inbound messages are delivered for all Oracle pre-built messages. The seeded event subscriptions may be configured during implementation to perform activities to address specific business requirements.

Define System Profile Values

The following table lists the System Profile options, that need to be set.

Profile Options

Profile System Option	Description	Required	Default Value
OM: Run Order Import for XML	Determines whether Order Import will run Asynchronously or Synchronously.	No	Synchronous

The directories listed in the above profiles should also be mentioned in the utl_file_dir parameter in init.ora

Please refer to the *Oracle XML Gateway User's Guide* for the setup information.

Setup the Subscribing System

Query the Inbound event and change the Subscribing system accordingly.

Define the Subscribing System for the event subscription for event 'oracle.apps.ont.oi.po_inbound.create.'

Subscriber: System - This will usually be the local system, the system on which the subscription code is to run.

(This data will be available as the initial setup, come as an XML file and will be loaded to the customer system. User of the system will be able to change it to work for their system settings).

Message Set Up

To implement a message with a trading partner, use XML Gateway message set up to define the trading partner or hub, code conversion values, and internal to external transaction name cross references. In addition, you may identify the XML Gateway system administrator to notify for system or process errors.

Cancelled Reason Behavior

Order Import populates the OE_ORDER_LINES_HISTORY column with the reason code for the cancellation. When change reasons are specified in the incoming Cancel_PO document at both the header and line level, the line level reason populates the above mentioned table (OE_ORDER_HEADERS_HISTORY does exist but is not populated). The only exception to this behavior occurs when the incoming document does not contain any POLINE tags (i.e. no lines specified but entire order being cancelled). In this case, the header level cancel reason is entered in the history table for each line being cancelled.

Note: This behavior is consistent with the Sales Orders window.

If the incoming document does not contain this tag, then the message map defaults the cancelled code to 'Not provided,' that corresponds to the meaning: 'No reason provided.'

Pre-Defined Workflow Event

A pre-defined Workflow event is seeded to run Order Import immediately if the Profile Option OM: Run Order Import for XML is set to 'Synchronous.' This Workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the Workflow Process it will run Order Import for the XML Message, which is currently being consumed.

Note: Cancel_PO utilizes the same workflows as Process_PO.

Implementation Considerations

Loading of Message Map and DTD

Oracle message maps are delivered and installed as a part of Oracle Order Management file structure and schema. They are automatically loaded into the XML Gateway repository using the LoadMap program. The reports are available to verify the success of the loading process.

Enabling and Consuming the Business Event

The necessary business events and subscriptions are also delivered and installed as part of the schema. They are loaded by the driver.

Open Interface Tracking

Setup: Customer Setup

To setup for Open Interface Tracking:

1. Set up the appropriate Electronic Messages.
2. Set the profile option OM: Electronic Message Integration Event Sources. This profile indicates the level at which electronic messages are tracked and has the values XML, EDI & XML and All (if the value is NULL, this is treated as XML). If the profile option is set to XML, this feature will track only orders with an order source of XML. Similarly, if the profile option is set to EDI & XML, only orders with order source of EDI or XML will be tracked.

Note: Order Management enables users to extract EDI 855 and 865 Acknowledgment data for orders with order source Online. To enable these orders for tracking, the value of the profile option must be set to All; setting the profile to EDI & XML will not track these orders.

3. Set the user access to the form. To allow the form to link to the OA Framework pages for Oracle Workflow and XML Gateway Transaction monitor, ensure that the responsibility with the Open Interface Tracking form has access to the form functions WF_G_DIAGRAM and TXMONITORRESULTSPG – these functions are part of the responsibility Workflow Administrator Web Applications, but you may need to add them to a menu within the responsibility containing the form.

4. Schedule the WF_DEFERRED Agent Listener.
5. Schedule the Workflow Background Process for Item Type OM: Open Interface Tracking.

Integration Event – oracle.apps.ont.oi.xml_int.status

Oracle Order Management has a Business Event System-based mechanism to provide subscribers with a way to perform processing at desired integration points.

Currently, there are three ways these events are triggered:

1. During processing of Electronic Messages (i.e. XML, EDI transactions) and Order Import. In this scenario, the event is raised automatically via the seeded workflows.
2. Because of Processing Constraint-based generation. In this scenario, you can determine the point where the event is raised by defining a Processing Constraint.
3. By inserting the workflow sub-process 'Raise Show Sales Order Event Sub Process' from the item type 'OM Show Sales Order' (OESO) in an Order Header (OEOH) workflow process. This method can be used, for example, to raise an event when an order is booked.

To leverage this functionality, you can define custom synchronous or asynchronous subscriptions to this event and perform any desired processing via the event subscription. For further information about defining event subscriptions, please refer to the Oracle Workflow documentation.

These events are controlled by the profile option, 'OM: Electronic Message Integration Event Sources'. The possible values are 'XML' (default value), 'XML & EDI' and 'All.' Users can use this profile option value to determine the Order Sources for which this event is raised.

1. Integration Event Parameter Details

The following tables summarize the event parameters that are supported by the integration event. Note that not every parameter will be populated in all cases, as described in subsequent sections.

The event **oracle.apps.ont.oi.xml_int.status** will contain the following parameters:

Parameters and Descriptions

Parameter Name	Parameter Description	Comments
XMLG_INTERNAL_CONTR OL_NUMBER	Unique document number generated by XML Gateway for inbound messages	Overloaded to provide a unique id for EDI inbounds and Order Import as well
XMLG_MESSAGE_ID	Unique message id generated by XML gateway for outbound messages	
XMLG_INTERNAL_TXN_TY PE	Typically 'ONT' except for Confirm BOD which is owned by 'ECX'	
XMLG_INTERNAL_TXN_SU BTYPE	Identifies XML transaction type	Overloaded to also contain EDI and Order Import transaction type, namely '850', '860', '855', '865', 'GENERIC'
DOCUMENT_DIRECTION	'IN' or 'OUT'	
XMLG_DOCUMENT_ID	Unique ID provided by OM for outbound messages	Overloaded to provide a unique id for outbound EDI transactions
TRADING_PARTNER_TYPE	'C'	
TRADING_PARTNER_ID	TP Party ID	
TRADING_PARTNER_SITE	TP Party Site ID	
DOCUMENT_NO	Sales Order Number	
ORG_ID	Organization ID	
PARTNER_DOCUMENT_NO	Purchase Order Number	
DOCUMENT_REVISION_N O	Change Sequence	
ONT_DOC_STATUS	'ACTIVE', 'SUCCESS', 'ERROR'	

Parameter Name	Parameter Description	Comments
MESSAGE_TEXT	FND MESSAGE detailing what has occurred.	
WF_ITEM_TYPE	Workflow Info for XML/EDI processing	
WF_ITEM_KEY	Workflow Info for XML/EDI processing	
ORDER_SOURCE_ID	Order Source	
SOLD_TO_ORG_ID	Customer ID	
ORDER_TYPE_ID	Order Type ID	
CONC_REQUEST_ID	Processing Concurrent Request ID	
PROCESSING_STAGE	Code to identify why this event was raised. Sample values include 'INBOUND_IFACE', 'OUTBOUND_SENT'.	
SUBSCRIBER_LIST	Comma delimited list of product short names of the intended subscribers eg 'ONT,CLN'	
DOCUMENT_STATUS	'SUCCESS', 'ERROR'	
RESPONSE_FLAG	'Y' for 3A8 Change PO Response documents	
HEADER_ID	Order Header identifier	
LINE_IDS	Only applicable for events raised via Processing Constraints	

1. Integration Event Parameter Population

Electronic Messaging/Order Import

The following table contains a Y if a particular parameter is populated for XML/EDI transactions as well as Order Import.

Inbound/Outbound Messages Parameter Names

Parameter Name	INBOUND MESSAGES	OUTBOUND MESSAGES
	Parameter Populated (Electronic Message Types)	Parameter Populated (Electronic Message Types)
XMLG_INTERNAL_CONTR OL_NUMBER	Y (XML, EDI, Order Import)	
XMLG_MESSAGE_ID		Y (XML, if document sent successfully)
XMLG_INTERNAL_TXN_TY PE	Y (XML, EDI, Order Import)	Y (XML, EDI)
XMLG_INTERNAL_TXN_SU BTYPE	Y (XML, EDI, Order Import)	Y (XML, EDI)
DOCUMENT_DIRECTION		Y (XML, EDI)
XMLG_DOCUMENT_ID		Y (XML, EDI)
TRADING_PARTNER_TYPE		Y (XML, except CBOD)
TRADING_PARTNER_ID		Y (XML, except CBOD)
TRADING_PARTNER_SITE		Y (XML, except CBOD)
DOCUMENT_NO	Y (XML, EDI, Order Import) (if import succeeds) (information is also derived for XML updates via 3A8/3A9 and processing stage IMPORT_FAILURE)	Y (XML if import succeeds or sales order already exists, EDI)
ORG_ID	Y (XML, EDI, Order Import)	Y (XML, EDI)
PARTNER_DOCUMENT_NO	Y (XML, EDI, Order Import)	Y (XML, EDI)

Parameter Name	INBOUND MESSAGES	OUTBOUND MESSAGES
	Parameter Populated (Electronic Message Types)	Parameter Populated (Electronic Message Types)
DOCUMENT_REVISION_NO	Y (XML, EDI, Order Import)	Y (XML, EDI)
DOCUMENT_STATUS	Y (XML, EDI, Order Import)	Y (XML, EDI)
ONT_DOC_STATUS	Y (XML, EDI, Order Import)	Y (XML, EDI)
MESSAGE_TEXT	Y (XML, EDI, Order Import)	Y (XML, EDI)
WF_ITEM_TYPE	Y (XML except when PROCESSING_STAGE is IMPORT_SUCCESS or IMPORT_FAILURE)	Y (XML, EDI *)
WF_ITEM_KEY	Y (XML except when PROCESSING_STAGE is IMPORT_SUCCESS or IMPORT_FAILURE)	Y (XML, EDI *)
ORDER_SOURCE_ID	Y (XML, EDI, Order Import)	Y (XML, EDI)
SOLD_TO_ORG_ID	Y (XML, EDI, Order Import)	Y (XML, EDI)
ORDER_TYPE_ID	Y (XML, EDI, Order Import) (if import succeeds) (information is also derived for XML updates via 3A8/3A9 and processing stage IMPORT_FAILURE)	Y (XML if import succeeds or sales order already exists, EDI)
HEADER_ID	Y (XML, EDI, Order Import) (if import succeeds) (information is also derived for XML updates via 3A8/3A9 and processing stage IMPORT_FAILURE)	Y (XML, EDI)
CONC_REQUEST_ID	Y (XML, EDI, Order Import)	Y (XML for 3A6 Show SO generated via concurrent program)

Parameter Name	INBOUND MESSAGES	OUTBOUND MESSAGES
	Parameter Populated (Electronic Message Types)	Parameter Populated (Electronic Message Types)
PROCESSING_STAGE	Y (XML, EDI, Order Import)	Y (XML, EDI)
SUBSCRIBER_LIST	Y (XML, EDI, Order Import)	Y (XML, EDI)
RESPONSE_FLAG	Y (XML only for Change PO response documents)	

* The EDI support is only active when the 11.5.10/Pack J EDI workflow is being used

The following table maps the value of the event parameter PROCESSING_STAGE to the message text for that stage:

PROCESSING_STAGE to the Message Text For That Stage

XML	EDI	Order Import	Processing Stage Description (CODE)	Sample Message Text
Y (3A8 Change PO Response only)			Data processed by inbound gateway (INBOUND_GA TEWAY)	"Change PO - Buyer response contained a rejected header and was not processed by Order Management" "Change PO - Buyer response contained one or more rejected lines which were not processed by Order Management"

XML	EDI	Order Import	Processing Stage Description (CODE)	Sample Message Text
Y	Y		Data received in Open Interface Tables (INBOUND_IFACE)	"Process PO – Data successfully entered into Open Interface tables" "850 POI – Data successfully entered into Open Interface tables"
Y			Import mode (PRE_IMPORT)	"Process PO – Order Import will be run in Asynchronous mode" "Process PO – Order Import will be run in Synchronous mode"
Y	Y	Y	Order Import result (IMPORT_SUCCESS, IMPORT_FAILURE)	"Process PO - Order Import successful" "Process PO – Order Import failed"
Y (except CBOD)			Outbound message triggered (OUTBOUND_TRIGGERED)	"OM is ready to send the Acknowledge PO message" "OM is ready to send the Show SO message"

XML	EDI	Order Import	Processing Stage Description (CODE)	Sample Message Text
Y (except CBOD)	Y *		Outbound setup result (OUTBOUND_SETUP)	"Acknowledge PO – Setup validation successful" "Acknowledge PO – Setup validation failed"
Y	Y		Outbound sent (OUTBOUND_SENT)	"Acknowledge PO message has been sent" "855 POAO message has been sent"

Note that for Outbound_Setup, the EDI support is only active when the 11.5.10/Pack J EDI workflow is being used

Constraint-based Event Raise

The following table contains a Y if a particular parameter is populated for Processing Constraint-based Event Raise:

Populated Parameters for Processing Constraint-based Event

Parameter Name	Parameter Populated (Electronic Message Types)
DOCUMENT_NO	Y
ORG_ID	Y
PARTNER_DOCUMENT_NO	Y
DOCUMENT_REVISION_NO	Y
ORDER_SOURCE_ID	Y

Parameter Name	Parameter Populated (Electronic Message Types)
SOLD_TO_ORG_ID	Y
HEADER_ID	Y
SUBSCRIBER_LIST	Y ('DEFAULT')
LINE_IDS	Y
	'ALL' for header-level triggering
	Colon-delimited string of Line Ids for line-level triggering

Note: For colon-delimited string of Line Ids for line-level triggering: If the values populating the line_id string exceeds 2000 characters, the relevant business event will be raised multiple times. The Raise Integration Event operation in the Processing Constraints window will not fail because the line_id string can accommodate more characters.

Workflow Subprocess-based Event Raise

The following table contains a Y if a particular parameter is populated for Workflow Subprocess-based Event Raise (using the 'Raise Show Sales Order Event Sub Process'):

Parameters Populated for Workflow Subprocess-based Event

Parameter Name	Parameter Populated (Electronic Message Types)
DOCUMENT_NO	Y
ORG_ID	Y
PARTNER_DOCUMENT_NO	Y
DOCUMENT_REVISION_NO	Y
ORDER_SOURCE_ID	Y

Parameter Name	Parameter Populated (Electronic Message Types)
SOLD_TO_ORG_ID	Y
ORDER_TYPE_ID	Y
HEADER_ID	Y
SUBSCRIBER_LIST	Y ('DEFAULT')
LINE_IDS	Y ('ALL')

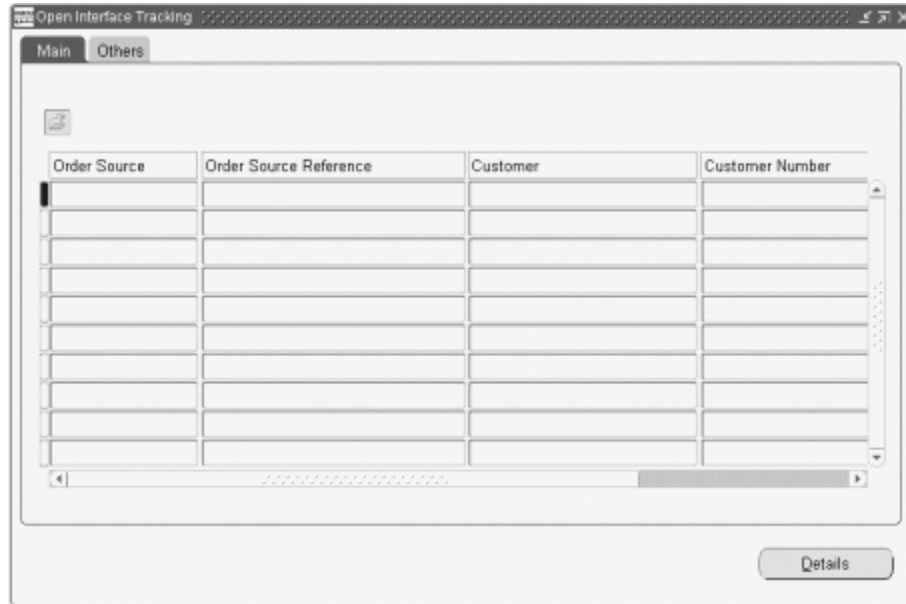
User Procedures

To query electronic messages:

1. Navigate to the Access the Open Interface Tracking Find window.

The screenshot shows a software interface for tracking electronic messages. A 'Find' dialog box is open over a 'Main' window. The 'Find' dialog contains several input fields for search criteria: 'Customer', 'Customer Number', 'Order Source', 'Order Source Reference', 'Includes Transaction', 'Start Date', and 'Last Updated'. There are 'Clear' and 'Find' buttons at the bottom of the dialog. The 'Main' window has a list of results on the left and a 'Details' button at the bottom right.

2. Enter in the desired search criteria and execute the query.
3. To view all the electronic messages for a particular order, select that message and click Details.



To view workflow, XML, and delete records:

1. Navigate to the Access the Open Interface Tracking Find window and query your records.
2. To view the workflow or XML for a particular electronic message, click Details.
3. To view all the electronic messages for a particular order, select that message and click Details.

Note: A concurrent program, Purge Open Interface Data, exists to delete multiple records from the table. This purge procedure can also be used to purge the Interface or Acknowledgment tables.

To purge records:

1. Navigate to the Concurrent Program Manager.
2. Choose the Purge Open Interface Data concurrent program. This concurrent program can also be used to purge the Interface and Acknowledgment tables.

To purge workflows:

Each Open Interface Tracking event is consumed with corresponding information being stored in the OE_EM_INFORMATION_ALL table using the Write History Entry process of the Item Type OM: Open Interface Tracking. Once the workflow for an event completes, the Workflow can be safely purged.

Since OM_PF J is a part of SCM_PF J (which includes CLN), the dependency below will not be an issue.

NOTE: If the customer has Order Management Pack J, but is not on Oracle Applications 11.5.10, then Order Management's seeded subscription to the event **oracle.apps.ont.oi.xml_int.status (required by this feature)** may not process correctly due an issue with another seeded subscription to the same event. This issue can also affect any other product or custom subscriptions to the above mentioned event. To avoid this problem, customers can either:

- Apply 11i.CLN.B Patch #2734844

OR

- Manually disable the seeded event subscription to the event `oracle.apps.ont.oi.xml_int.status` owned by Supply Chain Trading Connector (CLN), if the customer is not using the CLN product. The issue with this event subscription has been resolved in CLN patchset B.

Oracle Order Management XML Transactions

This chapter covers the following topics:

- What is RosettaNet?
- Process_PO
- Acknowledge_PO
- Show_SalesOrder
- Change_salesorder
- Inbound Change PO Request
- Cancel Purchase Order
- CONFIRM BOD
- Seeded Workflow for Inbound XML Messages
- Message Details
- Extension Tags
- Open Interface Tracking
- Event/Event Subscription
- Seeded workflow OEEM/Open Interface Tracking
- Event Parameters & Population

What is RosettaNet?

RosettaNet is a consortium of more than 400 of the world's leading Electronic Components (EC), Information Technology (IT), Semiconductor Manufacturing (SM), and Solution Provider (SP) companies. It is a self-funded, non-profit organization dedicated to creating, implementing and promoting open e-business standards. These

standards form a common e-business language, aligning processes between trading partners on a global basis.

RosettaNet standards offer a robust non-proprietary solution, encompassing data dictionaries, implementation framework, and business message schemas and process specifications, for e-business standardization.

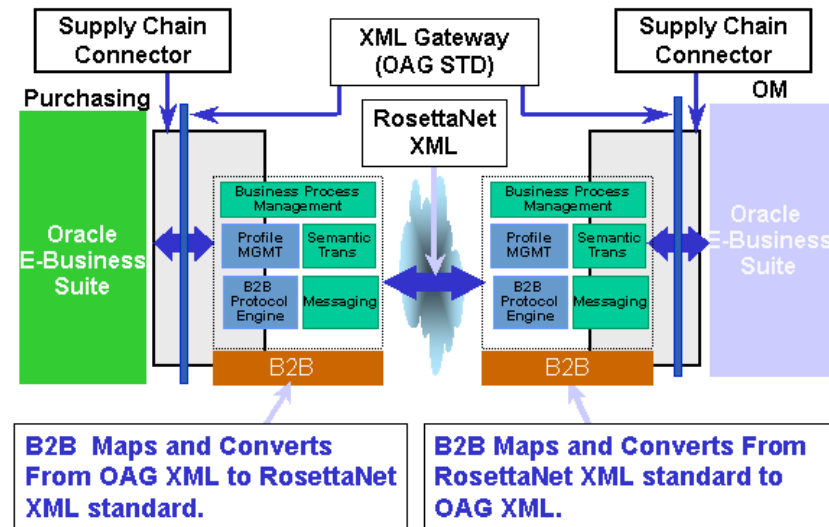
RosettaNet Partner Interface Processes (PIPs) are specialized system-to-system XML-based dialogs that define business processes between trading partners. The message dialog PIPs apply to the following core processes: Administration; Partner, Product and Service Review; Product Introduction; Order Management; Inventory Management; Marketing Information Management; Service and Support; and Manufacturing.

Oracle Order Management supports OAG transactions which correspond to Rosettanet transactions. The following table shows the relation between Rosettanet transactions and OAG transactions:

Rosettanet	OAG	Description
PIP 3A4 -Purchase Order Request	Process_PO	New purchase order from the buyer to the seller
PIP 3A4- Purchase Order confirmation	Acknowledge_PO	Acknowledgement for the new purchase order from the seller to the buyer
PIP 3A9 - Request Purchase Order Cancellation	Cancel_PO	Request for canceling a purchase order from the buyer to the seller
PIP 3A6 - Distribute Order Status	Show_Salesorder	Sales order status sent to the buyer from the seller.

The following diagram describes how Oracle Order Management supports Rosettanet transactions in conjunction with B2B server.

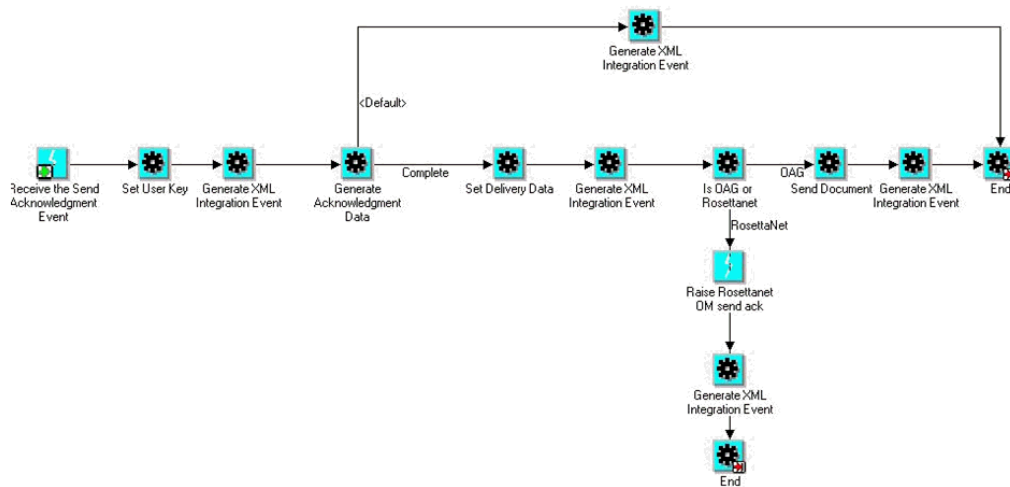
Partner Collaboration - RosettaNet



Changes in Order Management for RosettaNet Native Payloads

Supply Chain Trading Connector (CLN) has included 3 transaction sub-types for Process PO Acknowledgement, Change PO Acknowledgement, and Cancel PO Acknowledgement.

The change enables the workflow to trigger a new event, in case user chooses ROSETTANET as the protocol in the payload. Existing user of OAG doesn't have any impact. The pre-condition is that if a customer uses either OAG or ROSETTANET for all three messages.



The PL/SQL code for Is OAG or RosettaNet is:

```

select standard_code
  into l_standard_code
  from ecx_tp_details_v
 where tp_header_id = (select tp_header_id from ecx_tp_headers
                       where party_id = l_party_id and
                             party_site_id = l_party_site_id
                       and party_type = 'C') and
       transaction_type = 'ONT' and transaction_subtype = 'POA';

if (l_standard_code = 'OAG')
then
  -- Reached Here. Successful execution.
  x_resultout := 'OAG';
else
  x_resultout := 'RN';
end if;

```

Supply Chain Trading Connector (CLN) adds a deferred subscription to the event, so that Order Management will not have any dependency on CLN code. In the deferred subscription, CLN writes a workflow, which generates the appropriate acknowledgement for 3A4, 3A8 and 3A9. After "Raise Rosettanet OM Ack" activity, Generate XML Integration event is raised so that Order Managements XML Message monitoring system can display appropriate status.

Process_PO

Process_PO is used between seller and buyer to exchange information including pricing, availability and status of the order. Process_PO consists of the Inbound PO request and the outbound Acknowledge PO message, which reports the status of the order to the buyer. Typical users of this collaboration are high order volume industries, such as high tech and electronics.

Major Features

Oracle Order Management supports the Process_PO transaction for Open Applications Group (OAG) XML standard. The Process_PO transaction is between the buyer and seller. The buyer can send a new purchase order using this transaction. The following are the major features of this transaction within Oracle Order Management:

Consume the Process_PO Document

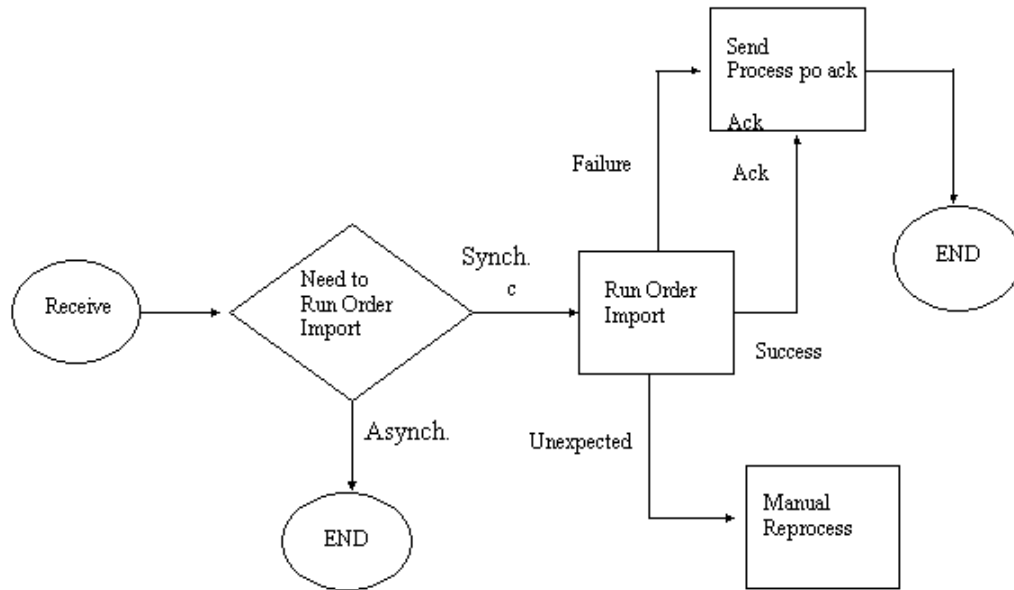
Oracle Order Management consumes the Process_PO (OAG) document from the buyer and creates an order in Oracle Order Management. The internal control number (which is unique for each message) of the incoming message is stored for later use to retrieve the sender information of the original message. This information is mapped on the outbound Acknowledgement message. Oracle Order Management consumes the Process_PO (OAG) document from the buyer and creates an order in Oracle Order Management.

Setup the Order Import Process in Synchronous or Asynchronous Mode

You can initiate Order Import for a single order or process multiple orders via concurrent processing. The profile option, OM: Run Order Import for XML, can be set at the user or site level. The possible values for this profile option will be Synchronous or Asynchronous. In Synchronous mode, Order Import automatically runs when the XML data is in the interface tables. In Asynchronous mode, Order Import needs to be run manually. The default is Synchronous.

Error Handling

This diagram outlines the error handling process:



Three types of outcomes can result from running Order Import process:

1. **Successful processing:** The confirmation API is called to send the acknowledgment to the sender with the Accept status. If the user chooses to book the order before sending out an acknowledgment message, the Accept with Changes status may be sent if there are changes that happened at the booking.
2. **Failed processing:** This status is occurs when there are validation errors or data setup errors. The acknowledgment is sent to the sender with the Reject status. The Order Import Concurrent program will complete normally.
3. **Unexpected Errors:** There can be some unexpected errors due to resource constraints that will require you to manually reprocess Order Import upon fixing those issues. The Order Import Concurrent program will not complete normally.

Order S	Orig Sys Docu	Change	Request ID	Operation Code	Sold To Org ID	Sold To Org
20	syp1008.11			INSERT		
20	ARDOCRE771	1		INSERT	1006	
20	syp1008.9		2198646			
20	syp1008.7		2198608			
20	syp1008.5		2198424			
20	ar.16		2193104		1006	
20	syp1008.4		2198343			
20	syp1007.5		2198222		1006	
20	syp1008.4		2198331			
20	syp1008.3		2198330			

Buttons: Lines, Discounts, Validate, Import, Actions, Sales Credits, Add Customers, Errors, Pricing Attributes

CONFIRM BOD

Upon successful receipt of an inbound XML document such as a Process_PO or Cancel_PO, Oracle Order Management will send the outbound OAG Confirm BOD message.

The major features of Confirm BOD are:

- This message contains information about the inbound XML message received, as well as the status and description of the inbound message received.
- Unlike the Acknowledge PO message, the Confirm BOD message is sent before the Order Import process is run. This is only a confirmation message.

Generate Confirm BOD (CBOD)

Upon receipt of the inbound Process_PO, Oracle Order Management can send out the Confirm BOD message to the buyer, indicating that the data has reached the seller. The Confirm BOD message is sent depending on the value of the <CONFIRMATION> tag in the inbound document. The Confirm BOD message is not sent if the value of this tag is 0 or 1. It is sent if the value of this tag is 2. The Confirm BOD message is generated on the successful insertion of the data in open interface tables. The message is generated by the workflow process that begins after XML gateway has inserted the data successfully in the interface tables of the destination application (Oracle Order Management). The Confirm BOD message is sent only to notify the buyer who may request a confirmation

that the data has been successfully inserted in the interface tables.

Message Details - Message Map ONT_3A4R_OAG72_IN.xgm

The 'ONT_3A4R_OAG72_IN' message map is created using the Oracle XML Gateway Message Designer tool.

Message Map Details

Name	Description
Message Map Name:	ONT_3A4R_OAG72_IN
Direction:	Inbound
(Internal) Transaction Type:	ONT
(Internal) Transaction Subtype:	POI
External Transaction Type:	PO
External Transaction Subtype:	PROCESS
DTD Directory:	xml/oag72
Map Directory:	patch/115/xml/US
Message Maps XGM File Name:	ONT_3A4R_OAG72_IN.xgm
Standard:	OAG
Release:	7.2
Format:	DTD
DTD Name:	PROCESS_PO_007

Columns Enabled for Code Conversion

Defaulted Column	Default Value, and Condition (If Any)
<UOM> (in schedule level <QUANTITY qualifier="ORDERED">)	Code Category – UOM, mapped to OE_LINES_INTERFACE.ORDER_QUANTIT Y_UOM

Key: H => OE_HEADER_INTERFACE, L => OE_LINES_INTERFACE

Defaulted Columns

Defaulted Columns	Default Value, and Condition (If Any)
H.ORDER_SOURCE_ID	20
H.ORDER_TYPE	Standard
H.XML_TRANSACTION_TYPE_CODE	POI
H.HEADER DEFAULT	1-4
L.LINE DEFAULT	1-4
L.ORDER_SOURCE_ID	20
L.ORIG_SYS_DOCUMENT_REF	H.ORIG_SYS_DOCUMENT_REF
L.XML_TRANSACTION_TYPE_CODE	POI

Key: H => OE_HEADER_INTERFACE, L => OE_LINES_INTERFACE

Derived Columns

Derived Columns	Source of Data, and Condition (If Any)
H.CREATED_BY	FND_GLOBAL.USER_ID
H.CREATION_DATE	SYSDATE

Derived Columns	Source of Data, and Condition (If Any)
H.INVOICE_TO_ORG_ID	PARTNRIDX of BillTo PARTNER
H.LAST_UPDATED_BY	FND_GLOBAL.LOGIN_ID
H.LAST_UPDATE_DATE	SYSDATE
H.LAST_UPDATE_LOGIN	FND_GLOBAL.USER_ID
H.ORG_ID	PARTNRIDX of SoldTo PARTNER, if this value is not null
H.SOLD_TO_ORG_ID	PARTNRIDX of SoldTo PARTNER
H.XML_MESSAGE_ID	XML Gateway Internal Control Number
L.CREATED_BY	FND_GLOBAL.USER_ID
L.CREATION_DATE	SYSDATE
L.LAST_UPDATED_BY	FND_GLOBAL.LOGIN_ID
L.LAST_UPDATE_DATE	SYSDATE
L.LAST_UPDATE_LOGIN	FND_GLOBAL.LOGIN_ID
L.ORDERED_QUANTITY, L.ORDER_QUANTITY_UOM	OAG Derivation from schedule level <QUANTITY qualifier="ORDERED"> tag
L.REQUEST_DATE	OAG Derivation from schedule level <DATETIME qualifier="NEEDELV"> tag
L.SHIP_TO_ORG_ID	PARTNRIDX of ShipTo PARTNER

Workflow Event Setup

Detail	Value
Event Name	oracle.apps.ont.oi.po_inbound.create

Detail	Value
Event Description	Order Management Generic Inbound Event
Subscription	R_OEOI_ORDER_IMPORT
Subscription Description	Oracle Order Management subscription to the event oracle.apps.ont.oi.po_inbound.create raised by the post process of the Process_PO XML mapping

Key: H => OE_HEADER_INTERFACE, L => OE_LINES_INTERFACE

Process_PO Message Map

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
CNTROLAREA	The fields included in this area provide information about the XML document, for example, BSR, SENDER, and DATETIME described in this table.	XMLG	Y
<BSR>	Shows the Business Service Request name per OAGI:	XMLG	Y
<VERB>	Value is PROCESS.		Y
<NOUN>	Value is PO.		Y
<REVISION>	Value is 007.		Y

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<SENDER>	Provides information on the system that sends the document:	XMLG	Y
<LOGICALID>			Y
<COMPONENT>	Sender system identifier: value is 1.		Y
<TASK>	Sender application name: value is PURCHASING		Y
<REFERENCEID>	Event or Action: value is POISSUE		Y
<CONFIRMATION>	Unique reference ID for this doc.		Y
<LANGUAGE>	Confirmation when doc is received:		Y
<CODEPAGE>	value of 0 means that none is required.		Y
<AUTHID>	ISO language in which the text fields are transmitted.		Y
	Character set used in this XML doc: value is US7ASCII.		
	System ID of the sender: value is APPS.		
<DATETIME>	Creation date and time of the XML document.	XMLG	Y
DATAAREA	The fields included in this area provide information about the data included in the XML document.		Y
POORDERHDR	This data type provides header level purchase order information. One Purchase Order Header data type is required per document.		Y
<POID>	This identifies the unique ID for the purchase order. For Standard Purchase Orders, the Purchase Order number from Oracle Purchasing is entered here. For Blanket Releases, value should be Blanket PO# - Release. It's the customer Purchase Order Number for Oracle Order Management.	OE_HEADERS_INTERFACE.Orig_System_Document_Ref OE_HEADERS_INTERFACE.CUSTOMER_PO_NUMBER	Y

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<POTYPE>	<p>This identifies various types of Purchase Orders. Standard or Blanket is used here. Oracle Order Management does not have any matching data for the Purchase Order type. The Order Type on Process PO is no longer automatically set to Standard.</p> <p>Oracle Order Management does not need map.</p>		Y
<ACKREQUEST>	<p>Specifies to the supplier if an acknowledgment is expected.</p> <p>Oracle does not support consuming Ack PO so this field is used only to indicate if acceptance for the Purchase Order is required. Values are:</p> <p>0 - Not Required</p> <p>2 – Required</p> <p>Oracle Order Management does not need map.</p>		N
<CONTRACTS>	<p>Seller's contract document number to be used only if this is a release from the blanket order (as it is only available for blanket orders and can only be loaded through PDOI)</p> <p>If this number is available populate it.</p> <p>Order Management does not need map.</p>		N
<DATETIME(DOCUMENT)>	<p>Timestamp for Purchase Order (Standard or Release) creation.</p> <p>Order Management does not need map.</p>		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<DESCRIPTION>	Description for PO Header. Order Order Management does not need map.		N
<OPERAMT(EXTENTE D(T)>	Total amount of the Purchase Order. Following are the fields included in this segment. Order Management does not need map.		N
<VALUE>			N
<NUMOFDEC>	Monetary amount of the PO.		N
<SIGN>	Indicates the number of decimals in the value field.		N
<CURRENCY>			N
<UOMVALUE> ¹	'+'or '-' indicates whether the amount is positive or negative.		N
<UOMNUMDEC>	Three character ISO currency code.		N
<UOM> ¹	Numeric value indicates the value of the factor when the amount is expressed in terms of multiples of Units of Measure (UOM). Represents number of decimals in the UOMVALUE field. Unit of Measure (UOM) indicates the units of the quantitative amount.		
<PORELEASE>	Indicates a new release number; used only when the PO Type is Blanket. Order Management does not need map.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA>	The following fields are provided by Oracle in this user area:	OE_HEADERS_INTERFACE.FOB_POINT_CODE	N
<DATETIME>	Start Active Date for Blanket PO		
<DATETIME>	Not applicable for Oracle Order Management.	OE_HEADERS_INTERFACE.FREIGHT_TERMS_CODE	
<FOB>	End Active Date for Blanket PO:		
<DESCRIPTN>	Not applicable for Oracle Order Management.		
<TERMID>	FOB shipping terms. This is the FOB point code.		
<FTTERM>	FOB Description.		
<DESCRIPTN>	FOB Terms.		
<TERMID>	Freight payment terms.		
<EXCHRATE>	Freight Description.		
<DATETIME>	Freight Terms.		
<DATETIME>	Currency Exchange Rate.		
<CONFIRM>	Date for the Exchange Rate		
DFFPOHEADER	Acceptance Due By date (or Ack by date when PO Ack functionality is supported in Oracle Purchasing)		
<ATTR1 - ATTR16>			
PCARDHDR			
<MEMBERNAME>			
<PCARDNUM>	Not applicable for Oracle Order Management.		
<DATETIME>			
<PCARDBRAND>	Indicates if the supplier confirms the Purchase Order: Y indicates confirmation.		
	N indicates no confirmation. Not applicable for Oracle Order Management.		
	PO header level DFF attributes (16) is not applicable for Oracle Order Management.		
	This segment contains PCARD details Not applicable for Oracle		

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
Order Management			
	Member Name on the Pcard		
	Pcard Number		
	Expiration Date of the Pcard		
	Brand of the Pcard (EDI does the mapping of the PCard brands, it is not possible in XML)		
<ORACLE.CHANGESEQUENCE>	Change Sequence value	OE_HEADERS_INTERFACE.CHANGE_SEQUENCE	
	Booked Flag		
<ORACLE.BOOKEDFLAG>	Supplier's Ref for the PO	OE_HEADERS_INTERFACE.BOOKED_FLAG	
<ORACLE.SUPPLIERDOCREF>		OE_HEADERS_INTERFACE.ORIG_SYSTEM_DOCUMENT_REF	
<USERAREA>Continued..			
<ORACLE.OWNER>	Install Base Owner	OE_HEADERS_INTERFACE.CURRENT_LOCATION	N
<ORACLE.CURRENTLOCATION>	Install Base Current Location	OE_HEADERS_INTERFACE.CURRENT_LOCATION	N
<ORACLE.INSTALLEDATLOCATION>	Install Base Installed At Location	OE_HEADERS_INTERFACE.INSTALLED_AT_LOCATION	N
<ORACLE.ORDERTYPE>	Order Type	OE_HEADERS_INTERFACE.ORDER_TYPE	N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
PARTNER	This data type provides information about the trading partner. Two occurrences of Partner data type are required: Supplier and SoldTo. Oracle ERP provides two additional Partner occurrences: BillTo and Carrier which are optional per DTD.		Y
PARTNER-Supplier	Oracle Order Management does not need to map this segment. This is supplier information.		Y
<NAME1>	Name of the trading partner.		Y
<ONETIME>	Indicates if this partner is established for this transaction only.		Y
<PARTNRID>	Uniquely identifies the partner in ERP.		Y
<PARTNERTYPE>	Identifies the type of partner. Value is Supplier.		Y
<CURRENCY>	Preferred operating currency of the partner.		N
<DESCRIPTION>	Not used.		N
<NAME2 - NAME9>	Not used.		N
<PARTNRIDX>	Unique identifier of the Partner Supplier		N
<TAXEXEMPT>	Not used.		N
<TAXID>	Tax identifier of the partner.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA>	PO Vendor level DFF attributes (16)		N
DFFVENDOR <ATTR1 - ATTR16> <CustomerNum>	Buyer's identifier in the supplier's system (vendor's customer number).		
ADDRESS	This data type provides address information for this partner. The following rows list fields for address data type related to partner supplier site		Y
<ADDRLINE1 - ADDRLINE9>	Lines of site address.		Y
<ADDRTYPE>	Not used.		N
<CITY>	City within the address.		Y
<COUNTRY>	Country within the address.		N
<COUNTY>	Not used.		N
<DESCRIPTN>	Supplier Site Name		N
<FAX1 - FAX9>	Fax numbers of the supplier site		N
<POSTALCODE>	Postal code within the address.		N
<REGION>	Not used.		N
<STATEPROVN>	State of Province within the address.		Y
<TAXJRSCTN>	Not used.		N
<TELEPHONE1 - TELEPHONE9>	Telephone numbers for this address.		N
<URL>	Not used.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA> DFFVENDORSITE <ATTR1 - ATTR16>	PO Vendor Site level DFF attributes (16)		N
CONTACT	This data type provides contact information for this supplier.		Y
<NAME1>	Contact name for the supplier. The contact is entered for each Purchase Order at the header. Use this as the supplier contact. Use the contact name and the telephone numbers of this contact should be used.		Y
<DESCRIPTN>	Not used.		N
<EMAIL>	Email address of the contact		N
<FAX1 - FAX9>	Fax Number of the contact		N
<NAME2 - NAME9>	Not used.		N
<TELEPHONE1 - TELEPHONE9>	Telephone number of the contact.		N
PARTNER-SoldTo			Y
<NAME1>	Name of the buyer company name Sold to name.		Y
<ONETIME>	Indicates if this partner is established for this transaction only.		Y
<PARTNRID>	Uniquely identifies the partner in ERP.This is the Sold_to customer ID that is internal to the buyers application.		Y

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<PARTNERTYPE>	Identifies the type of partner. Value is SoldTo.		Y
<CURRENCY>	Preferred operating currency of partner.		N
<PARTNRIDX>	Unique identifier of the partner. This is the EDI location code for the buyer. Use this code to derive the information about this buyer.	OE_HEADERS_INTERFACE.SOLD_TO_ORG_ID	N
ADDRESS	The following rows list fields for the Address data type related to Partner SoldTo. This is the Sold_to address information.		N
<ADDRLINE1 - ADDRLINE9>	Lines of address.		N
<CITY>	City within the address.		N
<COUNTRY>	Country within the address.		N
<FAX1 - FAX9>	Fax number		N
<POSTALCODE>	Postal code within the address.		N
<STATEPROVN>	State of province within the address.		N
<TELEPHONE1 - TELEPHONE9>	Telephone numbers for this address.		N
CONTACT	The following rows list fields for Contact data type related to Partner SoldTo.		Y
<NAME1>	Name of Buyer e.g. Pat Stock		Y
<DESCRIPTN>	Not used.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<EMAIL>	Email address for the Contact.		N
<FAX1 - FAX9>	Fax Number of the contact		N
<NAME2 - NAME9>	Alternate contact names		N
<TELEPHONE1 - TELEPHONE9>	Telephone number of the contact.		N
<USERAREA>	Not used.		N
PARTNER-BillTo	Bill to location in ERP.		N
<NAME>	Name of the trading partner. Bill to customer name.	OE_HEADERS_INTERFACE.INVOICE_CUSTOMER	Y
<ONETIME>	Indicates if this partner is established for this transaction only.		Y
<PARTNRID>	Uniquely identifies the Bill To Location ID in ERP. This is the EDI location code for the bill to customer site.		Y
<PARTNERTYPE>	Identifies the type of partner. Value is BillTo.		Y
<CURRENCY>	Preferred operating currency of partner.		N
<PARTNRIDX>	Unique identifier of the partner.	OE_HEADERS_INTERFACE.INVOICE_TO_ORG_ID	N
ADDRESS	The following rows list fields for the Address data type related to the Partner Bill_To.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<ADDRLINE1 - ADDRLINE9>	Lines of address.	OE_HEADERS_INTERFACE.INVOICE_ADDRESS1/2/3	N
<ADDRTYPE>	Not used.		N
<CITY>	City within the address.	OE_HEADERS_INTERFACE.INVOICE_CITY	N
<COUNTRY>	Country within the address.	OE_HEADERS_INTERFACE.INVOICE_COUNTRY	N
<COUNTY>	Not Used.	OE_HEADERS_INTERFACE.INVOICE_COUNTY	N
<POSTALCODE>	Postal code within the address.	OE_HEADERS_INTERFACE.INVOICE_POSTAL_CODE	N
<STATEPROVN>	State of province within the address.	OE_HEADERS_INTERFACE.INVOICE_STATE	N
<TELEPHONE1 - TELEPHONE9>	Telephone numbers for this address.		N
PARTNER-CARRIER	Carrier information is passed in this segment. The carrier may become a supplier in Oracle Purchasing in patchset release H or I which may impact how the carrier tags are derived and the address or contact segments may be used then. This segment is not applicable to Oracle Order Management.		N
<NAME1>	Name of trading partner.		Y

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<ONETIME>	Indicates if this partner is established for this transaction only.		Y
<PARTNRID>	Uniquely identifies the partner in ERP.		Y
<PARTNERTYPE>	Identifies the type of partner. Value is Carrier.		Y
ADDRESS	The Address data type is not used for Partner Carrier.		N
CONTACT	The Contact data type is not used for Partner Carrier.		N
PARTNER-End	End Customer		N
<NAME>	Name of Trading Partner. Bill to customer name.	OE_HEADERS_INTERFACE.END_CUSTOMER_NAME	Y
<ONETIME>	Indicates if this partner is established for this transaction only.		Y
<PARTNRID>	Uniquely identifies the Bill To Location Id in ERP. This is the EDI location code for the bill to customer site.	OE_HEADERS_INTERFACE.END_CUSTOMER_NUMBER	Y
<PARTNRTYPE>	Identifies the type of partner. Value is End.		Y
<CURRENCY>	Preferred operating currency of partner.		N
<PARTNRIDX>	Unique identifier of the partner.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
ADDRESS	The following rows list fields for Address data type related to the End Customer.		N
<ADDRLINE1 – ADDRLINE4>	Lines of address.	OE_HEADERS_INTERFACE.END_CUSTOMER_ADDRESS 1/2/3/4	N
<ADDRTYPE>	Not used		N
<CITY>	City within the address.	OE_HEADERS_INTERFACE.END_CUSTOMER_CITY	N
<COUNTRY>	Country within the address.	OE_HEADERS_INTERFACE.END_CUSTOMER_COUNTRY	N
<COUNTY>	County within the address	OE_HEADERS_INTERFACE.END_CUSTOMER_COUNTY	N
<POSTALCODE>	Postal code within the address.	OE_HEADERS_INTERFACE.END_CUSTOMER_POSTAL_CODE	N
<STATEPROVN>	State of Province within the address.	OE_HEADERS_INTERFACE.END_CUSTOMER_STATE	N
<TELEPHONE1 - TELEPHONE9>	Telephone numbers for this address.		N
CONTACT			N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<NAME>	Contact Name	OE_HEADERS_INTERFACE.END_CUSTOMER_CONTACT	N
POTERM	POTERM data type represents payment due dates and discounts.		Y
<DESCRIPTN>	Description of payment terms.		Y
<TERMID> ¹	Identifier for payment terms. This is the payment terms code for example Net30. Can be mapped to payment_terms in Order Order management on the header level.		Y
ATTCHREF	Attachments at the header level. This will only support TO SUPPLIER attachments.		N
<FILENAME>	File name of the file attached if the data type is file.		Y
<URI>	URL, if the data type is URL.		Y
<DATETIME (CREATION)>	Creation date of the document.		N
<DESCRIPTN>	Description of the document.		N
- <NOTES1- NOTES9>	If the data type is Long text or short text then pass the notes to the supplier in these elements.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA>	The following fields are provided by Oracle in the USERAREA: Sequence Number of the attachment. Datatype Name: Short Text, Long Text, Image, or OLE Object.		Y
<SEQNUM>			Y
<DATATYPE>			
POORDERLIN	This data type provides details of a purchase order line. At least one purchase order line data type is required. This data type will occur one or more times.		Y
<POLINENUM>	Line number of the purchase order. Original purchase order line number reference	OE_LINES_INTERF ACE.ORIG_SYS_LI NE_REF	Y
<QUANTITY(ORDERED)>	Indicates the quantity of the item ordered.		Y
<VALUE>	Numeric only, stores the value of quantity.		Y
<NUMOFDEC>	One character numeric only;		Y
<SIGN>	indicates the number of decimals in the value field.		Y
<UOM> ¹	A + or - indicates whether the quantity is positive or negative. The Unit of Measure indicates the units of the quantity.		
<ITEM>	Identifier of the product. Concatenate all the segments to display the item This is the customer item number.	OE_LINES_INTERF ACE.CUSTOMER_I TEM_NAME	Y
<DESCRIPTN>	Description of the item.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<ITEMX>	Vendor's Item Number. This is the item number internal to the supplier side (Oracle Order Management).		N
<USERAREA>			
<USERITEM DESCRIPTN>	User item description	OE_LINES_INTERF ACE.USER_ITEM_ DESCRIPTION	N
<DATETIME(NEEDEDEL V>	Need by date for the item.		N
<HAZRDMATL>	Hazardous material class description.		N
<ITEMRV>	Item revision number. This is the customer item revision.	OE_LINES_INTERF ACE.CUSTOMER_I TEM_REVISION	N
<ITEMRVX>	Not used		N
<NOTES1 - NOTES9>	Note to supplier.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<OPERAMT(UNIT)(T)>	Unit price of the item. Following are the fields included in this segment.		N
<VALUE>			N
<NUMOFDEC>	Monetary unit amount of purchase order line.		N
<SIGN>			N
<CURRENCY>	Indicates the number of decimals in the value field.		N
<UOMVALUE>	A + or - indicates whether the amount is positive or negative.		N
<UOMNUMDEC>			N
<UOM>	Three character ISO currency code.		N
	Numeric value indicates the value of the factor when amount is expressed in terms of multiples of UOM.		
	Represents number of decimals in the UOMVALUE field.		
	Unit of measure indicates the units of the quantitative amount.		
<USERAREA>	Oracle ERP provides the following fields within this userarea: Not applicable for Oracle Order Management.		
<REQUESTOR>			
<CATEGORYID>			
<CONTRACTPONUM>	Requestor of this line. The requester is at the distribution level and not at the line level. Need to find how EDI is deriving it as they have it at the line level.		
<CONTRACTPOLINENUM>			
	Item Category unique identifier.		
	Contract purchase order number for this line		
	Contract purchase order line number for this line. The contract line numbers do not exist in Oracle Purchasing.		

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<VENDORQUOTENUM>	Vendor's quote number for this line		N
<LISTPRICE>	List price of the item		
<MARKETPRICE>	Market price of the item		
<PRICENOTTOEXCEED>	Unit price not to exceed this amount.		
<NEGPRICE>	Negotiable Price Flag Y/N. Only applicable to Blankets and is called Price Override in Oracle Purchasing.		
<TAXABLE>			
<TXNREASONCODE>	If this item is taxable or not Y/N		
<TYPE1099>	Transaction Reason Code is used to group requisition lines for autocreating POs		
<LINEORDERTYPE>			
<HAZRDUNNUM>	Type 1099 Y/N		
<HAZRDUNDESC>			
DFFLINE	Line Order Type e.g. Goods, Services etc.		
<ATTR1-ATTR16>	UN Hazard number		
DFFITEM	UN Hazard Description		
<ATTR1-ATTR16>	Desc Flex Fields at the line level		
KFFITEM	Desc Flex Fields at the item level		
<ATTR1-ATTR16>	Key Flex Fields at the item level		
ATTCHREF-Item Level from Master Org	Item Level attachments from the master org		
ATTCHREF-Item Level from Inventory Org	Item Level attachments from the inventory org		
<USERAREA-CONTD..>	Each occurrence of attachments in the userarea above at the item level is treated as the ATTCHREF segment and has the following elements. This will support attachments TO SUPPLIER only.		N
<ATTCHREF>			

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<FILENAME>	File name of the file attached if the data type is file.		Y
<URI>	URL, if the data type is URL.		Y
<DATETIME (CREATION)>	Creation date of the document.		N
<DESCRIPTN>	Description of the document		N
<NOTES1- NOTES9>	If the data type is Long text or short text then pass the notes to the supplier in these elements.		N
<SEQNUM>	Sequence number of the attachment		Y
<DATATYPE>	Datatype Name - Short Text, Long Text, File, URL, Image or OLE Object.		Y
PARTNER	Not used at line level.		N
ATTCHREF	Attachments at the line level. This will support TO SUPPLIER attachments only.		N
<FILENAME>	File name of the attached file if the data type is file		Y
<URI>	URL, if the data type is URL.		Y
<DATETIME (CREATION)>	Creation date of the document.		N
<DESCRIPTN>	Description of the document.		N
<NOTES1- NOTES9>	If the data type is Long text or short text then pass the notes to the supplier in these elements.		N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA>	The following fields are provided by Oracle in the USERAREA.		Y
<SEQNUM>			Y
<DATATYPE>	Sequence Number of the attachment.		
	Datatype Name - Short Text, Long Text, Image or OLE Object.		
POLINESCHD	Data type for requested ship date information for this PO Line.		N
<<DATETIME(NEEDEDDELV)>>	Needed By delivery date. This is the request date for the line.	OE_LINES_INTERF ACE.REQUEST_DATE	Y
<QUANTITY(ORDERED)>	Indicates quantity of item ordered.	OE_LINES_INTERF ACE.ORDERED_Q	Y
<VALUE>	Numeric only, stores the value of quantity.	UANTITY	Y
<NUMOFDEC>	One character numeric only; indicates the number of decimals in the value field.	OE_LINES_INTERF ACE.ORDER_QUA	Y
<SIGN>		NTITY_UOM	Y
<UOM> ¹	A + or - indicates whether the quantity is positive or negative.		Y
	Unit of Measure indicates the units of the quantity.		
<PSCLINENUM>	Identifies the line number on the delivery schedule of the purchase order.	OE_LINES_INTERF ACE.ORIG_SYS_SH IPMENT_REF	N
	Shipment line reference from the original purchase order line.		

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA>	Promise Date	_LINES_INTERFAC	N
<DATETIME>	Last Acceptance Date	E.SHIP_TO_ORG_ID	N
<DATETIME>	Price Override, this is the new price. Note that is a complete	OE_LINES_INTERF	N
<PRICEOVRD>	Amount field as per OAG specs	ACE.SHIP_TO_AD	N
<TAXABLE>	Taxable Y/N	DRESS1/2/3	N
<TAXCODE>	Tax Code if Taxable is Y	OE_LINES_INTERF	N
<REQUESTOR>	Requestor for this shipment.	ACE.SHIP_TO_CIT	N
PARTNER-ShipTo	Oracle ERP provides the following fields related to ShipTo Partner data type within this user area:	Y	N
<NAME1>	Name of the ShipTo partner.	OE_LINES_INTERF	N
<ONETIME>	Indicates if this partner is established for this transaction only. This is always 0.	ACE.SHIP_TO_CO	N
<PARTNRID>	Uniquely identifies the partner in ERP.	UNTRY	N
<PARTNERTYPE>	Identifies the type of Partner. Value is ShipTo.	OE_LINES_INTERF	N
<CURRENCY>	Preferred operating currency of Partner.	ACE.SHIP_TO_POS	N
<PARTNRIDX>	Unique identifier of the Partner.	TAL_CODE	N
ADDRESS	Lines of address for Partner ShipTo.	OE_LINES_INTERF	N
<ADDRLINE1 - ADDRLINE9>	City within the address.	ACE.SHIP_TO_STA	N
<CITY>	Country within the address.	TE	N
<COUNTRY>	Postal code within the address.		N
<POSTALCODE>	State or province within the address.		N
<STATEPROVN>	Telephone numbers for this address.		N
<TELEPHONE1 - TELEPHONE9>			N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<USERAREA- CONTD..> <ATTCHREF>	Attachments at the Shipment level. This will support TO SUPPLIER attachments only. There could be multiple occurrences of this segment for each shipment schedule.		N
<FILENAME>	File name of the file attached if the data type is file.		Y
<URI>	URL, if the data type is URL.		Y
<DATETIME (CREATION)>	Creation date of the document.		N
<DESCRIPTN>	Description of the document		N
<NOTES1- NOTES9>	If the data type is Long text or short text then pass the notes to the supplier in these elements.		N
<SEQNUM>	Sequence number of the attachment.		Y
<DATATYPE>	Datatype Name - Short Text, Long Text, Image or OLE Object.		Y
<ORACLE.SUPPLIERS HIPMENTREF>	Suppliers reference for the shipment.		N
PARTNER-End	End Customer		N
<NAME>	Name of Trading Partner. Bill to customer name.	OE_HEADERS_INT ERFACE.END_CUS TOMER_NAME	Y
<ONETIME>	Indicates if this partner is established for this transaction only.		Y

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<PARTNRID>	Uniquely identifies the Bill To Location Id in ERP. This is the EDI location code for the bill to customer site.	OE_HEADERS_INT ERFACE.END_CUS TOMER_NUMBER	Y
<PARTNRTYPE>	Identifies the type of Partner. Value is 'End'.		Y
<CURRENCY>	Preferred operating currency of Partner.		N
<PARTNRIDX>	Unique identifier of the Partner.		N
<ADDRESS>	The following rows list fields for Address data type related to the End Customer.		N
<ADDRLINE1 – ADDRLINE4>	Lines of address.	OE_HEADERS_INT ERFACE.END_CUS TOMER_ADDRESS 1/2/3/4	N
<ADDRTYPE>	Not Used.		N
<CITY>	City within the address.	OE_HEADERS_INT ERFACE.END_CUS TOMER_CITY	N
<COUNTRY>	Country within the address.	OE_HEADERS_INT ERFACE.END_CUS TOMER_COUNTR Y	N
<COUNTY>	County within the address	OE_HEADERS_INT ERFACE.END_CUS TOMER_COUNTY	N
<POSTALCODE>	Postal code within the address.	OE_HEADERS_INT ERFACE.END_CUS TOMER_POSTAL_ CODE	N

OAG Element	Description/Comment	Oracle OM Table/Column	OAG Required
<STATEPROVN>	State of Province within the address.	OE_HEADERS_INTERFACE.END_CUSTOMER_STATE	N
<TELEPHONE1 - TELEPHONE9>	Telephone numbers for this address.		N
CONTACT			N
<NAME>	Contact Name	OE_HEADERS_INTERFACE.END_CUSTOMER_CONTACT	N
<ORACLE.OWNER>	Install Base Owner	OE_HEADERS_INTERFACE.IB_OWNER	N
<ORACLE.CURRENTLOCATION>	Install Base Current Location	OE_HEADERS_INTERFACE.CURRENT_LOCATION	N
<ORACLE.INSTALLEDATLOCATION>	Install Base Installed At Location	OE_HEADERS_INTERFACE.INSTALLED_AT_LOCATION	N

Note: Table Key

¹Code conversion is required for this field or data type.

Oracle XML Gateway Message Designer - Properties

The source DTD used is 003_process_po_007.dtd, revision 7.2.1 of the Open Application Group. Here are the other associated external reference DTD files:

- oagis_domains.dtd
- oagis_resources.dtd
- oagis_fields.dtd

- oagis_segments.dtd
- oagis_extensions.dtd
- oagis_entity_extensions.dtd

All the DTDs will be checked in ONT source area under \$ont/xml/oag72

The targets for the Inbound XML Message are the Oracle Order Management Open Interface tables OE_HEADERS_INTERFACE & OE_LINES_INTERFACE.

The PROCESS_PO DTD is a three level hierarchy with order lines split into one or more shipment lines. The Oracle Order Management architecture, however, is a two level hierarchy with Order Header and one or more Lines. The message map collapses the three level XML Message into a two level structure when the data is inserted in the Oracle Order Management Open Interface tables.

Please refer to the table Process_PO Message Map for a detailed analysis of the elements mapped, actions, and derivation rules used by the Message Map.

Both the message maps created using the Message Designer and their associated DTDs are stored in the database. The following Java programs are available to Load/Delete Maps or Load/Delete DTD's into/from the XML Gateway repository. Please refer to the *Oracle XML Gateway User's Guide* for more information.

Load/Delete Maps, Load/Delete DTD's

The following process is used only for customizations.

- java LoadMap <DB username> <DB password> <Hostname>:<Port>:<SID>
<mymap.xgm>
- Example: java oracle.apps.ecx.loader.LoadMap apps apps ap505dbs:1521:dev115
ONT_3A4R_OAG72_IN.xgm
- java DeleteMap <DB username> <DB password> <Hostname>:<Port>:<SID>
<mapname>

Example: java oracle.apps.ecx.loader.DeleteMap apps apps ap505dbs:1521:dev115
ONT_3A4R_OAG72_IN.xgm

The Message Map is an .xgm file which will be stored in the Order Management Source area under \$ont/patch/115/xml/US/ ONT_3A4R_OAG72_IN.xgm

Note: Keep maps and DTDs in sync between the Message Designer and the XML Gateway repository. When required, always reload the map and DTD as a pair.

Extension Tags

Oracle Order Management adds the following tags as extensions on the Process_PO XML documents sent to Oracle Order Management:

\$ont/xml/oag72/oagis_extensions.dtd

- <!ELEMENT FTTERM (DESCRIPTN?, TERMID?)>
<!ELEMENT FOB (DESCRIPTN?, TERMID?)>
<!ELEMENT USERITEMDESCRIPTN %STRDOM;>

The following changes are made by Procurement as extensions to the XML documents sent to Oracle Order Management. As a result, Oracle Order Management must also modify the same OAG DTDs to prevent parse errors at the XML gateway:

\$ont/xml/oag72/oagis_entity_extensions.dtd

- Add the following tags:
<ENTITY % DATETIME.EXCHRATE DATE "DATETIME">
Change <ENTITY % SEG_DATETIME_QUALIFIERS_EXTENSION "OTHER"> to
<ENTITY % SEG_DATETIME_QUALIFIERS_EXTENSION "OTHER |
EXCHRATE DATE">

\$ont/xml/oag72/oagis_extensions.dtd

- Add the following tags:
<!ELEMENT PAYMMETHOD (DESCRIPTN?, TERMID?)>
<!ELEMENT CREDITCRD (CARDID?, NAME?, (%DATETIME.EXPIRATION;?)>
<!ELEMENT STARTACTIVEDATE %STRDOM;>
<!ELEMENT ENDACTIVEDATE %STRDOM;>
<!ELEMENT CATEGORYID %STRDOM;>
<!ELEMENT REVISIONNUM %STRDOM;>
<!ELEMENT ATTACHMENT (TEXT?)>
<!ELEMENT EXCHRATE %STRDOM;>
<!ELEMENT CONFIRM %STRDOM;>
<!ELEMENT PCARDHDR
(MEMBERNAME?, PCARDNUM?, (%DATETIME.EXPIRATION;?), PCARDBRAND
?)>
<!ELEMENT MEMBERNAME %STRDOM;>
<!ELEMENT PCARDNUM %STRDOM;>

```

<!ELEMENT PCARDBRAND %STRDOM;>
<!ELEMENT CUSTOMERNUM %STRDOM;>
<!ELEMENT REQUESTOR %STRDOM;>
<!ELEMENT CONTRACTPONUM %STRDOM;>
<!ELEMENT CONTRACTPOLINENUM %STRDOM;>
<!ELEMENT VENDORQUOTENUM %STRDOM;>
<!ELEMENT LISTPRICE %STRDOM;>
<!ELEMENT MARKETPRICE %STRDOM;>
<!ELEMENT PRICENOTTOEXCEED %STRDOM;>
<!ELEMENT NEGPRICE %STRDOM;>
<!ELEMENT TAXABLE %STRDOM;>
<!ELEMENT TXNREASONCODE %STRDOM;>
<!ELEMENT TYPE1099 %STRDOM;>
<!ELEMENT LINEORDERTYPE %STRDOM;>
<!ELEMENT HAZRDUNNUM %STRDOM;>
<!ELEMENT HAZRDUNDESC %STRDOM;>
<!ELEMENT PRICEOVRD %STRDOM;>
<!ELEMENT DISTPROJECT
(REQUESTOR?,DISTNUM?,PROJECTNUM?,PROJECTTYPE?,TASKNUM?,(%QUA
NTITY.ORDERED;)?,CONVRATE,(%DATETIME.EXCHRATEDATE;)?,DESTTYPE?
,DFFDISTRIBUTN?)>
<!ELEMENT PROJECTNUM %STRDOM;>
<!ELEMENT DISTNUM %STRDOM;>
<!ELEMENT PROJECTTYPE %STRDOM;>
<!ELEMENT TASKNUM %STRDOM;>
<!ELEMENT CONVRATE %STRDOM;>
<!ELEMENT DESTTYPE %STRDOM;>
<!ELEMENT DFFPOHEADER
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>
<!ELEMENT DFFVENDORSITE
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR

```

IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<!ELEMENT DFFVENDOR

(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<!ELEMENT DFFLINE

(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<!ELEMENT DFFITEM

(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<!ELEMENT KFFITEM

(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?,ATTRIBUTE17?,ATTRIBUTE18?,ATTRIBUTE19?,ATTRIBUTE20?)>

<!ELEMENT DFFDISTRIBUTN

(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<!ELEMENT ATTRIBUTE1 %STRDOM;>

<!ELEMENT ATTRIBUTE2 %STRDOM;>

<!ELEMENT ATTRIBUTE3 %STRDOM;>

<!ELEMENT ATTRIBUTE4 %STRDOM;>

<!ELEMENT ATTRIBUTE5 %STRDOM;>

<!ELEMENT ATTRIBUTE6 %STRDOM;>

<!ELEMENT ATTRIBUTE7 %STRDOM;>

<!ELEMENT ATTRIBUTE8 %STRDOM;>

<!ELEMENT ATTRIBUTE9 %STRDOM;>

```

<!ELEMENT ATTRIBUTE10 %STRDOM;>
<!ELEMENT ATTRIBUTE11 %STRDOM;>
<!ELEMENT ATTRIBUTE12 %STRDOM;>
<!ELEMENT ATTRIBUTE13 %STRDOM;>
<!ELEMENT ATTRIBUTE14 %STRDOM;>
<!ELEMENT ATTRIBUTE15 %STRDOM;>
<!ELEMENT ATTRIBUTE16 %STRDOM;>
<!ELEMENT ATTRIBUTE17 %STRDOM;>
<!ELEMENT ATTRIBUTE18 %STRDOM;>
<!ELEMENT ATTRIBUTE19 %STRDOM;>
<!ELEMENT ATTRIBUTE20 %STRDOM;>

<!ELEMENT TANDC %STRDOM;>
<!ELEMENT GLOBALCONTRACT %STRDOM;>
<!ELEMENT GLOBALCONTRACTLIN %STRDOM;>
<!ELEMENT CONSIGNEDINV %STRDOM;>
<!ELEMENT DROPSHIPDETAILS (DROPSHIPMENT?, DROPSHIPCUSTNAME?,
SHIPINSTR?,
PACKINSTR?, SHIPMETHOD?, CUSTOMERPONUM?, CUSTOMERLINENUM?,
CUSTOMERSHIPNUM?,
CUSTOMERDESC?)>
<!ELEMENT DROPSHIPMENT %STRDOM;>
<!ELEMENT DROPSHIPCUSTNAME %STRDOM;>
<!ELEMENT SHIPINSTR %STRDOM;>
<!ELEMENT PACKINSTR %STRDOM;>
<!ELEMENT SHIPMETHOD %STRDOM;>
<!ELEMENT CUSTOMERPONUM %STRDOM;>
<!ELEMENT CUSTOMERLINENUM %STRDOM;>
<!ELEMENT CUSTOMERSHIPNUM %STRDOM;>
<!ELEMENT CUSTOMERDESC %STRDOM;>
<!ELEMENT SHIPPINGCONTROL %STRDOM;>
<!ELEMENT CONTRACTNUM %STRDOM;>
<!ELEMENT CONFIGID %STRDOM;>

```

```

<!ELEMENT PSCLNSTATUS %STRDOM;>
<!ELEMENT JOBTITLE %STRDOM;> -- HR Job title
<!ELEMENT CONTRACTORFIRSTNAME %STRDOM;> -- Contractor first name
<!ELEMENT CONTRACTORLASTNAME %STRDOM;> -- Contractor last name
<!ELEMENT PRICEDIFFERENTIAL
(NUMBER?,ENTITYTYPE?,MULTIPLIER?,PRICETYPE?)>
--repeatable segment
<!ELEMENT NUMBER %STRDOM;> --price differential line num
<!ELEMENT ENTITYTYPE %STRDOM;> -- Entity type
<!ELEMENT MULTIPLIER %STRDOM;> -- multiplier for the standard PO line

```

Acknowledge_PO

Overview

Oracle Order Management supports the Acknowledge_PO transaction for the OAG XML standard.

Oracle Order Management generates the Acknowledge_PO message upon consuming the Process_PO inbound message and sends an Acknowledge_PO message to the buyer. The Acknowledge_PO message is sent in both the Synchronous and Asynchronous processing of the incoming orders.

This outbound message carries the status of the order received via the inbound Purchase Order request to the buyer. The message also contains the reason code for the rejected orders.

Major Features

Major features of this transaction are as follows:

Generate and Send Acknowledge_PO Message Out

Oracle Order Management generates the Acknowledge_PO message upon consuming the Process_PO inbound message (by triggering the Order Import and entering the order in Order Management tables). At this point Order Management must trigger the workflow to send an Acknowledge_PO message to the buyer. This message should be generated within the timeframe agreed upon in the Trading Partner Agreement.

The Acknowledge_PO message is generated in both the Synchronous and Asynchronous processing of the incoming orders.

Acknowledge_PO message is populated with the information about the incoming Process_PO request. The message ID of the incoming Purchase Order is stored and retrieved later during the creation of the Acknowledge_PO.

For the details of this message and mapping to Order Management, please refer to the table Acknowledge_PO OAG Message Map Details.

Oracle Order Management generates the Acknowledge_PO message upon consuming the Process_PO inbound message and sends an Acknowledge_PO message to the buyer.

The Acknowledge_PO message is sent in both the Synchronous and Asynchronous processing of the incoming orders.

Acknowledge_PO message has information about the incoming Process_PO request.

Possible Trigger Points for Acknowledge_PO Message

In the case of failure at the XML gateway processing BEFORE the order reaches the Order Import tables, there is no action required by Oracle Order Management. The XML gateway notification process handles the exception by notifying the System Administrator on the buyer's side.

Possible Trigger Points

The following are the possible trigger points for the Acknowledge_PO message. These points are exclusive to each other, which means that for an order only one Acknowledge_PO message is generated at one of these points in the process.

After the Order import process runs, if there is a validation error and the Order import process cannot create an order in the base order table, the Acknowledge_PO message is sent to the buyer with a **Reject** status. If the Order Import process succeeds, the Acknowledge_PO message is sent to the buyer with an Accepted status at order entry.

Message Map

Message Map

Name	Description
Message Map Name	ONT_3A4A_OAG72_OUT_PO
Direction	Outbound
(Internal) Transaction Type	ONT
(Internal) Transaction Subtype	POA
External Transaction Type	PO
External Transaction Subtype	ACKNOWLEDGE

Name	Description
DTD Directory	xml/oag72
Map Directory	patch/115/xml/US
Message Maps XGM File Name	ONT_3A4A_OAG72_OUT_PO.xgm
Standard	OAG
Release	7.2
Format	DTD
DTD Name	004_acknowledge_po_008.dtd

Workflow Event Setup

Detail	Value
Event Name	oracle.apps.ont.oi.po_ack.create
Event Description	Event for Process_PO Outbound Acknowledgment
Subscription	R_OEOA_SEND_ACKNOWLEDGMENT
Subscription Description	Oracle Order Management subscription to the event oracle.apps.ont.oi.po_ack.create raised for sending acknowledgment for Process_PO.

Key: H => OE_HEADER_ACKS and L => OE_LINE_ACKS

Acknowledge_PO OAG Message Map Details

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<ACKNOWLEDGE_PO_008>				
<CNTROLAREA>				
<BSR>				
<VERB>ACKNOWLEDGE</VERB>	XMLG		Y	
<NOUN>PO</NOUN>	XMLG		Y	
<REVISION>007</REVISION>	XMLG		Y	
</BSR>				
<SENDER>				
<LOGICALID>CPAG</LOGICALID>	XMLG		Y	
<COMPONENT>ORDERMGMT</COMPONENT>	XMLG		Y	
<TASK>ACKPO</TASK>	XMLG		Y	
<REFERENCEID>CPAGPOBERLIN02</REFERENCEID>	XMLG		Y	
<CONFIRMATION>0</CONFIRMATION>	XMLG		Y	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<LANGUAGE>ENG</LANGUAGE>	XMLG		Y	
<CODEPAGE>CP000111</CODEPAGE>	XMLG		Y	
<AUTHID>RSCHULTE</AUTHID>	XMLG		Y	
<SENDER>				
<DATETIME qualifier="CREATION">	XMLG		Y	
<YEAR>1996</YEAR>	XMLG		Y	
<MONTH>06</MONTH>	XMLG		Y	
<DAY>30</DAY>	XMLG		Y	
<HOUR>23</HOUR>	XMLG		Y	
<MINUTE>59</MINUTE>	XMLG		Y	
<SECOND>59</SECOND>	XMLG		Y	
<SUBSECOND>0000</SUBSECOND>	XMLG		Y	
</DATETIME>				
</CONTROLAREA>				
<DATAAREA>				

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<ACKNOWLEDGE_PO>				
<POORDERHDR>				
<DATETIME qualifier="DOCUMENT">	Date time stamp of the Sales order creation. Derived using Convert To OAG Datetime.	H.ORDERED_DATE	Y	
<YEAR>1996</YEAR>			Y	
<MONTH>06</MONTH>			Y	
<DAY>30</DAY>			Y	
<HOUR>23</HOUR>			Y	
<MINUTE>59</MINUTE>			Y	
<SECOND>59</SECOND>			Y	
<TIMEZONE>+0100</TIMEZONE>			Y	
<SUBSECOND>0000</SUBSECOND>			Y	
</DATETIME>				

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<OPERAMT qualifier="EXTENDED" type="T">	Total amount for the Order. This is not stored but calculated using Convert To OAG Amount.	H.ORDER_TOTAL	Y	
<VALUE>670</VALUE>	Total monetary amount of the order.		Y	
<NUMOFDEC>2</NUMOFDEC>	Indicates the number of decimals in the value field.		Y	
<SIGN>+</SIGN>	A + or - indicates whether the amount is positive or negative.		Y	
<CURRENCY>USD</CURRENCY>	Three character ISO currency code.		Y	
<UOMVALUE>1</UOMVALUE>	Numeric value indicates the value of the factor when amount is expressed in terms of multiples of UOM.		Y	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<UOMNUMDEC>0</UOMNUMDEC>	Represents number of decimals in the UOMVALUE field.		Y	
<UOM>EACH</UOM>	Unit of Measure indicates the units of the quantitative amount.		Y	
</OPERAMT>			Y	
<POID>12345678</POID>	Original Customer PO number from the inbound process_PO should be populated here.	H.CUSTOMER _PO_NUMBER	Y	
<POTYPE>224</POTYPE>	'STANDARD' Indicates the type of the PO. At the moment only Standard PO will be the only type supported. The Order Type on Process Purchase Order will no longer automatically be set to Standard.		Y	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<PORELEASE>1</PORELEASE>	PO release number. Is not populated by OM. Purchasing will split the Po release number from the PO number which has a format of PO Number release number *See Notes		N	
<ACKHEADER>				
<SENDER>	* See Notes		Y	
<LOGICALID>CPAG</LOGICALID>	XML Gateway API	XML Gateway API	Y	
<COMPONENT>PURCHASING</COMPONENT>	XML Gateway API	XML Gateway API	Y	
<TASK>POCREATE</TASK>	XML Gateway API	XML Gateway API	Y	
<REFERENCEID>CPAGPOBERLIN02</REFERENCEID>	XML Gateway API	XML Gateway API	Y	
<CONFIRMATION>0</CONFIRMATION>	XML Gateway API	XML Gateway API	Y	
<LANGUAGE>ENG</LANGUAGE>	XML Gateway API	XML Gateway API	Y	
<CODEPAGE>CP000111</CODEPAGE>	XML Gateway API	XML Gateway API	Y	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<AUTHID>CMKURT </AUTHID>	XML Gateway API	XML Gateway API	Y	
</SENDER>				
<ACKCODE>1</ACK CODE>	Status Code * See Notes	H.FIRST_ACK_ CODE	Y	
<NOTES index="1">This is a test message</NOTES>	Reason Code * See Notes		N	
<SALESORDID>3232 5</SALESORDID>	Sales Order Number	H.ORDER_NU MBER	N	
</ACKHEADER>			N	
<PARTNER>			N	
<NAME index="1">CPAG</N AME>	Trading partner name . In this case Buyers name. Derived from OE_XML_PRO CESS_UTIL.GE T_SOLD_TO_E DI_LOC			
<ONETIME>0</ONE TIME>	0			
<PARTNERID>A123< /PARTNERID>	Internal customer number in the sellers system (OM).	H.SOLD_TO_O RG_ID		

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<PARTNERIDX>A123</PARTNERIDX>	Unique identifier for the sold to partner. Value should be EDI Location code for the sold-to partner. Derived from OE_XML_PROCESS_UTIL. GET_SOLD_TO_EDI_LOC.			
<PARTNERTYPE>SoldTo</PARTNERTYPE>	Partner Type. Value should be 'Sold To'			
</PARTNER>				
<PARTNER>			N	
<NAME index="1">CPAG</NAME>	Trading partner name. In this case Suppliers name.	H.SHIP_FROM_ORG		
<ONETIME>0</ONETIME>	0			
<PARTNERID>A123</PARTNERID>	Supplier number This will be the internal location number for the seller.	H.SHIP_FROM_ORG_ID		

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<PARTNERIDX>A123</PARTNERIDX>	Unique identifier for the supplier. EDI location code for the supplier. Setup in Inventory/setup /Org/Location. This code is used by the buyer (PO) to identify the supplier.			
<PARTNERTYPE>Supplier</PARTNERTYPE>	Partner type. The value should be 'Supplier'			
</PARTNER>				
<POTERM>			N	
<TERMID>XXX</TERMID>	Payment term code.	H.PAYMENT_TERM_ID	Y	Y
<DESCRIPTN>XXX</DESCRIPTION>	Description of the payment term		N	
</POTERM>				
</POORDERHDR>				
<POORDERLIN>				

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<QUANTITY qualifier="ORDERED >	This is the original ordered Qty. Required by OAG but will be ignored by Oracle Purchasing. This quantity would need to be stored from the Original PO (Process_PO inbound). Derived Using Convert to OAG Quantity.	L.ORDERED_Q UANTITY	Y	
<VALUE>1</VALUE >			Y	
<NUMOFDEC>0</N UMOFDEC>			Y	
<SIGN>+</SIGN>			Y	
<UOM>EACH</UO M>			Y	
</QUANTITY>			Y	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<OPERAMT qualifier="UNIT" type="T">	Changed Unit selling price. This the new price if any changes are made to the price. If price is not changed this can be left null. Derived using Convert to OAG Operating Amount.	L.UNIT_SELLI NG_PRICE	N	
<VALUE>6500000</V ALUE>			Y	
<NUMOFDEC>2</N UMOFDEC>			Y	
<SIGN>+</SIGN>			Y	
<CURRENCY>USD</ CURRENCY>			Y	
<UOMVALUE>1</U OMVALUE>			Y	
<UOMNUMDEC>0</ UOMNUMDEC>			Y	
<UOM>EACH</UO M>			Y	
</OPERAMT>				
<POLINENUM>1</P OLINENUM>	PO Line number of the original PO.	L.ORIG_SYS_LI NE_REF	Y	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<ITEMX>XXXXXX</ITEMX>	Customer Item number.	L.CUSTOMER_ITEM	N	
<USERAREA>				
<USERITEMDESCRIP PTN>item desc </USERITEMDESCRIP PTN>		L.USER_ITEM_DESCRIPTION		
</USERAREA>				
<ACKLINE>				
<ACKCODE>1</ACKCODE>	Status Code. See Notes.	L.FIRST_ACK_CODE	Y	
<NOTES index="1">This is a test message</NOTES>	Reason Code. See Notes.		N	
</ACKLINE>				
<POLINESCHD>				

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<DATETIME qualifier="NEEDELV">	Needed by delivery date as from the Original PO. Required by OAG and will be ignored by Purchasing. This should be mapped to the request date in OM. Derived using Convert To OAG Datetime.	L.REQUEST_D ATE	Y	
<YEAR>1996</YEAR>			Y	
<MONTH>07</MONTH>			Y	
<DAY>01</DAY>			Y	
<HOUR>23</HOUR>			Y	
<MINUTE>59</MINUTE>			Y	
<SECOND>59</SECOND>			Y	
<SUBSECOND>0000</SUBSECOND>			Y	
</DATETIME>				

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<QUANTITY qualifier="ORDERED ">	Original Order Quantity. This quantity would need to be stored from the Original PO (Process_PO inbound). Derived using Convert To OAG Quantity.	L.ORDERED_Q UANTITY	Y	
<VALUE>1</VALUE >			Y	
<NUMOFDEC>0</N UMOFDEC>			Y	
<SIGN>+</SIGN>			Y	
<UOM>EACH</UO M>			Y	Y
</QUANTITY>				
<PSCLINENUM>1</P SCLINENUM>	Original Schedule shipment line number from the original PO.	L.ORIG_SYS_S HIPMENT_REF	N	
</USERAREA>				
<ACKCODE>X</AC KCODE>		L.FIRST_ACK_ CODE	N	

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
<DATETIME qualifier = " PROMDELV">	Promised delivery date from OM. Derived using Convert To OAG Datetime.	L.SCHEDULE_ ARRIVAL_DA TE	N	
<YEAR>1996</YEAR >			Y	
<MONTH>07</MON TH>			Y	
<DAY>01</DAY>			Y	
<HOUR>23</HOUR>			Y	
<MINUTE>59</MIN UTE>			Y	
<SECOND>59</SECO ND>			Y	
<SUBSECOND>0000< /SUBSECOND>			Y	
</DATETIME>				
<SALESORDID>123</ SALESORDID>	Sales Order number for this order line.	H.ORDER_NU MBER	N	
<SOLINENUM>1</S OLINENUM>	Sales order line number *. See Notes.	L.LINE_NUMB ER	N	
</USERAREA>				
</POLINESCHD>				

OAG Element	Description /Comment	Oracle OM Table /Column	OAG Req Y/N	Code Conversion Needed?
</POORDERLIN>				
</ACKNOWLEDGE_PO>				
</DATAAREA>				
</ACKNOWLEDGE_PO_008>				
** XMLG – XML gateway				

Message Map Notes

Name	Description
POHEADER/PORELEASE	This is the Purchase Order release number that Order Management doesn't store. Currently, the solution is provided by Oracle Application Server 10g where the Purchase Order release number is concatenated to the Purchase Order number and sent in as a Customer PO number. On the Purchasing side this concatenated number is separated again by Oracle Application Server 10g and populated in to the release number columns for Purchasing.
ACKHEADER / ACKCODE	Status code – <ul style="list-style-type: none"> • 0- Accepted • 2- Reject

Name	Description
ACKHEADER / NOTES	Reason code – The value of the Ackcode field determines the values of this text field: Accept or Reject When ACK PO is used as a confirmation for Change PO or Cancel_PO the Reason code is not required.
POLINE / ACKLINE / ACKCODE	Status Code – <ul style="list-style-type: none"> • Accept if all the shipment lines are accepted (even with changes) • Rejected if any of the shipment lines is rejected.
POLINE / POLNSTATUS	Line Status – <ul style="list-style-type: none"> • Open - PO line is open to receiving. • Closed - PO line has completed normally. It is no longer available to receiving. • Cancelled – PO line has completed abnormally or has been deleted. The PO is no longer open to receiving.
USERAREA / SOLINENUM	This element contains sales order line number. Its been added here to allow 3A4 transaction to create multiple Sales orders from one PO.
POLINESCHD/ USERAREA/ ACKLINE /ACKCODE	Status code – <ul style="list-style-type: none"> • 0- Accept • 2- Reject
POLINESCHD / DATETIME(NEEDEDELV)	This date element will contain the request_date.
POLINESCHD / QUANTITY (ORDERED)	This quantity element will contain the ordered_quantity.

Message Map 'ONT_3A4A_OAG72_OUT.xgm'

The message map 'ONT_3A4A_OAG72_OUT' is created using the Oracle XML Gateway

Message Designer tool. Please refer to Table, "Message Map" for the detailed map analysis.

The source DTD used is 004_acknowledge_po_008.dtd, revision 7.2.1 of the Open Application Group. The other associated external reference DTD files are;

- oagis_domains.dtd
- oagis_resources.dtd
- oagis_fields.dtd
- oagis_segments.dtd

All the DTD's will be checked in ONT source area under \$ont/xml/oag72

The target for the Outbound XML Message are the Order Management Acknowledgment tables OE_HEADER_ACKS & OE_LINE_ACKS.

The ACKNOWLEDGE_PO_008 DTD is a three level hierarchy with Order Lines split into one or more Shipment Lines. The Order Management architecture is however a two level one with the order header and one or more lines. The message map will expand the two level structure of the Order Management tables to three level XML Message.

Please refer to Table, "Message Map" for a detail analysis of the elements mapped, actions and derivation rules used by the Message Map.

Both the message map created using the Message Designer and its associated DTD's are stored in the database. The following Java programs are available to Load/Delete Maps or Load/Delete DTD's into/from the XML Gateway repository. Please refer to the *Oracle XML Gateway User's Guide* for more information.

Load/Delete Maps, Load/Delete DTD's

The following process is used only for customizations.

- java LoadMap <DB username> <DB password> <Hostname>:<Port>:<SID>
<mymap.xgm>
- Example: java oracle.apps.ecx.loader.LoadMap apps apps ap505dbs:1521:dev115
ONT_3A4A_OAG72_OUT_PO.xgm
- java DeleteMap <DB username> <DB password> <Hostname>:<Port>:<SID>
<mapname>

Example: java oracle.apps.ecx.loader.DeleteMap apps apps ap505dbs:1521:dev115
ONT_3A4A_OAG72_OUT_PO.xgm

The Message Map is a .xgm file which will be stored in the Order Management Source area under \$ont/patch/115/map/ ONT_3A4A_OAG72_OUT_PO.xgm

Note: Maps and DTD's must be kept in sync between the Message

Designer and the XML Gateway repository. When in doubt, always reload the map and DTD as a pair.

Pre-Defined Workflow Event

If the profile option OM: Run Order Import for XML is set to Synchronous, a pre-defined Workflow event is seeded to run Order Import if the Profile Option OM: Process Transaction is set to ONLINE. This Workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the Workflow Process, Order Import is run for the XML Message which is currently being consumed. After the Run Order Import completes (Successfully or With Error), the next event is raised to continue processing.

Seeded Workflow

Send XML PO Acknowledgment

Show_SalesOrder

The Show_SalesOrder message is used by the seller to send order status information, by line item for an order to a buyer. This outbound message carries the status of the order line.

Generate and send Show_salesorder message out

Oracle Order Management generates the outbound Show SO message at various trigger points described in following sections. The Show SO message contains status information about the order at the line item level. For the details of this message and mappings refer to Table, "Message Map".

Periodical

The Show_salesorder message can be sent periodically depending on the period set in the trading partner agreement between the buyer and the seller. Examples of this would be daily, weekly, or at a particular time of the day.

Order Management provides the concurrent program called by the triggering mechanism, scheduling the process of the concurrent program to generate the show_salesorder message. This API has following parameters:

1. **Trading partner ID (EDI Location code)** – Order Management created the Show_salesorder message for this trading partner.
2. **Open Orders Only** – This is the order status that Order Management creates the show_salesorder message for. The API chooses between open or closed orders. If Open Orders Only is set to Yes, only open orders will be picked up. If Open Orders

Only is set to No, only closed orders will be picked up.

3. **Closed for Days** – Order Management is currently not supporting the use of this parameter. This would be number of days. Order Management would create a show_salesorder message for all closed orders, which are closed within last 'N' number of days.
4. **Sales Order No From** – Starting at sales order number.
5. **Sales Order No To** – End at sales order number.
6. **Sales Order date from**- From sales order creation date.
7. **Sales Order date to** – To sales order date.
8. **Customer PO No From** – Starting at Customer PO number.
9. **Customer PO No To** – End at Customer PO number.

Order Changes

Changes in the attributes of the order that occur during order processing. This could be shipping or scheduling for example. If changes to any of these attributes occur in a booked order, the Show Sales Order message is sent.

Changes in following attributes are supported at this point:

- Unit selling Price,
- Ordered Qty
- Scheduled arrival date
- Shipped Qty
- Scheduled Ship Date

Any time any of these attributes change for a booked order Order Management generates a Show_Salesorder message.

Adding a new line to a booked order also generates a Show_Salesorder message. Consequently, splitting a line also generates a Show_Salesorder message, due to the quantity change and the generation of the new line.

- Status change due to business events – Booking the order, shipping, or scheduling of lines - the Show_SalesOrder is generated when the order is booked. It will also, due to detection of the corresponding attribute changes, be generated for scheduling and shipping of lines.

Business Scenarios and Process Flow

The Show_SalesOrder message is created by the Order Management in response to various trigger points. The period set by the customer is usually one of the key trigger points. Additionally, you can initiate show_salesorder as a result of any or a combination of the following scenarios:

1. The order is booked
2. If the Unit selling price, Schedule delivery date, or ordered quantity changed systematically (for booked orders)
3. The line is scheduled
4. The line is shipped
5. The line is cancelled
6. The line is added to a booked order

The XML Gateway forms the SHOW_SO message.

Show_SalesOrder Message Map

Key: H => OE_HEADER_ACKS, L => OE_LINE_ACKS

Show_SalesOrder Message Map

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG re qd Y/ N	Code Conv ersio n Need ed Y/N
<SHOW_SALESORDER_006>				
<CNTROLAREA>			Y	
<BSR>				
<VERB>SHOW</VERB>	XMLG			
<NOUN>SALESORDER</NOUN>	XMLG			
>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<REVISION>006</REVISION>	XMLG			
</BSR>				
<SENDER>				N
<LOGICALID>XGRB1109</LOGICALID>	XMLG			
<COMPONENT>OM</COMPONENT>	XMLG			
<TASK>SO</TASK>	XMLG			
<REFERENCEID>9534223449</REFERENCEID>	XMLG			
<CONFIRMATION>1</CONFIRMATION>	XMLG			
<LANGUAGE>ENG</LANGUAGE>	XMLG			
<CODEPAGE>CP001001</CODEPAGE>	XMLG			
<AUTHID>SMITHJ</AUTHID>	XMLG			
</SENDER>				
<DATETIME qualifier="CREATION">	XMLG			Y
<YEAR>1998</YEAR>				
<MONTH>12</MONTH>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG re qd Y/ N	Code Conv ersio n Need ed Y/N
<DAY>29</DAY>				
<HOUR>00</HOUR>				
<MINUTE>25</MINUTE>				
<SECOND>45</SECOND>				
<SUBSECOND>0000</SUBSECON D>				
</DATETIME>				
</CNTROLAREA>				
<DATAAREA>				
<SHOW_SALESORDER>				
<SOHEADER>				
<SALESORDID>S0001</SALESOR DID>	Sales order number.	H.ORDER_NUMB ER	Y	
<DATETIME qualifier="DOCUMENT">	Date time stamp of sales order creation.	H.ORDERED_DA TE		
<YEAR>1998</YEAR>	Derived using Convert To OAG Datetime.			
<MONTH>12</MONTH>				
<DAY>29</DAY>				
<HOUR>00</HOUR>				
<MINUTE>25</MINUTE>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<SECOND>45</SECOND>				
<SUBSECOND>0000</SUBSECOND>				
</DATETIME>				
<POID>123</POID>	Customer PO number.	H.ORIG_SYS_DOCUMENT_REF	N	
<SOSTATUS>XXX</SOSTATUS>	Sales order status (see notes).	H.FIRST_ACK_CODE	N	
<PARTNER>			N	
<NAME index="1">ACME</NAME>	Trading partner name . In this case Buyers name. Derived from OE_XML_PROCESS_UTIL.GET_SOLD_TO_EDITION.			
<ONETIME>0</ONETIME>	0			
<PARTNRID>AC01234</PARTNRID>	Internal customer number in the sellers system (OM).	H.SOLD_TO_ORG_ID		
<PARTNRIDX>1234</PARTNRIDX>	Unique identifier for the sold to partner. Value should be EDI Location code for the sold-to partner. Derived from OE_XML_PROCESS_UTIL.GET_SOLD_TO_EDITION.			

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG re qd Y/ N	Code Conv ersio n Need ed Y/N
<PARTNRTYPE>sold-to</PARTNRTYPE>	Partner Type. Value should be 'Sold To'			
</PARTNER>				
<PARTNER>				N
<NAME index="1">ACME</NAME>	Trading partner name. In this case Suppliers name			
<ONETIME>0</ONETIME>	0			
<PARTNRID>AC01234</PARTNRID>	Supplier number This will be the internal location number for the seller.	H.SHIP_FROM_ORG_ID		
<PARTNRIDX>1234</PARTNRIDX>	Unique identifier for the supplier. EDI location code for the supplier. Setup in Inventory/setup/Org/Location. This code is used by the buyer (PO) to identify the supplier.	ECX_OAG_CONT ROLAREA_TP_V. SOURCE_TP_LOCATION_CODE (XMLG)		
<PARTNRTYPE>Supplier</PARTNRTYPE>	Partner type. The value should be Supplier.			
</PARTNER>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<SALESINFO>			N	
<SALESPERSN>BOB</SALESPERSN>	Sales person name. Derived from OE_XML_PROCESS_UTIL.GET_SALES_PERSON			
</SALESINFO>				
</SOHEADER>				
<USERAREA>				
<REVISIONNUM>1</REVISIONNUM>	Po revision number. Om does not need to map this.			
<PORELEASE>1</PORELEASE>	Po release Number. OM does not need to map this.			
<ORACLE.SUPPLIER.DOCREF>A123</ORACLE.SUPPLIERDOCRE F>	Supplier reference for the Customer PO	H.ORIG_SYS_DOCUMENT_REF	N	Must have
< /USERAREA>				
</SOHEADER>				
<SOLINE>				
<SOLINENUM>1</SOLINENUM>	Sales order line number	L.LINE_NUMBER	Y	Must Have

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<SOLNSTATUS>OPEN</SOLNSTATUS>	Sales Order Line status (see notes)	L.FIRST_ACK_CO DE	N	Nice to Have
< ITEM>XXX<ITEM>	Supplier's item number This will be the item number from OM		N	Must Have
<DESCRIPTN>ACME WIDGET</DESCRIPTN>	Item Description		N	Nice to have
<ITEMX>XXX<ITEMX>	Buyer's item number	L.CUSTOMER_ITEM	N	Must Have
<OPERAMT qualifier="UNIT" type="T">	Unit selling price. Derived using Convert to OAG Operating Amount.	L.UNIT_SELLING_PRICE	N	Must have
<VALUE>6500000</VALUE>				
<NUMOFDEC>2</NUMOFDEC>				
<SIGN>+</SIGN>				
<CURRENCY>USD</CURRENCY>				
<UOMVALUE>1</UOMVALUE>				
<UOMNUMDEC>0</UOMNUMDEC>				
<UOM>EACH</UOM>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
</OPERAMT>				
<USERAREA>				
<USERITEMDESCRIPTN>itemdes </USERITEMDESCRIPTN>		L.USER_ITEM_DE SCRIPTION		
<ORACLE.SUPPLIERLINEREF>1 </ORACLE.SUPPLIERLINEREF>	Suppliers reference for the customer PO line	L.ORIG_SYS_LIN E_REF	N	Must have
</USERAREA>				
<SOLINE>				
<OPERAMT qualifier="UNIT" type="T">	Unit selling price. Derived using Convert to OAG Operating Amount.	L.UNIT_SELLING _PRICE	N	
<VALUE>6500000</VALUE>				
<NUMOFDEC>2</NUMOFDEC>				
<SIGN>+</SIGN>				
<CURRENCY>USD</CURRENCY> >				
<UOMVALUE>1</UOMVALUE>				
<UOMNUMDEC>0</UOMNUM DEC>				
<UOM>EACH</UOM>				
</OPERAMT>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<SOLINENUM>1</SOLINENUM>	Sales order line number	L.LINE_NUMBER	Y	
<ITEMX>XXX<ITEMX>	Buyer's item number	L.CUSTOMER_ITEM	N	
<SOLNSTATUS>OPEN</SOLNSTATUS>	Sales Order Line status (see notes)	L.FIRST_ACK_CODE	N	
<USERAREA>				
<USERITEMDESCRIPTN>itemdesc</USERITEMDESCRIPTN>		L.USER_ITEM_DESCRIPTION		
</USERAREA>				
<SOSCHEDULE>				
<SOSLINENUM>123</SOSLINENUM>	Schedule line number.	L.SHIPMENT_NUMBER	N	
<POLINENUM>123</POLINENUM>	Customer PO line number	L.ORIG_SYS_LINE_REF	N	
<PSCLINENUM>123</PSCLINENUM>	Customer PO schedule line number.	L.ORIG_SYS_SHIPMENT_REF	N	
<QUANTITY qualifier="ORDERED">	Quantity ordered. Derived using Convert To OAG Quantity.	L.ORDERED_QUANTITY	N	
<VALUE>1</VALUE>				
<NUMOFDEC>0</NUMOFDEC>				
<SIGN>+</SIGN>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<UOM>EACH</UOM>				
</QUANTITY>				Y
<DATETIME qualifier="NEEDELV">	Needed by delivery date as from the Original PO. This should be mapped to the request date in OM. Derived using Convert To OAG Datetime	L.REQUEST_DATE	N	
<YEAR>1998</YEAR>				
<MONTH>12</MONTH>				
<DAY>29</DAY>				
<HOUR>00</HOUR>				
<MINUTE>25</MINUTE>				
<SECOND>45</SECOND>				
<SUBSECOND>0000</SUBSECOND>				
</DATETIME>				
<DATETIME qualifier="DELIVSCHED">	Date time on which the goods are scheduled to be delivered. Derived using Convert To OAG Datetime	L.SCHEDULE_ARRIVAL_DATE	N	
<YEAR>1998</YEAR>				
<MONTH>12</MONTH>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
<DAY>29</DAY>				
<HOUR>00</HOUR>				
<MINUTE>25</MINUTE>				
<SECOND>45</SECOND>				
<SUBSECOND>0000</SUBSECOND>				
</DATETIME>				
<USERAREA>				
<ORACLE.SUPPLIERSHIPMENTREF>1	Suppliers reference for customer Shipment	L.ORIG_SYS_SHIPMENT_REF	N	Must have.
</ORACLE.SUPPLIERSHIPMENTREF>				
<ORACLE.SPLITLINE>N</ORACLE.SPLITLINE>	Flag indicating that this line is split.	Derived	N	Must have.
<ORACLE.SPLITFROMLINEREF>10</ORACLE.SPLITFROMLINEREF>	Line reference for the original line that this shipment has split from	L.SPLIT_FROM_LINE_ID	N	May have when line is split.
</USERAREA>				
</SOSCHEDULE>				
</SOLINE>				
</SHOW_SALESORDER>				

OAG Element	Desc/Comment	Oracle OM Table/Column	OAG required Y/N	Code Conversion Needed Y/N
</DATAAREA>				
</SHOW_SALESORDER_006>				

Message Map Notes

SO Status	SO Line Status
Open- Any status other than the following ones in Order Management.	Open- Any status other than the following ones in Order Management.
Closed- Closed status in Order Management	Shipped – Shipped status in Order Management
Cancelled- Cancelled status in Order Management	Closed- Closed status in Order Management
	Cancelled- Cancelled status in Order Management

Message Map

Name	Description
Message Map Name:	ONT_3A6_OAG72_OUT_SSO
Direction:	Outbound
(Internal) Transaction Type:	ONT
(Internal) Transaction Subtype:	SSO
External Transaction Type:	SO
External Transaction Subtype:	SHOW

Name	Description
DTD Directory:	xml/oag72
Map Directory:	patch/115/xml/US
Message Maps XGM File Name:	ONT_3A6_OAG72_OUT_SSO.xgm
Standard:	OAG
Release:	7.2
Format:	DTD
DTD Name:	SHOW_SALESORDER_006

Workflow Event Setup

Detail	Value
Event Name	oracle.apps.ont.oi.show_so.create
Event Description	Event for Outbound Show_SalesOrder
Subscription	R_SHOW_SALES_ORDER
Subscription Description	Event subscription for the oracle.apps.ont.oi.show_so.create event for Outbound Show_SalesOrder.

Message Set Up

To implement a message with a trading partner, use the XML Gateway message set up to define the trading partner or hub, code conversion values, and internal to external transaction name cross references. In addition, you may identify the XML Gateway system administrator to relate system or process errors.

Message Map ONT_3A6_OAG72_OUT_SSO.xgm

The message map ONT_3A6_OAG72_OUT_SSO is created using the Oracle XML Gateway Message Designer tool. Please refer to Message Map table for the detailed map

analysis.

The source DTD used is 091_show_salesorder_006.dtd, revision 7.2.1 of the Open Application Group. The other associated external reference DTD files are:

- oagis_domains.dtd
- oagis_resources.dtd
- oagis_fields.dtd
- oagis_segments.dtd

All the DTD's are checked in ONT source area under \$ont/xml/oag72.

The Source for the Outbound XML Show SO Message is the Order Management Acknowledgment tables OE_HEADER_ACKS & OE_LINE_ACKS.

The 091_show_salesorder_006 DTD is a three level hierarchy with order lines split into one or more shipment lines. The Order Management architecture is however a two level one with the order header and one or more lines. The message map will expand the two level structure of the Order Management tables to a three level XML Message.

Please refer to Table, "Message Map" for a detail analysis of the elements mapped, actions and derivation rules used by the Message Map.

Both the message map created using the Message Designer and its associated DTD's are stored in the database. The following Java programs are available to Load/Delete Maps or Load/Delete DTD's into/from the XML Gateway repository. Make sure you change your environment to your development environment and execute "source /ecxdev/ecx/utills/javaenv_wfidc.sh" to set up your Java environment.

Load/Delete Maps, Load/Delete DTD's

The following process is used only for customizations.

- java LoadMap <DB username> <DB password> <Hostname>:<Port>:<SID>
<mymap.xgm>

Example: java oracle.apps.ecx.loader.LoadMap apps apps ap505dbs:1521:dev115
ONT_3A6_OAG72_OUT_SSO.xgm

- java DeleteMap <DB username> <DB password> <Hostname>:<Port>:<SID>
<mapname>

Example: java oracle.apps.ecx.loader.DeleteMap apps apps ap505dbs:1521:dev115
ONT_3A6_OAG72_OUT_SSO.xgm

The Message Map is a .xgm file which will be stored in the Order Management Source area under \$ont/patch/115/map/ ONT_3A6_OAG72_OUT_SSO.xgm

Note: Maps and DTD's must be kept in sync between the Message

Designer and the XML Gateway repository. When in doubt, always reload the map and DTD as a pair.

Pre-Defined Workflow: Order Management Show_SalesOrder

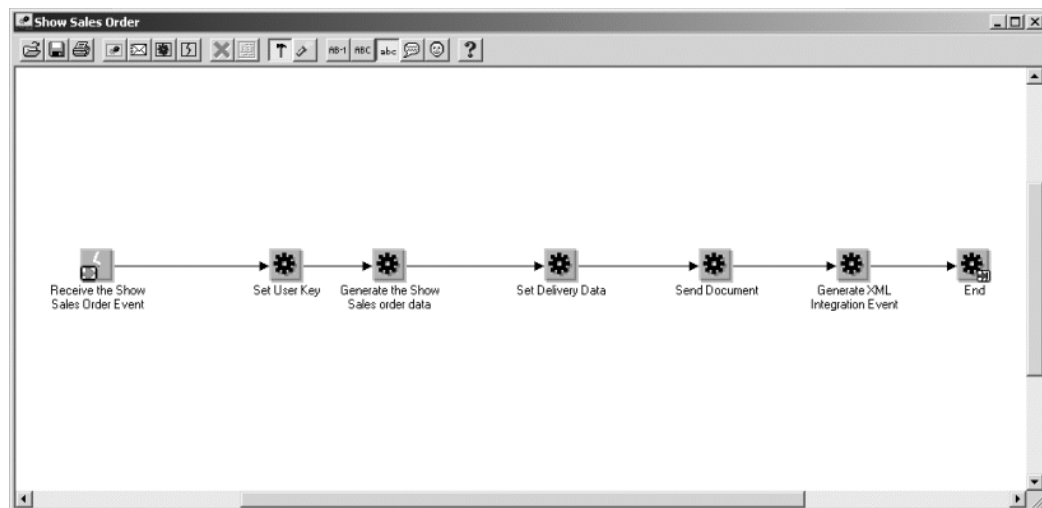
A pre-defined Workflow Process Show_SalesOrder is provided as part of OESO Workflow, which generates the Outbound Show_SalesOrder.

This Workflow receives the event to Show_SalesOrder, then generates the data using the Acknowledgment tables, and then use XML Gateway's SEND DOCUMENT function to deliver the Show_SalesOrder XML to the trading partner.

The Show Sales Order concurrent program is available in the Concurrent Requests window.

Seeded Workflow

The figure below displays the Show_SalesOrder Workflow:



Change_salesorder

Overview

Using the Change_salesorder, the seller can:

- Acknowledge acceptance of a pending purchase order line
- Initiate changes to a purchase order

You can set up a requirement for a response document for seller-initiated changes within the Trading Partner Agreement. For example, the buyer must respond with

PIP3A8, Request Purchase Order Change, to accept the proposed changes. If the changes are not accepted, PIP3A9, request Purchase Order Cancel must be sent by the buyer.

Both Order Import of closed orders and HVOP are currently not supported for XML orders.

Generate a Change_salesorder Message Outbound From Order Management

Oracle Order management generates a Change_salesorder message when the sales order is changed either manually or by the process_order API. The change_salesorder may also be generated in case of change of status of the sales order from Pending to Accept or Reject, and notify to the buyer of the change.

The following are the assumed definitions for Accept, Reject, and Pending:

- Accept - Booked status
- Reject - Order cannot be Entered and is rejected.
- Pending - Entered status.

Note: While the RosettaNet standard allows Pending lines to go to Reject, in Order Management, this implies that 3A7 would be sent when an Entered order is rejected. However, since currently all of the 3A7 (and 3A6) processing requires an Order to be Booked, triggering 3A7 on the transition from Pending to Reject is not supported.

Place an Order on Hold and Clear Holds

When changes occur, a Change_salesorder message notifies the buyer to put the order on hold. This hold can be cleared as follows:

Changes are made to the order after booking and the order is put on hold before sending 3A7 to the buyer

Purchasing sends a Request Purchase Order Change (3A8) transaction verifying the acceptance of the proposed changes.

- The 3A8 Response is consumed and the order line attributes mentioned below are compared. If any of these attributes do not match the current data on the order line, the 3A8 Response is rejected.

If the attributes do match, then the 3A8 Response is accepted and the hold is removed. The order progresses through the processing.

Note: If there is a 3A8 that comes in as an independent request

before 3A7 response then the hold is removed and the 3A8 request will be acted upon. If there is a subsequent 3A8 that arrives as a response for the same 3A7, 3A8 is processed and depending if the response has values that match the order attributes mentioned below or it will be accepted or rejected.

- If purchasing rejects the 3A7 proposed changes, then purchasing sends a Request Purchase Order Cancel (3A9) transaction, canceling the original purchase order.

Order Management receives and consumes the Request Purchase Order Cancel (3A9) and cancels the purchase order within Order Management.

If Cancel PO is received for the lines on hold the lines are removed from hold and cancelled.

If Oracle Purchasing rejects the 3A7 proposed changes, then Purchasing can also send back a 3A8 Response using the ACKCODE tag with a value of 2 to communicate Reject. Note that no value or a value of 0 in this tag implies Accept.

If the ACKCODE tag has a value of 2 at either Line or Shipment level for a Change PO Response, Order Management does not consume that line. However, Order Management does process the Accepted lines (i.e. ACKCODE tag has no value or a value of 2).

If the ACKCODE tag has a value of 2 at Header level, Order Management doesn't consume the entire document.

This feature of Accepting and Rejecting changes is not required - if the ACKCODE tag is not populated, Order Management will automatically process every line to check if the attributes match the values currently on the Sales Order.

For the implementations where the seller chooses not to require a response for 3A7 there is a setup to state that the response is not required for the 3A7 created by the seller. In this case the order is put on hold. A profile option OM: Change SO Response Required controls the response at site level with values Y or N.

Ability to Match Incoming Change_PO (3A8) or Cancel_PO (3A9) to The Original Change_salesorder (3A7)

Order Management must be able to match the incoming 3A8 (which can be a response to the 3A7 originated by Order Management) or 3A9. The match can be done using the following two criteria together:

- Customer purchase order number + Trading partner (buyer)id
- Inbound 3A8 will have a flag, that indicates if the specific 3A8 is a response (to 3A7) or is a new request. This flag separates the responses from the independent 3A8 that may occur for the same order for which 3A7 was generated.
- 3A9 is processed and results in the removal of the hold and cancellation (partial or

full) of the lines.

Integration With Supply Chain Trading Connector (CLN)

Supply Chain Trading Connector records the transaction history when the trading partners exchange messages. Order Management incorporates the API calls during the processing of the XML messages.

The following point within the transaction dialog is recorded in the CLN module:

- Change sales order message sent
- One or more buyer rejected lines are received in the 3A8 Change PO response to an earlier 3A7 Change Sales Order message.
- 3A8 Change PO response to an earlier 3A7 Change Sales Order message is rejected by the buyer at the header level.

Message Map

3A7 Message Map

The message map is created using Oracle XML Gateway's message designer.

Map Name - ONT_3A7_OAG72_OUT_SO

DTD Name - 087_change_salesorder_008.dtd (Change Sales Order)

Note: Most data values in the message map are seeded.

Load/Delete Maps, Load/Delete DTD's

The following steps are carried out only for customizations.

1. `java LoadMap DB username> DB password> Hostname>:Port>:SID> mymap.xgm>`

Example: `java oracle.apps.ecx.loader.LoadMap apps apps ap505dbs:1521:dev115 ONT_3A7R_OAG72_OUT_SO.xgm`

2. `java DeleteMap DB username> DB password> Hostname>:Port>:SID> mapname>`

Example: `java oracle.apps.ecx.loader.DeleteMap apps apps ap505dbs:1521:dev115 ONT_3A7R_OAG72_OUT_SO.xgm`

The Message Map is a .xgm file which will be stored in the Order Management Source area under \$ont/patch/115/map/ ONT_3A7R_OAG72_OUT_SO.xgm.

Note: Maps and DTD's must be kept in sync between the Message

Designer and the XML Gateway repository. When in doubt, always reload the map and DTD as a pair.

Transaction

The following is the seeded data for the 3A7 transaction:

- Transaction Type – ONT
- Transaction Sub Type – CSO
- External Transaction Type – SALESORDER
 - External Transaction Sub Type – CHANGE

ONT_3A7_OAG72_OUT_SO

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<CHANGE_SALES ORDER_008>			
<CNTROLAREA>		Y	
<BSR>		Y	
<VERB>CHANGE< /VERB>		Y	
<NOUN>SALESOR DER</NOUN>		Y	
<REVISION>008</R EVISION>		Y	
</BSR>		Y	
<SENDER>		Y	
<LOGICALID>XGR XMLG B1109</LOGICALI D>		Y	

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<COMPONENT>SALES</COMPONENT>	XMLG	Y	
<TASK>OM</TASK>	XMLG	Y	
<REFERENCEID>9534223449</REFERENCEID>	XMLG	Y	
<CONFIRMATION>1</CONFIRMATION>	XMLG	Y	
<LANGUAGE>ENG</LANGUAGE>	XMLG	Y	
<CODEPAGE>CP001001</CODEPAGE>	XMLG	Y	
<AUTHID>SMITHJ</AUTHID>	XMLG	Y	
</SENDER>		Y	
<DATETIME qualifier="CREATION">	XMLG	Y	
<YEAR>1998</YEAR>		Y	
<MONTH>11</MONTH>		Y	
<DAY>21</DAY>		Y	

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<HOUR>00</HOUR>		Y	
<MINUTE>25</MINUTE>		Y	
<SECOND>45</SECOND>		Y	
<SUBSECOND>0000</SUBSECOND>		Y	
<TIMEZONE>-0600</TIMEZONE>		Y	
</DATETIME>		Y	
</CONTROLAREA>		Y	
<DATAAREA>		Y	
<CHANGE_SALES ORDER>		Y	
<SOHEADER>		Y	
<SALESORDID>S0001</SALESORDID>	Sales order number	Y	Oe_Header_Acks.ORDER_NUMBER
<SALESORG index="1">ORG12345</SALESORG>	Sales Organization code	Y	
<NOTES1>Invalid ID</NOTES1>	Accept or Reject? (See notes)	N	"Accepted"
<POID>1234</POID>	Customer PO number	Y	Oe_Header_Acks.CUSTOMER_PO_NUMBER

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<SOSTATUS>Open </SOSTATUS>	Sales Order Status (see notes)	N	Oe_Header_Acks.FIRST_ACK_CODE
<USERAREA>		N	
<PORELEASE>1</PORELEASE>	PO release number. No map to OM column	N	
<REVISIONNUM>2</REVISIONNUM>	PO revision number. No map to OM column	N	
<ORACLE.SUPPLIERDOCREF>123</ORACLE.SUPPLIERDOCREF>	Suppliers reference for the PO	N	Oe_header_Acks.Orig_Sys_Document_Ref
<ORACLE.OWNER>>ORACLE.OWNER>End Customer	IB Owner	N	OE_HEADERS_INTERFACE.IB_OWNER
<ORACLE.CURRENTLOCATION>Ship To Location</ORACLE.CURRENTLOCATION>	IB Current Location	N	OE_HEADERS_INTERFACE.CURRENT_LOCATION
<ORACLE.INSTALLEDATLOCATION>Bill To Location</ORACLE.INSTALLEDATLOCATION>	IB Installed At Location	N	OE_HEADERS_INTERFACE.IB_INSTALLED_AT_LOCATION
</USERAREA>		N	
<PARTNER>		N	

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<NAME index="1">ACME</ NAME>	Name of Sold To Trading Partner	Y	Derived from Oe_Header_Acks.S OLD_TO_ORG_ID
<ONETIME>0</ON ETIME>		Y	
<PARTNRID>YYY Y</PARTNRID>	Unique ID for Sold To Trading Partner	Y	Oe_Header_Acks.S OLD_TO_ORG_ID
<PARTNRTYPE>So ldTo</PARTNRTY PE>		Y	
<PARTNRIDX>XX XX</PARTNRIDX>	EDI Location Code for Sold To Trading Partner	N	Derived from Oe_Header_Acks.S OLD_TO_ORG_ID
</PARTNER>			
<PARTNER>			
<NAME index="1">ACME</ NAME>	Name of supplier		
<ONETIME>0</ON ETIME>		N	
<PARTNRID>AC01 234</PARTNRID>	Unique id for the supplier	Y	Oe_Header_Acks.S HIP_FROM_ORG_I D
<PARTNRIDX>AC 01234</PARTNRID X>	Location code of the supplier	N	Ecx_Oag_ControlA rea_Tp_V.SOURCE _TP_LOCATION_ CODE

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<PARTNRTYPE>SUPPLIER</PARTNRTYPE>		Y	
</PARTNER>		Y	
</SOHEADER>		Y	
<SOLINE>		Y	
<OPERAMT qualifier="UNIT" type="T">	Unit price	N	Derived from Oe_Line_Acks.UNIT_SELLING_PRICE
<VALUE>1</VALUE>		N	
<NUMOFDEC>0</NUMOFDEC>		N	
<SIGN>+</SIGN>		N	
<CURRENCY>USD</CURRENCY>			
<UOMVALUE>1</UOMVALUE>			
<UOMNUMDEC>0</UOMNUMDEC>		N	
<UOM>EACH</UOM>		Y	
</OPERAMT>		Y	
<QUANTITY qualifier="ORDERED">	Ordered quantity	Y	

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<VALUE>1</VALUE>		Y	
<NUMOFDEC>0</NUMOFDEC>		Y	
<SIGN>+</SIGN>		Y	
<UOM>EACH</UOM>		Y	
</QUANTITY>		Y	
<SOLINENUM>1</SOLINENUM>	Sales order line number		Oe_Line_Acks.LINE_NUMBER
<NOTES1/>	Accept or Reject? (See notes)	N	"Accepted"
<POLINENUM>1</POLINENUM>	Customer po line number	Y	Oe_Line_Ack.CUSTOMER_LINE_NUMBER
<SOLNSTATUS>Open</SOLNSTATUSES>	Sales Order Line status (see notes)	Y	Oe_Line_Acks.FIRST_ACK_CODE
<ITEM>123456</ITEM>	Supplier Item number	Y	
<ITEMX>123456</ITEMX>	Buyer item number	Y	Oe_Line_Acks.CUSTOMER_ITEM
<USERAREA>			
<ORACLE.SUPPLIERLINE>1</ORACLE.SUPPLIERLINE>	Suppliers reference for customer PO line	N	Oe_Line_Acks.ORIG_SYS_LINE_REF

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
</USERAREA>			
<SOSCHEDULE>		N	
<DATETIME qualifier="DELIVSCHED">	Scheduled arrival date	N	Derived from Oe_Line_Acks.SCHEDULE_ARRIVAL_DATE
<YEAR>1998</YEAR>		N	
<MONTH>11</MONTH>		N	
<DAY>21</DAY>			
<HOUR>00</HOUR>			
<MINUTE>25</MINUTE>			
<SECOND>45</SECOND>			
<SUBSECOND>0000</SUBSECOND>		Y	
<TIMEZONE>-0600</TIMEZONE>		Y	
</DATETIME>			
<DATETIME qualifier="NEEDELV">	Requested ship date	Y	Derived from Oe_Line_Acks.REQUEST_DATE
<YEAR>1998</YEAR>			

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<MONTH>11</MONTH>		N	
<DAY>21</DAY>		Y	
<HOUR>00</HOUR>		Y	
<MINUTE>25</MINUTE>		Y	
<SECOND>45</SECOND>		Y	
<SUBSECOND>0000</SUBSECOND>			
<TIMEZONE>-0600</TIMEZONE>		Y	
</DATETIME>		N	
<QUANTITY qualifier="ORDERED">	Scheduled quantity	N	Derived from Oe_Line_Acks.ORDERED_QUANTITY, ORDER_QUANTITY_UOM
<VALUE>1</VALUE>		Y	
<NUMOFDEC>0</NUMOFDEC>		Y	
<SIGN>+</SIGN>		Y	
<UOM>EACH</UOM>		Y	

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
</QUANTITY>			
<SOSLINENUM>1</SOSLINENUM>	Sales Order shipment number	Y	Oe_Line_Acks.SHIPMENT_NUMBER
<PSCLINENUM>1</PSCLINENUM>	Customer schedule line number	N	Oe_Line_Acks.CUSTOMER_SHIPMENT_NUMBER
<USERAREA>			
<ORACLE.SUPPLIERSHIPMENTREF>1</ORACLE.SUPPLIERSHIPMENTREF>	Suppliers reference for customer shipment	N	Oe_line_Acks.ORIG_SYS_SHIPMENT_REF
<ORACLE.SPLITLINE>N</ORACLE.SPLITLINE>			
<ORACLE.SPLITFROMLINEREF>1</ORACLE.SPLITFROMLINEREF>	Split from line reference	N	Oe_line_Acks.SPLIT_FROM_LINE_ID
<DATETIME qualifier="SHIPSCHED">	Schedule ship date	N	Derived from Oe_Line_Acks.SCHEDULE_SHIP_DATE
<YEAR>1998</YEAR>		N	
<MONTH>11</MONTH>		N	
<DAY>21</DAY>		N	

OAG	Comment/Description	OAG Req'd Y/N	OM table/Column
<HOUR>00</HOUR>		N	
<MINUTE>25</MINUTE>		N	
<SECOND>45</SECOND>		N	
<SUBSECOND>0000</SUBSECOND>		N	
<TIMEZONE>-0600</TIMEZONE>		N	
</DATETIME>		N	
</USERAREA>			
</SOSCHEDULE>			
</SOLINE>			
</CHANGE_SALESORDER>			

Statuses and Descriptions

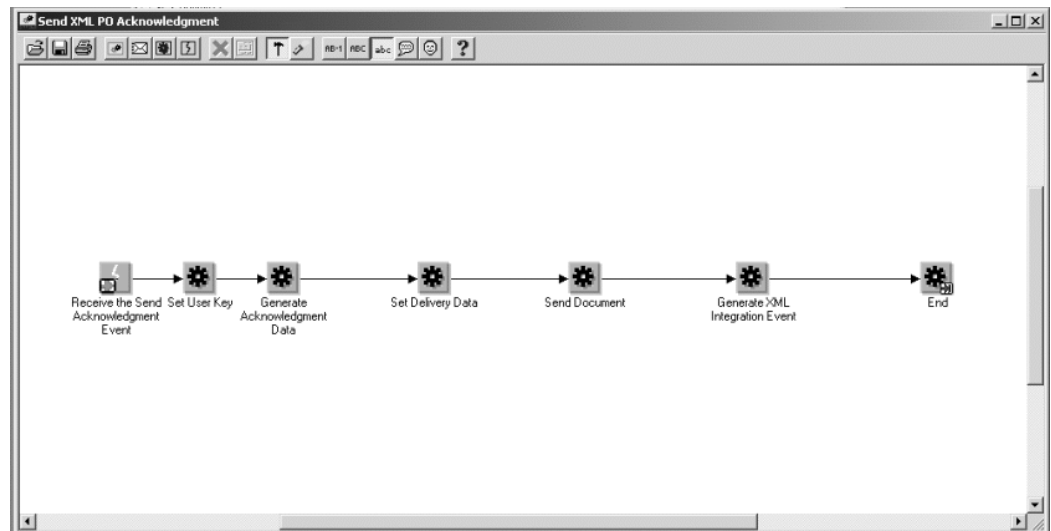
Status	Description
So Status	Open- Any status other than the following ones in Order Management.
	Cancelled- Cancelled status in OM

So Line Status	Open- Any status other than the following ones in Order Management. On Hold – Line is on 3A7 Change Notification Hold Cancelled- Cancelled status in Order Management
NOTES (Header and Line level)	The Notes can be used in case of 3A7 being sent in order to change Pending status to either Accept or Reject. In case of Reject it is necessary to populate this element. Possible values - Accepted

Pre-Defined Workflow Event

If the profile option OM: Run Order Import for XML is set to Synchronous, a pre-defined workflow event is seeded to run Order Import if the Profile Option OM: Process Transaction is set to ONLINE. This workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the workflow process, it runs Order Import for the current XML Message. After Run Order Import completes (Successfully or With Error), the process that raises the business event is called.

Seeded Workflow



Message Details

This table describes the data types and fields in the DTD used by the message map. The message map may remove used fields so that empty data tags are neither generated for outbound transactions nor examined by inbound transactions.

Message Map Details

Message Map Details

Message Map Name:	ONT_3A7_OAG72_OUT_SO
Direction:	OUT
(Internal) Transaction Type:	ONT
(Internal) Transaction Subtype:	CSO
External Transaction Type:	SALESORDER
External Transaction Subtype:	CHANGE
DTD Directory:	ont/xml/oag72
Map Directory:	patch/115/xml/US
Message Maps XGM File Name:	ONT_3A7_OAG72_OUT_SO.xgm
Standard:	OAG
Release:	7.2
Format:	DTD
DTD Name:	087_change_salesorder_008.dtd

Workflow Event Setup

The same event/subscription/workflow that was developed for Show sales order is reused for Change Sales Order.

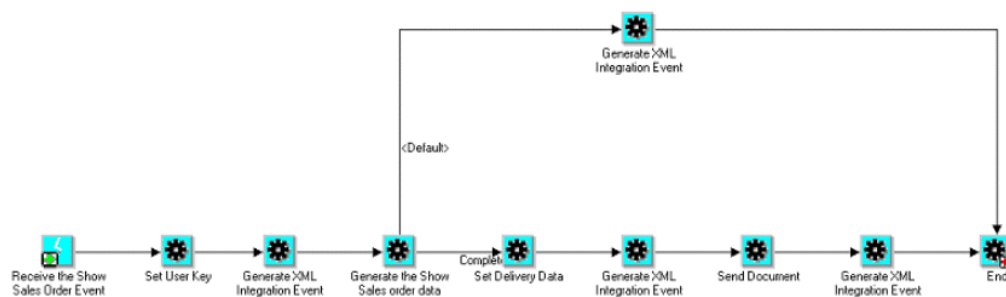
Workflow Event Setup

Detail	Value
Event Name	oracle.apps.ont.oi.show_so.create

Detail	Value
Event Description	Event for 3A6 Outbound Show sales order
Subscription	R_SHOW_SALES_ORDER
Subscription Description	Event subscription for the oracle.apps.ont.oi.show_so.create event for Outbound Show sales order.

Workflow (oexwfoa.wft)

See figure. Process R_SHOW_SALES_ORDER



Lookups

Lookups

Lookup Type	Code	Meaning	Enabled
HOLD_TYPE	CHANGE_NOTIFICATION_HOLD	Change Notification Hold for 3A7	No
RELEASE_REASON	3A7_RESPONSE_RECEIVED	Response to 3A7 is received. Hold released automatically.	No

3A7 Hold

Hold Name	Hold Type	Description	Hold ID
3A7 Change Notification Hold	CHANGE_NOTIFICATION_HOLD	Line level hold - Waiting for a response on 3A7	56

Inbound Change PO Request

Overview

The Inbound Change PO Request enables a buyer to change a purchase order and the seller to acknowledge if the changes are accepted, rejected, or pending. Depending on the response, the trading partner can do the following:

- Change purchase orders
- Acknowledge whether the buyer accepts the changes made by the seller

Major Features

Consume the Change_PO Message Inbound From Purchasing

Oracle Order Management consumes the Change_PO message inbound from Purchasing, thereby updating the purchase order previously entered.

Changes to the purchase order follow the same rules for changing a purchase order that currently exist in Order Management.

Scenario 1: Redistribution of the quantity over shipments (Oracle Purchasing)

Original PO

Redistribution - Before

Line	Shipment	Qty
1	1	5
1	2	5

Line	Shipment	Qty
1	3	5

If the quantity of the 3 shipments in above example is redistributed over 2 shipments
Change PO looks as follows:

Redistribution - Two Shipments

Line	Shipment	Qty
1	1	7
1	2	8
1	3	0

Scenario 2: Cancellation of a shipment (Oracle Purchasing)

Cancellation of one of the shipments can cause a Change PO to start (Oracle Purchasing).

Original PO

Cancellation of a Shipment

Line	Shipment	Qty
1	1	6
1	2	5
1	3	5

If Shipment 3 is cancelled Change PO would look as follows:

After Cancellation

Line	Shipment	Qty
1	3	0

Changes supported

Change PO supports change in the following elements:

- Bill to location
- Ship to location (Schedule level)
- Quantity (schedule level)
- Need by date (Schedule level)
- Price
- Payment terms

Set up the Order Import Process to be Run in Synchronous or Asynchronous Mode

You can setup a profile option to start Order Import for every order or wait and process multiple orders via concurrent processing. The Profile Option is called OM: Run Order Import for XML, and can be set up at the site level. The possible values for this profile option are Asynchronous and Synchronous.

Integration With Supply Chain Trading Connector (CLN)

The Supply Chain Trading Connector module records the transaction history when the trading partners exchange messages. To enable this Order Management has incorporated the API calls during the processing of the XML messages. In the case of 3A8 Change PO request message collaboration history is recorded at the key points in processing Change PO.

- When Change PO comes in and is consumed by Order Management by entering the data into Order Import tables (and before running the Order Import Process)
- Before running Order Import (if Order Import is going to be run immediately)
OR
- If Order Import is not going to be run and is going to be run in batch mode later on
- Order Import completed successfully /Failed

Support For Split Lines

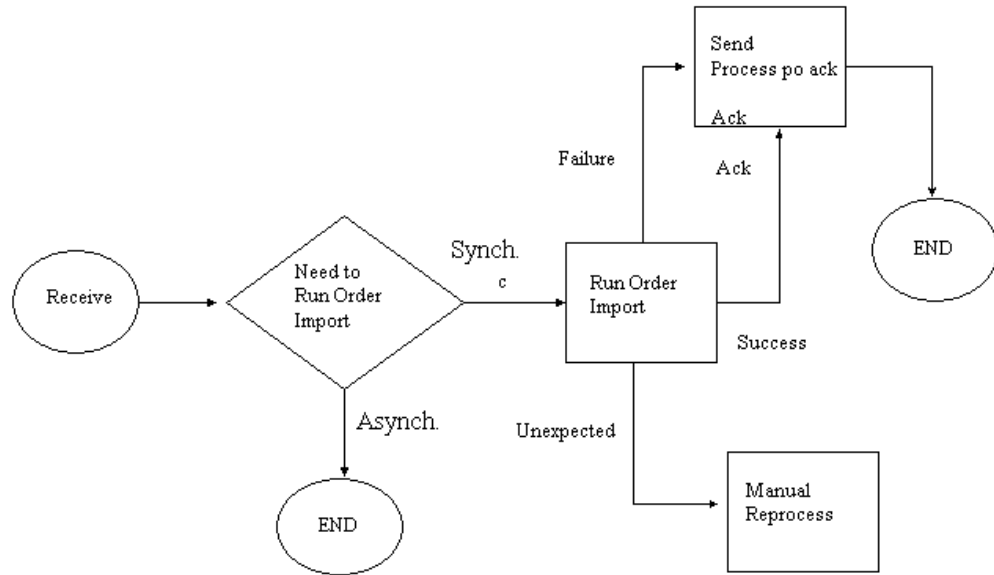
Order Management supports splitting of lines for multiple reasons. When the line is split, new shipment information is created for the line and stored in Order Management. In this case the 3A7 is sent to the buyer notifying of this change. Upon receiving the information about the new line buyer's system sends the acknowledgement either accepting or rejecting this new line. If the new line is accepted by the buyer, the buyer sends the 3A8 (change PO) as a response with the new buyer information as well as the reference for the new line sent by the seller.

If the line is rejected by the buyer, 3A8 can be sent by the buyer to cancel the split or original line.

Error Handling

The error handling process deals with three possible types of outcomes that occur after the order import process is run:

- **Successful processing** - In this case the confirmation API is called to send the Acknowledgment out to the sender with the Accept status.
- **Failed processing** - This is the status when there are validation errors or data setup errors. In this case the confirmation API is called to send the acknowledgment out to the sender with the Reject Status.
- **Unexpected Errors** - There can be some unexpected errors due to resource constraints and will require reprocessing of the Order Import upon fixing those issues. This process is manual.



Message Map

ONT_3A8R_OAG72_IN.xgm

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<CHANGE_PO_006>				
<CNTROLAREA>				
<BSR>		Y		
<VERB value="CHANGE">CHANGE</VERB>		Y		
<NOUN value="PO">PO</NOUN>		Y		
<REVISION value="006">006</REVISION>		Y		
</BSR>		Y		

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<SENDER>		Y		
<LOGICALID> XX141HG09 </LOGICALID>		Y		
<COMPONENT> PUR </COMPONENT>		Y		
<TASK> MAINT </TASK>		Y		
<REFERENCEID> 95129945823449 </REFERENCEID>		Y		
<CONFIRMATION> 1 </CONFIRMATION>		Y		
<LANGUAGE> ENG </LANGUAGE>		Y		
<CODEPAGE> test </CODEPAGE>		Y		
<AUTHID> CMKURT </AUTHID>		Y		
</SENDER>				
<DATETIME qualifier= "CREATION ">	Timestamp for PO (Standard or Release) creation	Y		
<YEAR> 1998 </YEAR>		Y		
<MONTH> 11 </MONTH>		Y		
<DAY> 20 </DAY>		Y		

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table/Column
<HOUR>17</HOUR>		Y		
<MINUTE>06</MINUTE>		Y		
<SECOND>45</SECOND>		Y		
<SUBSECOND>0000</SUBSECOND>		Y		
<TIMEZONE>-0600</TIMEZONE>		Y		
</DATETIME>				
</CNTROLAREA>				
<DATAAREA>				
<CHANGE_PO>				
<POHEADER>				
<POID>a</POID>	<p>Unique ID for the PO.For Standard POs, the PO number from Oracle</p> <p>Purchasing is entered here. For Sales Agreement</p> <p>Releases, it should be Sales Agreement PO#-Release#.</p>	Y	OE_HEADERS_INTERFACES.Cust_po_number	PO_ECX_HEADER_ARCH - V.contractb PHA.SEGMENT1
<NOTES>Notes</NOTES>		N		

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<REASONCODE>Admin Error</REASONCODE>	Reason for the change	N	OE_HEADERS_INTERFACE.reason	
<USERAREA>				
<ORGID>	Org Id		OE_HEADERS_INTERFACE.org_id	
<ORACLE.CHANGES EQUENCE>12</ORACLE.CHANGES EQUENCE>	Change Sequence		OE_HEADERS_INTERFACE.change_sequence	
<ORACLE.BOOKEDFLAG>Y</ORACLE.BOOKEDFLAG>	Booked Flag		OE_HEADERS_INTERFACE.booked_flag	
<MSGTYPE>Response</MSGTYPE>	Response Flag		OE_HEADERS_INTERFACE.response_flag	
<ACKCODE>	A value of 2 for a 3A8 Change PO response document implies a header level buyer rejection and will result in the entire document not being processed.			

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<ORACLE.SUPPLIER DOCREF>122 </ORACLE.SUPPLIER DOCREF>	Suppliers reference for the Customer PO. Needs to be sent back by the buyer as a reference for the newly created line on 3A8 response as well as 3A9. Also needed to be sent on every new change request for the new line.	N	OE_HEADE RS_INTERF ACE.ORIG_ SYS_DOCU MENT_REF	
</USERAREA>				
<PARTNER>	Supplier This data type provides information about the trading partner. No need to map this segment for OM	Y		
<NAME1>abcd</NAME1>	Name of the trading partner			PO_ECX_HE ADER_ARCH _V.supp_org_ name PO_VENDOR S.VENDOR_N AME
<ONETIME>a </ONETIME>	Indicator of whether this partner is Established for this transaction only.	Y		PO_ECX_HE ADER_ARCH _V.supp_otf DECODE (PO_VENDO RS.ONE_ TIME_FLAG,' N','0','1')

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<PARTNRID>a </PARTNRID>	Unique identifier for the partner in Oracle Applications (Supplier Number).	Y		PO_ECX_HE ADER_ARCH _V.supplier_p artner_id PO_VENDOR S.SEGMENT1
<PARTNRTYPE> Supplier </PARTNRTYPE>		Y		
<CURRENCY>a</CUR RENCY>	Preferred operating currency of the partner.			PO_ECX_HE ADER_ARCH _V.supp_Curr ency PO_VENDOR S.PAYMENT_ CURRENCY_ CODE
<PARTNRIDX>a </PARTNRIDX>	Unique identifier of the partner Supplier.	Y		PO_ECX_HE ADER_ARCH - V.supplier_pa rtner_id_x NVL(PO_VE NDOR_SITES _ALL_ECE_T P_LOCATIN_ CODE,PHA.V ENDOR_ID
<ADDRESS>	Address of the partner			
<ADDRLINE index="1 >a</ADDRLINE>				
<CITY>a</CITY>				

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<COUNTRY>a </COUNTRY>				
<DESCRIPTN>a </DESCRIPTN>				
<FAX index="1">a </FAX>				
<POSTALCODE>a </POSTALCODE>				
<STATEPROVN>a </STATEPROVN>				
<TELEPHONE index="1">a</TELEPHONE>				
</ADDRESS>				
</PARTNER>				
<PARTNER>	Sold To This data type provides information about the trading partner	Y		
<NAME1>abcd</NAME1>	Name of the buyer company			PO_ECX_HEADER_ARCH _V.buying_organization _NAME HR_ALL_ORGANIZATION_UNITS_TL _NAME

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<ONETIME>a </ONETIME>	Indicator of whether this partner is Established for this transaction only.			PO_ECX_HE ADER_ARCH _V.buying_ org_otf '0
<PARTNRID>a </PARTNRID>	Unique identifier for the partner in Oracle Applications (Supplier Number).			
<PARTNRTYPE>Sold To </PARTNRTYPE>	Sold To			
<CURRENCY>a</CURRENCY>	Preferred operating currency of the partner.			PO_ECX_HE ADER_ARCH _V.buying_or g_currency GL_SETS_OF_ BOOKS.CURR ENCY_ CODE
<PARTNRIDX>a </PARTNRIDX>	Unique identifier of the partner.		OE_HEADE RS_INTERF ACE.sold_to _org_id	PO_ECX_HE ADER_ARCH _V.partner_id _x HR_LOCATI ONS_ALL.EC E_TP_LOCAT ION_CODE
<TAXID>a</TAXID>	Tax identifier			
<ADDRESS>	Address of the partner. This information needs to be mapped for the case where EDI location code is not sent.			

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
	<ADDRLINE index="1 >a</ADDRLINE>			HR_ORGANI ZATION_UNI TS_ V.ADDRESS_ LINE_1-3
	<CITY>a</CITY>			HR_ORGANI ZATION_UNI TS_ V.TOWN_OR _CITY
	<COUNTRY>a </COUNTRY>			PO_ECX_HE ADER_ARCH _V.buying_ org_country
	<COUNTY>a </COUNTY>			OE_HEADER S_INTERFAC E.INVOICE_C OUNTY
	<POSTALCODE>a </POSTALCODE>			HR_ORGANI ZATION_UNI TS_ V.POS TAL _CODE
	<STATEPROVN>a </STATEPROVN>			HR_ORGANI ZATION_UNI TS_ V.RE GI ON2
	<TELEPHONE index="1 >a</TELEPHONE>			HR_ORGANI ZATION_UNI TS_ V.TELEPHON E_NUMBER_ 1-3

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
</ADDRESS>				
</PARTNER>				
<PARTNER>	Bill to partner- This is an optional segment			
<NAME1>abcd</NAME1>	Name of the bill to partner			PO_ECX_HE ADER_ARCH _V.buying_or g_name HR_ALL_OR GANIZATIO N_UNITS_TL .NAME
<ONETIME>a </ONETIME>				PO_ECX_HE ADER_ARCH _V.billto_ org_otf '0
<PARTNRID>a </PARTNRID>				PO_ECX_HE ADER_ARCH _V.partner_id PHA.ORG_ID
<PARTNRTYPE>Bill to</PARTNRTYPE>				
<CURRENCY>a</CURRENCY>				FINANCIALS _SYSTEM_PA RAMS_ ALL.PAYME NT_CURREN CY_CODE

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<PARTNRIDX>a </PARTNRIDX>			OE_HEADER RS_INTERFACE ACE.invoice _to_org_id	HR_LOCATIONS ALL.EC E_TP_ LOCATION_ CODE
<ADDRESS>	The address information needs to be mapped for the case where EDI Location code is not sent.			
<ADDRLINE index="1 >a</ADDRLINE>			OE_HEADER RS_INTERFACE ACE.INVOICE CE_ADDRESS SS1-3	HR_LOCATIONS ALL.ADD RESS_ LINE_1-3
<CITY>a</CITY>			OE_HEADER RS_INTERFACE ACE.INVOICE CE_CITY	HR_LOCATIONS ALL.TOWN_OR_ CITY
<COUNTRY>a </COUNTRY>			OE_HEADER RS_INTERFACE ACE.INVOICE CE_COUNTRY	HR_LOCATIONS ALL.COUNTRY
<COUNTY>a </COUNTY>				OE_HEADERS INTERFACE SHIP_TO_C OUNTY
<POSTALCODE>a </POSTALCODE>			OE_HEADER RS_INTERFACE ACE.INVOICE CE_POSTAL _CODE	HR_LOCATIONS ALL.POSTAL _CODE

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table/Column
<STATEPROVN>a </STATEPROVN>			OE_HEAD RS_INTER ACE.INVOI CE_STATE	HR_LOCATI ONS_ALL.RE GION_2
<TELEPHONE index="1">a</TELEPHONE>				HR_LOCATI ONS_ALL.TE LEPHONE_ NUMBER_1-3
</ADDRESS>				
</PARTNER>				
<PARTNER>	Ship to partner- This is an optional segment			
<NAME1>abcd</NAME1>	Name of the Ship to partner			
<ONETIME>a </ONETIME>				
<PARTNRID>a </PARTNRID>				
<PARTNRTYPE>Ship To</PARTNRTYPE>				
<CURRENCY>a</CURRENCY>				
<PARTNRIDX>a </PARTNRIDX>			OE_HEAD RS_INTER ACE.ship_to _org_id	

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<ADDRESS>	The address information needs to be mapped for the case where EDI Location code is not sent.			
<ADDRLINE index="1">a</ADDRLINE>			OE_HEADE RS_INTERF ACE.SHIP_T O_ADDRES S1-3	
<CITY>a</CITY>			OE_HEADE RS_INTERF ACE.SHIP_T O_CITY	
<COUNTRY>a</COUNTRY>			OE_HEADE RS_INTERF ACE.SHIP_T O_COUNTR Y	
<POSTALCODE>a</POSTALCODE>			OE_HEADE RS_INTERF ACE.SHIP_T O_POSTAL_ CODE	
<STATEPROVN>a</STATEPROVN>			OE_HEADE RS_INTERF ACE.SHIP_T O_STATE	
<TELEPHONE index="1">a</TELEPHONE>				
</ADDRESS>				
<Partner>	End Customer - This is an optional segment			

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<NAME1>abcdNAME 1>abcd>	Name of the End Customer		OE_HEADE RS_INTERF ACE.END_C USTOMER_ NAME	
<ONETIME>aONETI ME>a>				
<PARTNRID>aPARTN RID>a>			OE_HEADE RS_INTERF ACE.END_C USTOMER_ NUMBER	
<PARTNRTYPE>EndP ARTNRTYPE>End>				
<CURRENCY>aCURR ENCY>a>				
<PARTNRIDX>aPART NRIDX>a>				
<ADDRESS>				
<ADDRLINE index="1">aADDRLIN E index="1">a>			OE_HEADE RS_INTERF ACE.END_C USTOMER_ ADDRESS1- 4	
<CITY>a</CITY>			OE_HEADE RS_INTERF ACE. END_CUST OMER _CITY	

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
	<COUNTRY>a</COUNTRY>		OE_HEADE RS_INTERF ACE. END_CUST OMER_COU NTRY	
	<COUNTY>a</COUNTY>		OE_HEADE RS_INTERF ACE. END_CUST OMER_COU NTY	
	<POSTALCODE>a</POSTALCODE>		OE_HEADE RS_INTERF ACE. END_CUST OMER_POS TAL_CODE	
	<STATEPROVN>a</STATEPROVN>		OE_HEADE RS_INTERF ACE. END_CUST OMER STATE	
	<TELEPHONE index="1">a</TELEPHONE>			
	</ADDRESS>			
	<CONTACT>			
	<NAME>		OE_HEADE RS_INTERF ACE.END_C USTOMER_ CONTACT	

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
</CONTACT>				
</PARTNER>				
<POTERM>	Payment term information			
<DESCRIPTN>a</DESCRIPTN>	Description of payment terms.			PO_ECX_HEADER_ARCH _V.paymet_terms_descrip tion AP_TERMS.D ESCRIPTIN
<TERMID> Immediate </TERMID>	Identifier for payment terms.		OE_HEADERS_INTERF ACE. payment_term	PO_ECX_HEADER_ARCH _V.payme t_terms_name AP_TERMS.N AME
<DAYSNUM>a </DAYSNUM>				No mapping provided by PO although genrated XML has these elements.
</POTERM>				
</POHEADER>				
<POLINE>	This data type provides details of a PLine.At least one PO line data type is required.This data type will occur one or more times.			

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<OPERAMT qualifier="UNIT" type="T">	Unit price of the item		OE_LINES_I NTERFACE. customer_ite m_net_price	
<VALUE>a</VALUE>	Monetary unit amount of the PO line.			PO_ECX_LIN E_ARCH_V.p rice PO_LINES_A RCHIVE_ALL .UNIT_PRICE
<NUMOFDEC>a </NUMOFDEC>				
<SIGN>a</SIGN>				
<CURRENCY>a </CURRENCY>	Three-character ISO currency code.			PO_ECX_HE ADER_ARCH _V.po_currenc y PHA.CURRE NCY_CODE
<UOMVALUE>a</UO MVALUE>				
<UOMNUMDEC>a </UOMNUMDEC>				
<UOM>a</UOM>				
</OPERAMT>				
<QUANTITY qualifier="ORDERED >	Quantity of the item ordered. This will be the new qty if there is a change in the qty on schedule level.			

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<VALUE>a</VALUE>	Numeric Value of the qty			PO_ECX_LINE_ARCH_V.quantity PO_LINES_ARCHIVE_ ALL.QUANTITY
<NUMOFDEC>a</NUMOFDEC>				
<SIGN>a</SIGN>				
<UOM>a</UOM>				
</QUANTITY>				
<REASONCODE>abc</REASONCODE>	Reason Code. PO does not use this tag. If blank this should be defaulted as 'No Reason Provided'	N		
<POLINENUM>a</POLINENUM>	Po Line number	Y	OE_LINES_INTERFACE.customer_line_number	PO_ECX_LINE_ARCH_V.line_num PO_LINES_ARCHIVE_ALL.LINE_NUM
<DESCRIPTN>a</DESCRIPTN>	Item Description. No need to map for OM.			PO_ECX_LINE_ARCH_V.description PO_LINES_ARCHIVE_ALL.ITEM_DESCRIPTION

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<ITEMRV>a </ITEMRV>	Customer item revision			PO_ECX_LIN E_ARCH_V.it emrv PO_LINES_A RCHIVE_ALL .ITEM_REVISI ON
<ITEMRVX>a</ITEMR VX>	Not used			
<ITEM>a</ITEM>	Identifier of the product. This is the customer item number		OE_LINES_I NTERFACE. customer_ite m_name	PO_ECX_LIN E_ARCH_V.it em MTL_SYSTE M_ITEMS_B_ KfV.SEGME NT1
<ITEMX>a</ITEMX>	Supplier item number. This is the item number in OM			PO_ECX_LIN E_ARCH_V.it emx PO_LINES_A RCHIVE_ ALL.VENDO R_PRODUCT _NUM
<ITEMTYPE>a </ITEMTYPE>				
<NOTES index="1">a </NOTES>	Notes to supplier.		OE_LINES_I NTERFACE. change_com ments	PO_ECX_LIN E_ARCH_V.n ote_to_vendor PO_LINES_A RCHIVE_ALL .NOTE_TO_V ENDOR

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<USERAREA>				
<ACKCODE>	A value of 2 for a 3A8 Change PO response document implies a line level buyer rejection and will result in that line not being processed.			
<ORACLE.SUPPLIER_LINE_REF>1	Suppliers reference for the Customer PO Line.	N	OE_LINES_INTERFACE.	
</ORACLE.SUPPLIER_LINE_REF>	Needs to be sent back by the buyer as a reference for the newly created line on 3A8 response as well as 3A9. Also needed to be sent on every new change request for the new line.		orig_sys_line_ref	
</USERAREA>				
<SCHEDULE>				
<USERAREA>				
<PARTNER>	Bill to partner- This is an optional segment			
<NAME1>abcd</NAME1>	Name of the bill to partner			
<ONETIME>a</ONETIME>				
<PARTNRID>a</PARTNRID>				
<PARTNRTYPE>Bill to</PARTNRTYPE>				

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<CURRENCY>a</CURRENCY>				
<PARTNRIDX>a</PARTNRIDX>			OE_LINES_INTERFACE. invoice_to_org_id	
<ADDRESS>	The address information needs to be mapped for the case where EDI Location code is not sent.			
<ADDRLINE index="1">a</ADDRLINE>			OE_LINES_INTERFACE. INVOICE_T O_ADDRESS1-3	
<CITY>a</CITY>			OE_LINES_INTERFACE. INVOICE_T O_CITY	
<COUNTRY>a</COUNTRY>			OE_LINES_INTERFACE. INVOICE_T O_COUNTRY	
<COUNTY>a</COUNTY>			OE_LINES_INTERFACE. INVOICE_T O_COUNTY	
<POSTALCODE>a</POSTALCODE>			OE_LINES_INTERFACE. INVOICE_T O_POSTAL_CODE	

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<STATEPROVN>a </STATEPROVN>			OE_LINES_I NTERFACE. INVOICE_T O_STATE	
<TELEPHONE index="1">a</TELEPHONE>				
</ADDRESS>				
<PARTNER>	Ship to partner			
<NAME1>abcd</NAME1>	Name of the Ship to partner			PO_ECX_LIN E_LOC_ARC H_V.Shipto_o rg_name HR_ORGAN IZATION_UN ITS_V .NAME
<ONETIME>a </ONETIME>				
<PARTNRID>a </PARTNRID>				PO_ECX_LIN E_LOC_ARC H_V.partner_i d PO_LINE_LO CATIONS_A RCHIVE_ALL .ORG_ID
<PARTNRTYPE>Ship to</PARTNRTYPE>	Value ShipTo			PO_ECX_LIN E_LOC_ARC H_V.shipto_o rg_partner_ty pe 'ShipTo'
<CURRENCY>a</CURRENCY>				

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<PARTNRIDX>a </PARTNRIDX>	EDI Location code		OE_HEADERS_INTERFACE. ACE.ship_to _org_id	HR_LOCATIONS_ALL.ED E_TP_LOCATION_CODE
<ADDRESS>				
<ADDRLINE index="1 >a</ADDRLINE>	Address of the ship to		OE_HEADERS_INTERFACE. ACE.SHIP_TO_ADDRES S1-3	HR_LOCATIONS_ALL.ADD RESS_LINE_1,2,3
<CITY>a</CITY>	City		OE_HEADERS_INTERFACE. ACE.SHIP_TO_CITY	HR_LOCATIONS_ALL.TOW N_OR_CITY
<COUNTRY>a </COUNTRY>	Country		OE_HEADERS_INTERFACE. ACE.SHIP_TO_COUNTR Y	HR_LOCATIONS_ALL.COUN TRY
<COUNTY>a </COUNTY>			OE_LINES_INTERFACE. SHIP_TO_COUNTY	
<POSTALCODE>a </POSTALCODE>	Postal Code		OE_HEADERS_INTERFACE. ACE.SHIP_TO_POSTAL_ CODE	HR_LOCATIONS_ALL.PO STAL_CODE
<STATEPROVN>a </STATEPROVN>	State		OE_HEADERS_INTERFACE. ACE.SHIP_TO_STATE	HR_LOCATIONS_ALL.REGION_2

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<TELEPHONE index="1">a</TELEPHONE>	Telephone			HR_LOCATIONS_ALL.TELEPHONE_NUMBER 1,2,3
</ADDRESS>				
</PARTNER>				
<OPERATION>	Operation Code		OE_LINES_INTERFACE.operation_code	
<ACKCODE>	A value of 2 for a 3A8 Change PO Response document implies a line level buyer rejection and will result in that line not being processed.			
<ORACLE.SUPPLIER_SHIPMENT_REF>1	Supplier's reference for the customer PO shipment line. Needs to be sent back by the buyer as a reference for the newly created line on 3A8 response as well as 3A9. Also needed to be sent on every new change request for the new line.	N	OE_LINES_INTERFACE.orig_sys_shipment_ref	
</ORACLE.SUPPLIER_SHIPMENT_REF>				
</USERAREA>				

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<DATETIME qualifier="NEEDDELV">	Need by Delivery date. This will be the new date if there are any changes.		OE_LINES_I NTERFACE. request_date	PO_ECX_LIN E_LOC_ARC H_V.need_by _date PO_LINE_LO CATIONS_A RCHIVE_ALL .NEED_BY_D ATE
<YEAR>1998</YEAR>				
<MONTH>11 </MONTH>				
<DAY>01</DAY>				
<HOUR>00</HOUR>				
<MINUTE>00</MINU TE>				
<SECOND>00</SECO ND>				
<SUBSECOND>0000 </SUBSECOND>				
<TIMEZONE>-0600 </TIMEZONE>				
</DATETIME>				

OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<QUANTITY qualifier="ORDERED" >	Quantity of the item ordered. This will be the new qty.		OE_LINES_INTERFACE. ordered_quantity	PO_ECX_LINE_LOC_ARCHIVE_ALL. H_V.ordered_quantity
			OE_LINES_INTERFACE. order_quantity_uom	(PO_LINE_LOCATIONS_ARCHIVE_ALL. QUANTITY-PO_LINE_LOCATIONS_ARCHIVE_ALL. QUANTITY_CANCELED)
<VALUE>a</VALUE>				
<NUMOFDEC>a</NUMOFDEC>				
<SIGN>a</SIGN>				
<UOM>a</UOM>				
</QUANTITY>				
<PSCLINENUM>a</PSCLINENUM>	Line number of the delivery schedule for the PO.	Y	OE_LINES_INTERFACE. customer_shipment_number	PO_ECX_LINE_LOC_ARCHIVE_ALL. H_V.shipment_num PO_LINE_LOCATIONS_ARCHIVE_ALL. SHIPMENT_NUM
<DESCRIPTN>				

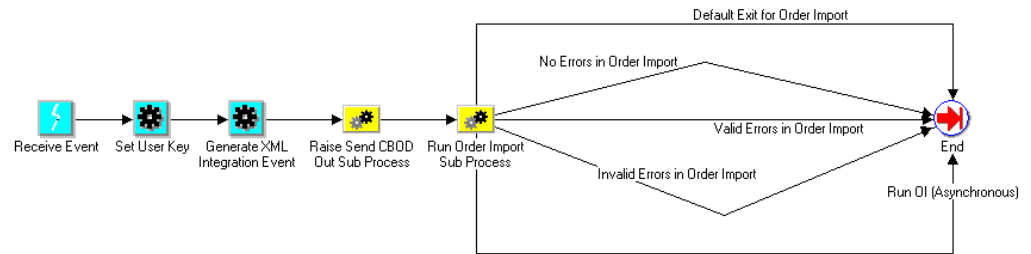
OAG	Comment/Desc	Reqd Y/N	Oracle OM Table/Column	Oracle PO Table /Column
<REASONCODE>	Reason for cancellation		OE_LINES_INTERFACE. change_reason	
</SCHEDULE>				
</POLINE>				
</CHANGE_PO>				
</DATAAREA>				
</CHANGE_PO_006>				

Note: Note: When the columns customer_item_net_price and customer_payment_term have values, the system calculates a different value, and you will get a warning. You can also see those column's values in the Sales Orders form.

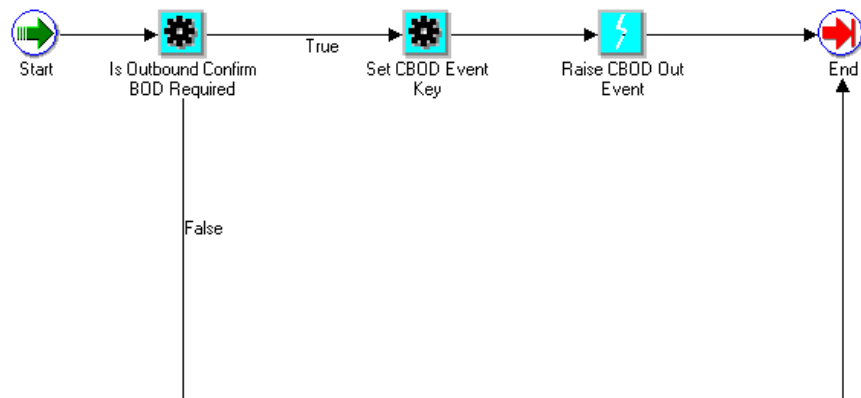
Seeded Workflow for Inbound XML Messages

A pre-defined Workflow event is seeded to run Order Import immediately if the Profile Option OM: Run Order Import for XML is set to 'Synchronous'. This Workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the workflow, Order Import is run for the XML message, that is currently being consumed. The following Workflow is designed using Workflow Builder 2.6

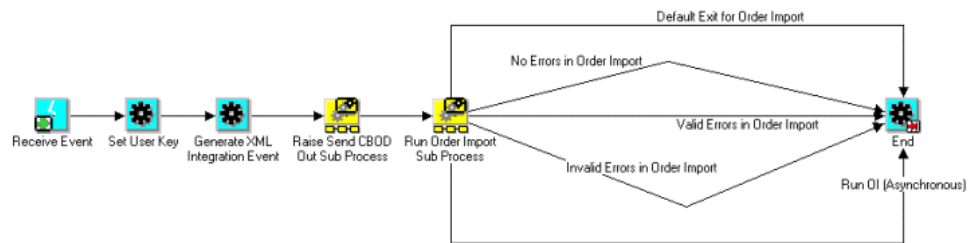
Order Import Flow - Generic



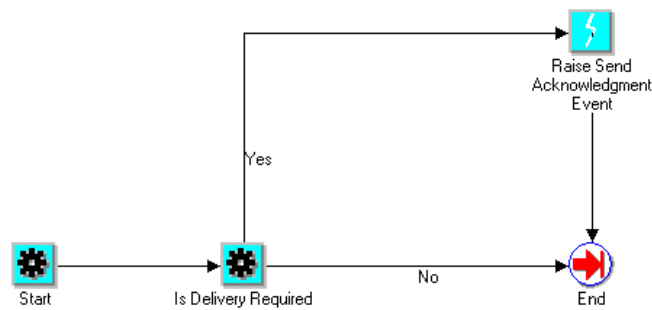
Raise Send CBOD Out Sub Process



Run Order Import Sub Process



Raise Send Acknowledgement Sub Process



Message Details

This table describes the data types and fields in the DTD used by the message map. The message map may remove used fields so that empty data tags are neither generated for outbound transactions nor examined by inbound transactions.

ONT_3A8R_OAG72_IN.xgm

Message Map Name:	ONT_3A8R_OAG72_IN.xgm
Direction:	Inbound
(Internal) Transaction Type:	ONT

(Internal) Transaction Subtype:	CHO
External Transaction Type:	PO
External Transaction Subtype:	CHANGE
DTD Directory:	\$ont/xml/oag72
Map Directory:	\$ont/patch/115/xml/US
Message Maps XGM File Name:	ONT_3A8R_OAG72_IN
Standard:	OAG
Release:	7.2
Format:	DTD
DTD Name:	057_change_po_006.dtd

Columns Enabled for Code Conversion

Tag	Category
<TERMID> (in header level <POTERM>	PAYMENT_TERMS
<MSGTYPE> (in header level <USERAREA>)	DOC_PURPOSE_CODE
<UOM> (in schedule level <QUANTITY qualifier="ORDERED">)	UOM
<OPERATION> (in schedule level USERAREA)	ACTION_CODE

Defaulted Columns

Defaulted Columns	Default Value, and Condition (if any)
H.ORDER_SOURCE_ID	20
H.XML_TRANSACTION_TYPE_CODE	'CHO'
H.CHANGE_REASON	'Not provided'
L.ORDER_SOURCE_ID	20
L.ORIG_SYS_DOCUMENT_REF	H.ORIG_SYS_DOCUMENT_REF
L.XML_TRANSACTION_TYPE_CODE	'CHO'
L.CHANGE_REASON	'Not provided'

Derived Columns

Derived Columns	Source of Data, and Condition (if any)
H.CREATED_BY	FND_GLOBAL.USER_ID
H.CREATION_DATE	SYSDATE
H.INVOICE_TO_ORG_ID	PARTNRIDX of BillTo PARTNER
H.LAST_UPDATED_BY	FND_GLOBAL.LOGIN_ID
H.LAST_UPDATE_DATE	SYSDATE
H.LAST_UPDATE_LOGIN	FND_GLOBAL.USER_ID
H.ORG_ID	PARTNRIDX of SoldTo PARTNER, if this value is not null
H.SOLD_TO_ORG_ID	PARTNRIDX of SoldTo PARTNER
H.SHIP_TO_ORG_ID	PARTNRIDX of ShipTo PARTNER

Derived Columns	Source of Data, and Condition (if any)
H.XML_MESSAGE_ID	XML Gateway Internal Control Number
L.CALCULATE_PRICE_FLAG	'Y' if L.CUSTOMER_ITEM_NET_PRICE is NOT NULL
L.CHANGE_SEQUENCE	H.CHANGE_SEQUENCE
L.CREATED_BY	FND_GLOBAL.USER_ID
L.CREATION_DATE	SYSDATE
L.INVOICE_TO_ORG_ID	PARTNRIDX of BillTo PARTNER
L.LAST_UPDATED_BY	FND_GLOBAL.LOGIN_ID
L.LAST_UPDATE_DATE	SYSDATE
L.LAST_UPDATE_LOGIN	FND_GLOBAL.LOGIN_ID
L.ORDERED_QUANTITY, L.ORDER_QUANTITY_UOM	OAG Derivation from schedule level <QUANTITY qualifier="ORDERED"> tag
L.OPERATION_CODE	Derived based on stored procedure if <OPERATION> tag at schedule level is NULL
L.REQUEST_DATE	OAG Derivation from schedule level <DATETIME qualifier="NEEDELV"> tag
L.SHIP_TO_ORG_ID	PARTNRIDX of ShipTo PARTNER

Workflow Event Setup

Detail	Value
Event Name	oracle.apps.ont.oi.po_inbound.create
Event Description	OM Generic Inbound Event
Subscription	R_OEOI_ORDER_IMPORT

Detail	Value
Subscription Description	Oracle Order Management subscription to the event oracle.apps.ont.oi.po_inbound.create raised by the post process of the Change PO XML mapping

Extension Tags

Order Management added the following tags as extensions on the Process PO XML documents sent to OM:

```
$ont/xml/oag72/oagis_extensions.dtd
```

```
<!ELEMENT FTTERM (DESCRIPTN?, TERMID?)>
```

```
<!ELEMENT FOB (DESCRIPTN?, TERMID?)>
```

```
<!ELEMENT OPERATION %STRDOM;>
```

```
<!ELEMENT ORGID %STRDOM;>
```

```
<!ELEMENT USERITEMDESCRIPTN %STRDOM;>
```

```
<!ELEMENT ORACLE.CHANGESEQUENCE %STRDOM;>
```

```
<!ELEMENT ORACLE.BOOKEDFLAG %STRDOM;>
```

```
<!ELEMENT ORACLE.SUPPLIERDOCREF %STRDOM;>
```

```
<!ELEMENT ORACLE.SUPPLIERLINEREF %STRDOM;>
```

```
<!ELEMENT ORACLE.SUPPLIERSHIPMENTREF %STRDOM;>
```

```
<!ELEMENT ORACLE.SPLITFROMLINEREF %STRDOM;>
```

```
<!ELEMENT ORACLE.SPLITLINE %STRDOM;>
```

```
$ont/xml/oag72/oagis_entity_extensions.dtd
```

Added the following tags:

```
<!ENTITY % DATETIME.EXCHRATE DATE "DATETIME">
```

Change <!ENTITY % SEG_DATETIME_QUALIFIERS_EXTENSION "OTHER"> to

```
<!ENTITY % SEG_DATETIME_QUALIFIERS_EXTENSION "OTHER |
EXCHRATE DATE">
```

```
$ont/xml/oag72/oagis_extensions.dtd
```

Added the following tags:

```
<!ELEMENT PAYMMETHOD (DESCRIPTN?, TERMID?)>
```

```
<!ELEMENT CREDITCRD (CARDID?, NAME?, (%DATETIME.EXPIRATION;?)>
```



```

<ELEMENT STARTACTIVEDATE %STRDOM;>
<ELEMENT ENDACTIVEDATE %STRDOM;>
<ELEMENT CATEGORYID %STRDOM;>
<ELEMENT REVISIONNUM %STRDOM;>
<ELEMENT ATTACHMENT (TEXT?)>
<ELEMENT EXCHRATE %STRDOM;>
<ELEMENT CONFIRM %STRDOM;>
<ELEMENT PCARDHDR
(MEMBERNAME?,PCARDNUM?,(%DATETIME.EXPIRATION;)?,PCARDBRAND?)><!
ELEMENT MEMBERNAME %STRDOM;>
<ELEMENT PCARDNUM %STRDOM;>
<ELEMENT PCARDBRAND %STRDOM;>
<ELEMENT CUSTOMERNUM %STRDOM;>
<ELEMENT REQUESTOR %STRDOM;>
<ELEMENT CONTRACTPONUM %STRDOM;>
<ELEMENT CONTRACTPOLINENUM %STRDOM;>
<ELEMENT VENDORQUOTENUM %STRDOM;>
<ELEMENT LISTPRICE %STRDOM;>
<ELEMENT MARKETPRICE %STRDOM;>
<ELEMENT PRICENOTTOEXCEED %STRDOM;>
<ELEMENT NEGPRICE %STRDOM;>
<ELEMENT TAXABLE %STRDOM;>
<ELEMENT TXNREASONCODE %STRDOM;>
<ELEMENT TYPE1099 %STRDOM;>
<ELEMENT LINEORDERTYPE %STRDOM;>
<ELEMENT HAZRDUNNUM %STRDOM;>
<ELEMENT HAZRDUNDESC %STRDOM;>
<ELEMENT PRICEOVRD %STRDOM;>
<ELEMENT DISTPROJECT
(REQUESTOR?,DISTNUM?,PROJECTNUM?,PROJECTTYPE?,TASKNUM?,(%QUANTI
TY.ORDERED;)?,CONVRATE,(%DATETIME.EXCHRATE;)?,DESTTYPE?,DFFDI
STRIBUTN?)>
<ELEMENT PROJECTNUM %STRDOM;>
<ELEMENT DISTNUM %STRDOM;>

```

```

<ELEMENT PROJECTTYPE %STRDOM;>

<ELEMENT TASKNUM %STRDOM;>

<ELEMENT CONVRATE %STRDOM;>

<ELEMENT DESTTYPE %STRDOM;>

<ELEMENT
DFFPOHEADER(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIB
UTE5?,ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,A
TTTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRI
BUTE16?)><ELEMENT
DFFVENDORSITE(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRI
BUTE5?,ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,
ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATT
RIBUTE16?)>

<ELEMENT
DFFVENDOR(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUT
E5?,ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATT
RIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBU
TE16?)>

<ELEMENT
DFFLINE(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,
ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBU
TE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE16
?)>

<ELEMENT
DFFITEM(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,
ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBU
TE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,A
TTTRIBUTE16?)>

<ELEMENT
KFFITEM(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,
ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBU
TE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE16
?,ATTRIBUTE17?,ATTRIBUTE18?,ATTRIBUTE19?,ATTRIBUTE20?)>

<ELEMENT
DFFDISTRIBUTN(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRI
BUTE5?,ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?
,ATTRIBUTE10?,ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATT
RIBUTE15?,ATTRIBUTE16?)>

<ELEMENT ATTRIBUTE1 %STRDOM;>

<ELEMENT ATTRIBUTE2 %STRDOM;>

```

```

<ELEMENT ATTRIBUTE3 %STRDOM;>
<ELEMENT ATTRIBUTE4 %STRDOM;>
<ELEMENT ATTRIBUTE5 %STRDOM;>
<ELEMENT ATTRIBUTE6 %STRDOM;>
<ELEMENT ATTRIBUTE7 %STRDOM;>
<ELEMENT ATTRIBUTE8 %STRDOM;>
<ELEMENT ATTRIBUTE9 %STRDOM;>
<ELEMENT ATTRIBUTE10 %STRDOM;>
<ELEMENT ATTRIBUTE11 %STRDOM;>
<ELEMENT ATTRIBUTE12 %STRDOM;>
<ELEMENT ATTRIBUTE13 %STRDOM;>
<ELEMENT ATTRIBUTE14 %STRDOM;>
<ELEMENT ATTRIBUTE15 %STRDOM;>
<ELEMENT ATTRIBUTE16 %STRDOM;>
<ELEMENT ATTRIBUTE17 %STRDOM;>
<ELEMENT ATTRIBUTE18 %STRDOM;>
<ELEMENT ATTRIBUTE19 %STRDOM;>
<ELEMENT ATTRIBUTE20 %STRDOM;>
<ELEMENT TANDC %STRDOM;>
<ELEMENT GLOBALCONTRACT %STRDOM;>
<ELEMENT GLOBALCONTRACTLIN %STRDOM;>
<ELEMENT CONSIGNEDINV %STRDOM;>
<ELEMENT DROPSHIPDETAILS (DROPSHIPMENT?, DROPSHIPCUSTNAME?,
SHIPINSTR?, PACKINSTR?, SHIPMETHOD?, CUSTOMERPONUM?,
CUSTOMERLINENUM?, CUSTOMERSHIPNUM?, CUSTOMERDESC?)>
<ELEMENT DROPSHIPMENT %STRDOM;>
<ELEMENT DROPSHIPCUSTNAME %STRDOM;>
<ELEMENT SHIPINSTR %STRDOM;>
<ELEMENT PACKINSTR %STRDOM;>
<ELEMENT SHIPMETHOD %STRDOM;>
<ELEMENT CUSTOMERPONUM %STRDOM;>
<ELEMENT CUSTOMERLINENUM %STRDOM;>
<ELEMENT CUSTOMERSHIPNUM %STRDOM;>

```

<ELEMENT CUSTOMERDESC %STRDOM;>
<ELEMENT SHIPPINGCONTROL %STRDOM;>
<ELEMENT CONTRACTNUM %STRDOM;>
<ELEMENT CONFIGID %STRDOM;>
<ELEMENT PSCLNSTATUS %STRDOM;>

Cancel Purchase Order

Overview

The Cancel Purchase Order message is used by the buyer to cancel a purchase order that was previously created by the buyer. This is an inbound message from buyer to the seller. Cancellation of the individual lines is allowed using this message. Cancellation of the PO follows same rules for canceling a purchase order that currently exist in Order Management. Partial cancellations of the line is allowed.

An Acknowledgement is sent to the buyer when the Cancel_PO message is consumed by Order Management. Either an Accept or Reject status for the Cancel_PO request by the buyer will be sent. The Acknowledge PO message is used for the acknowledgement for the inbound Cancel_PO message.

Major Features

Consume the Cancel_PO Message Coming Inbound From Purchasing

Oracle Order Management consumes the Cancel_PO message inbound from Purchasing, thereby canceling the Purchase order previously entered. The whole purchase order is cancelled due to this inbound message. Cancellation of the individual lines is allowed.

Cancellation of the PO should follow same rules for canceling a purchase order that currently exist in Order Management. Partial cancellations of the line are allowed.

For example – If the order Qty is 10 on the original order and Cancel_PO comes in with the order Qty of 6. After processing the transaction, the Order Qty of the order will become 6 thereby cancelling 4 out of original Qty.

See Table, "Message Map Details" for the details and mapping of this message.

Acknowledge Cancellation of Purchase Order

You can confirm the deletion of the purchase order back to the buyer. Confirmation back to the buyer is done via an Acknowledge_PO message outbound from Order Management and will indicate that the purchase order has been canceled.

If the cancellation request cannot be fulfilled, in other words if the request falls outside of the allowable rules for cancellation, then an Acknowledge_PO message outbound

from Order Management must be sent indicating that the purchase order deletion cannot be completed at this time and has been rejected.

Generate Confirm BOD (CBOD)

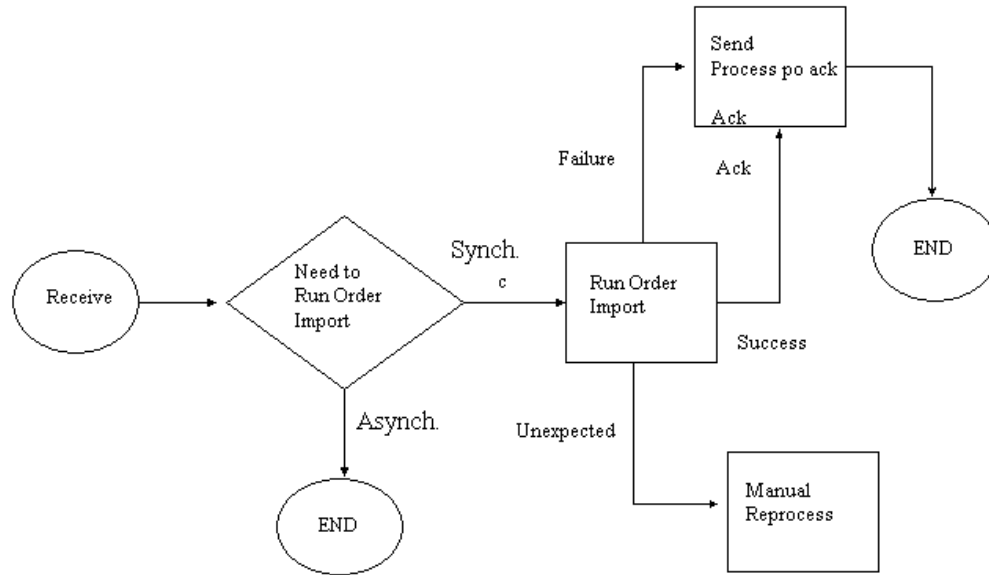
Upon receipt of the inbound Cancel_PO, Order Management can send out the Confirm BOD to the buyer indicating that the data has reached the seller. The confirm bod will not be sent out every time. It will be sent depending on the value of the <CONFIRMATION> tag in the inbound document. The CBOD will not be sent if the value of that tag is 0 or 1. It will be sent if the value of that tag is 2. Confirm BOD should be generated on the successful insertion of the data in open interface tables. It can be part of the workflow that begins after XML Gateway has finished the insertion of the data successfully. At this point only the 'Successful' status will be sent back on the CBOD as its unlikely that failure will occur after Order Management receives the control of the process. Any failure before this point will be trapped by XML gateway and notification will be sent to the buyer. Confirm BOD is only generated if the inbound document has required confirmation.

Set up the Order Import Process to Run in Synchronous or Asynchronous Mode

You can set the profile option to initiate Order Import for every order or wait and process multiple orders via concurrent processing. The profile option is OM: Run Order Import for XML, that is set at the user or site level. The possible values for this profile option will be Synchronous or Asynchronous.

Error Handling

See figure on the Error Handling Process.



The error handling process deals with the three kinds of outcomes that occur as a result of running order import.

- **Successful processing** - In this case the confirmation API will be called to send the Acknowledgment out to the sender with the 'Accept' status.
- **Failed processing** - This will be the status when there are validation errors or data setup errors. In this case the acknowledgment will be sent out to the sender with the 'Reject' status.
- **Unexpected errors** - There can be some unexpected errors due to resource constraints and such which will require reprocessing of the Order Import upon fixing those issues. This process is manual.

Business and Process Flow

1. The CANCEL_PO OAG XML Message is placed on an AQ present in the E-Business Suite Database.
2. The CANCEL_PO Message is consumed using XML Gateway.
3. The XML Gateway will raise the appropriate event.
4. A CBOD is generated based on the Confirm flag in the message or the trading partner setup. See: Generate Confirm BOD (CBOD)
5. The seeded Order Management Workflow will receive this event and start the Cancel Flow.

6. Order import is used to process the message. The Order Management Module accepts the message based on set rules and tolerances previously agreed upon between the trading partners. The customer order is Cancelled.
7. The Order Management Inbound Workflow will then raise an event that will be subscribed by the Outbound Workflow, and in turn will generate the data for the OAG Acknowledge PO.
8. The XML Gateway forms the ACKNOWLEDGE_PO message.
9. This message is placed on an AQ present on the e-Business Suite Database.

CONFIRM BOD

Upon receipt of an inbound XML document such as a Process_PO or Cancel_PO, Order Management has implemented the outbound OAG Confirm BOD message to signal the successful receipt of data.

Major features of this transaction are as follows:

- This message contains information about the inbound XML message received, as well as the status and description of the inbound message received.
- Unlike the Acknowledge PO message, the Confirm BOD is sent before Order Import is run – it is purely a confirmation.

Message Map

Most data values in the message map are seeded.

Message Map 'ONT_3A9R_OAG72_IN.xgm'

The message map 'ONT_3A9R_OAG72_IN' is created using the Oracle XML Gateway Message Designer tool. Please refer to Table, "Message Map Details" for the detailed map analysis.

The source DTD used is 058_cancel_po_006.dtd, revision 7.2.1 of the Open Application Group. The other associated external reference DTD files are:

- oagis_domains.dtd
- oagis_resources.dtd
- oagis_fields.dtd
- oagis_segments.dtd
- oagis_extensions.dtd
- oagis_entity_extensions.dtd

All the DTD's will be checked in ONT source area under \$ont/xml/oag72.

The target for the Inbound XML Message are the Order Management Open Interface tables OE_HEADERS_INTERFACE & OE_LINES_INTERFACE.

The CANCEL_PO DTD is a three level hierarchy with Order Lines split into one or more Shipment Lines. The Order Management architecture is however a two level one with Order Header and one or more Lines. The message map will collapse the three level XML Message into a two level structure when the data is inserted in the Order Management Open Interface tables.

Please refer to Table, "Message Map Details" for a detail analysis of the elements mapped, actions and derivation rules used by the Message Map.

Both the message map created using the Message Designer and its associated DTD's are stored in the database. The following Java programs are available to Load/Delete Maps or Load/Delete DTD's into/from the XML Gateway repository. Please refer to the *Oracle XML Gateway Manual* for more information.

Load/Delete Maps, Load/Delete DTD's

The following process is used only for customizations.

- java LoadMap DB username> DB password> Hostname>:Port>:SID> mymap.xgm>
- Example: java oracle.apps.ecx.loader.LoadMap apps apps ap505dbs:1521:dev115 ONT_3A9R_OAG72_IN.xgm
- java DeleteMap DB username> DB password> Hostname>:Port>:SID> mapname>
Example: java oracle.apps.ecx.loader.DeleteMap apps apps ap505dbs:1521:dev115 ONT_3A9R_OAG72_IN.xgm

The Message Map is a .xgm file which will be stored in the Order Management Source area under \$ont/patch/115/map/ ONT_3A9R_OAG72_IN.xgm

For details on how the Message Map will be delivered to Customers please refer to the Table, "Business and Process Flow" Section.

Note: Maps and DTD's must be kept in sync between the Message Designer and the XML Gateway repository. When in doubt, always reload the map and DTD as a pair.

key: H => OE_HEADERS_INTERFACE, L => OE_LINES_INTERFACE

Cancel_PO Message Map

OAG Element	Description or Comment	Oracle OM Table/Column Target: OE_HEADERS _INTERFACE (H) OE_LINES_IN TERFACE (L)	OAG Reqd Yes or No (Y/N)	Code Convers ion Needed Y/N	Default Value
<hr/>					
<CANCEL_PO_006 >					
<CNTROLAREA>					
<BSR>					
<VERB>CANCEL</ VERB>					
<NOUN>PO</NO UN>					
<REVISION>006</ REVISION>					
</BSR>					
<SENDER>					
<LOGICALID>XX1 41HG09</LOGICA LID>					
<COMPONENT>P UR</COMPONEN T>					
<TASK>MAINT</T ASK>					
<hr/>					

OAG Element	Description or Comment	Oracle OM Table/Column	OAG Reqd	Code Convers ion Needed Y/N	Default Value
		Target: OE_HEADERS _INTERFACE (H)	Yes or No (Y/N)		
		OE_LINES_IN TERFACE (L)			
	<REFERENCEID>9 5129945823449</RE FERENCEID>				
	<CONFIRMATION >1</CONFIRMATI ON>				
	<LANGUAGE>EN G</LANGUAGE>				
	<CODEPAGE>test< /CODEPAGE>				
	<AUTHID>CMKU RT</AUTHID>				
	</SENDER>				
	<DATETIME qualifier="CREATI ON">				
	<YEAR>1998</YEA R>				
	<MONTH>11</MO NTH>				
	<DAY>21</DAY>				
	<HOUR>16</HOU R>				

OAG Element	Description or Comment	Oracle OM Table/Column Target: OE_HEADERS _INTERFACE (H) OE_LINES_IN TERFACE (L)	OAG Reqd Yes or No (Y/N)	Code Convers ion Needed Y/N	Default Value
<MINUTE>46</MI NUTE>					
<SECOND>45</SE COND>					
<SUBSECOND>000 0</SUBSECOND>					
</DATETIME>					
</CNTROLAREA>					
<DATAAREA>					
<CANCEL_PO>					
Source CANCEL_PO					
<POHEADER>					
POID		H.CUSTOMER _PO_NUMBER	Y		
SITELEVEL			Y		
DESCRIPTN		H.CHANGE_R EASON			Not provided.
NOTES index="1"		H.CHANGE_C OMMENTS			
<USERAREA>					

OAG Element	Description or Comment	Oracle OM Table/Column Target: OE_HEADERS _INTERFACE (H) OE_LINES_IN TERFACE (L)	OAG Reqd Yes or No (Y/N)	Code Convers ion Needed Y/N	Default Value
< PARTNER>					
PARNTRTYPE (SoldTo)					
PARTNRIDX	Derive Address Id from Location Code Execute Procedure Call	H.SOLD_TO_ ORG_ID	Y		
</PARTNER>					
<ORACLE.CHAN GESEQUENCE>		H.CHANGE_S EQUENCE			
<ORACLE.SUPPLI ERDOCREF>		H.ORIG_SYS_ DOCUMENT_ REF			
</USERAREA>					
	Execute Function Call.	H.CREATION _DATE	Y		SYSDATE
	Execute Function Call.	H.LAST_UPD ATED_BY	Y		FND_GLOBA L.LOGIN_ID
	Execute Function Call.	H.LAST_UPD ATE_DATE	Y		SYSDATE
	Execute Function Call.	H.CREATED_ BY	Y		FND_GLOBA L.USER_ID

OAG Element	Description or Comment	Oracle OM Table/Column Target: OE_HEADERS _INTERFACE (H) OE_LINES_IN TERFACE (L)	OAG Reqd Yes or No (Y/N)	Code Convers ion Needed Y/N	Default Value
	Get Pre-Defined Variable Value.	H.ORDER_SO URCE_ID	Y		20
	Derive Org Id from (SoldTo) Location Code.	H.ORG_ID	Y		
		H.XML_TRAN SACTION_TY PE_CODE	Y		CPO
</POHEADER>					
<POLINE>					
POLINENUM		L.CUSTOMER _LINE_NUMB ER	Y		
NOTES index="1"		L.CHANGE_C OMMENTS			
<USERAREA>					
<ORACLE.SUPPLI ERLINEREF>		L.ORIG_SYS_L INE_REF			
</USERAREA>					
<SCHEDULE>					

OAG Element	Description or Comment	Oracle OM Table/Column Target: OE_HEADERS _INTERFACE (H) OE_LINES_IN TERFACE (L)	OAG Reqd Yes or No (Y/N)	Code Convers ion Needed Y/N	Default Value
<QUANTITY qualifier="ORDERE D">	Convert from OAG using standard function.	L.ORDERED_ QUANTITY			
VALUE	Convert from OAG using standard function.	L.ORDERED_ QUANTITY			0
NUMOFDEC	Convert from OAG using standard function.	L.ORDERED_ QUANTITY			
SIGN	Convert from OAG using standard function.	L.ORDERED_ QUANTITY			
UOM ¹	Convert from OAG using standard function.	L.ORDER_QU ANTITY_UOM			
</QUANTITY>					
PSCLINENUM		L.CUSTOMER _SHIPMENT_ NUMBER	Y		
DESCRIPTN		L.CHANGE_R EASON			Not provided.
	Execute Function Call.	L.CREATION_ DATE	Y		SYSDATE

OAG Element	Description or Comment	Oracle OM Table/Column Target: OE_HEADERS _INTERFACE (H) OE_LINES_IN TERFACE (L)	OAG Reqd Yes or No (Y/N)	Code Convers ion Needed Y/N	Default Value
	Execute Function Call.	L.LAST_UPDA TED_BY	Y		FND_GLOBA L.LOGIN_ID
	Execute Function Call.	L.LAST_UPDA TE_DATE	Y		SYSDATE
	Execute Function Call.	L.CREATED_B Y	Y		FND_GLOBA L.USER_ID
	Get Pre-Defined Variable Value.	L.ORDER_SO URCE_ID	Y		20
		L.ORG_ID	Y		
		L.XML_TRAN SACTION_TY PE_CODE	Y		CPO
<USERAREA>					
<ORACLE.SUPPLI ERSHIPMENTREF >		L.ORIG_SYS_S HIPMENT_RE F			
</USERAREA>					
</SCHEDULE>					
</POLINE>					
</CANCEL_PO>					
</DATAAREA>					

OAG Element	Description or Comment	Oracle OM Table/Column	OAG Reqd	Code Conversion Needed	Default Value
		Target: OE_HEADERS _INTERFACE (H)	Yes or No (Y/N)	Y/N	
		OE_LINES_IN TERFACE (L)			
</CANCEL_PO_00 6>					

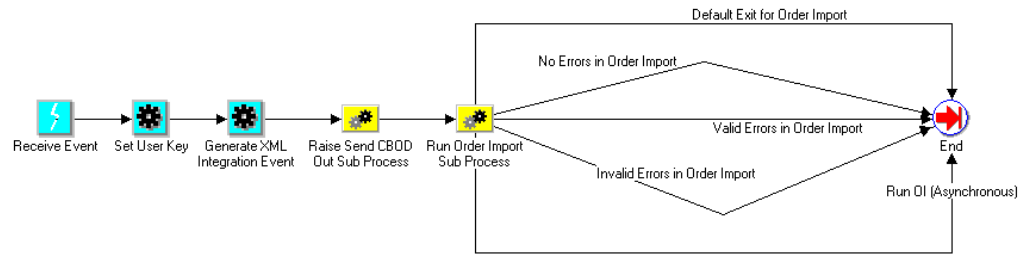
Note: ¹UOM: The code category for this source is UOM.

Cancel Reason Code: The cancel reason code is not mandatory for OAG while it is mandatory for Order Management's column. If the reason code is not sent by customer Order Management defaults it to No Reason Provided. This is seeded in Cancel_code lookup codes (OE_LOOKUP) as part of installation on Cancel_PO. The reason codes populated by the customer must be valid lookup codes in Cancel_Code lookup code.

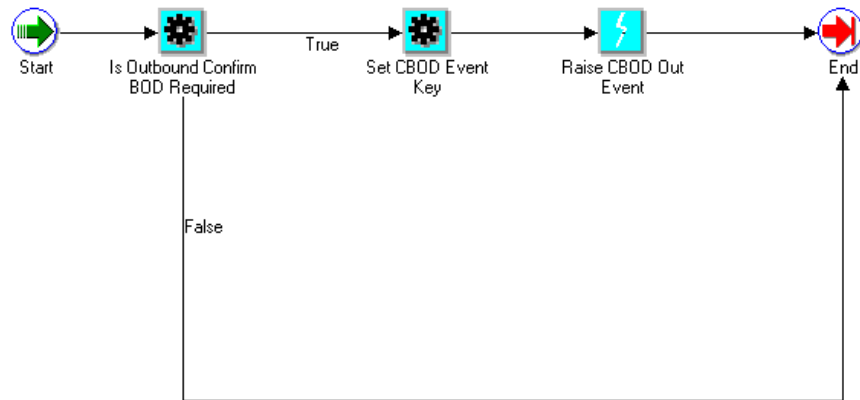
Seeded Workflow for Inbound XML Messages

A pre-defined workflow event is seeded to run Order Import immediately if the profile option OM: Run Order Import for XML is set to Synchronous. This workflow subscribes to the event raised by the Post Process of the XML Gateway. As part of the workflow process Order Import is run for the XML message, that is currently being consumed. The following Workflow is designed using Workflow Builder.

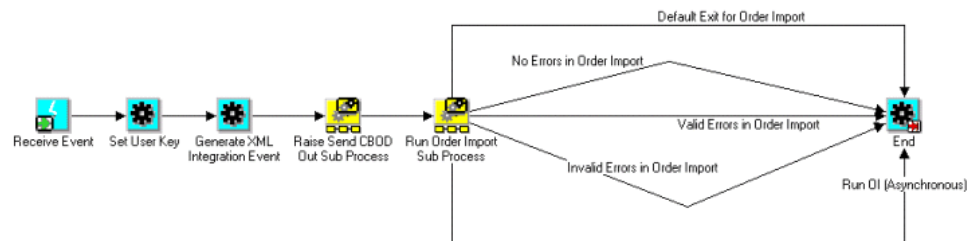
Order Import Flow - Generic



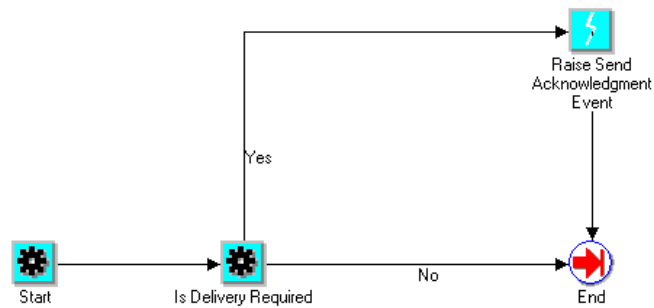
Raise Send CBOD Out Sub Process



Run Order Import Sub Process



Raise Send Acknowledgement Sub Process



Message Details

This table describes the data types and fields in the DTD used by the message map.

The message map may remove used fields so that empty data tags are neither generated for outbound transactions nor examined by inbound transactions.

Message Map Details

Name	Description
Message Map Name:	ONT_3A9R_OAG72_IN
Direction:	Inbound
(Internal) Transaction Type:	ONT
(Internal) Transaction Subtype:	CPO
External Transaction Type:	PO
External Transaction Subtype:	CANCEL
DTD Directory:	xml/oag72
Map Directory:	patch/115/xml/US
Message Maps XGM File Name:	ONT_3A9R_OAG72_IN.xgm

Name	Description
Standard:	OAG
Release:	7.2
Format:	DTD
DTD Name:	058_cancel_po_006.dtd

For the next three tables, refer to Table, "Message Map Details".

Columns Enabled for Code Conversion

Column	Description
<UOM> (in schedule level <QUANTITY qualifier="ORDERED">)	Code Category – UOM, mapped to OE_LINES_INTERFACE.ORDER_QUANTIT Y_UOM

Key: H => OE_HEADER_INTERFACE, L => OE_LINES_INTERFACE

Defaulted Columns

Defaulted Columns	Default Value, and Condition (if any)
H.CHANGE_REASON	'Not provided' (if Header-level DESCRIPTN tag is NULL>
H.OPERATION_CODE	UPDATE'
H.ORDER_SOURCE_ID	20
H.XML_TRANSACTION_TYPE_CODE	'CPO'
L.CHANGE_REASON	'Not provided' (if Line-level DESCRIPTN tag is NULL>
L.OPERATION_CODE	'UPDATE'

Defaulted Columns	Default Value, and Condition (if any)
L.ORDERED_QUANTITY	0 (if derived ORDERED_QUANTITY is NULL)
L.ORDER_SOURCE_ID	20
L.ORIG_SYS_DOCUMENT_REF	H.ORIG_SYS_DOCUMENT_REF
L.XML_TRANSACTION_TYPE_CODE	'CPO'

Derived Columns

Derived Columns	Source of Data, and Condition (if any)
H.CANCELLED_FLAG	Set to 'Y', if no lines are passed
H.CREATED_BY	FND_GLOBAL.LOGIN_ID
H.CREATION_DATE	SYSDATE
H.LAST_UPDATED_BY	FND_GLOBAL.LOGIN_ID
H.LAST_UPDATE_DATE	SYSDATE
H.ORG_ID	PARTNRIDX of SoldTo PARTNER, if this value is not null
H.SOLD_TO_ORG_ID	PARTNRIDX of SoldTo PARTNER
H.XML_MESSAGE_ID	XML Gateway Internal Control Number
L.CREATED_BY	FND_GLOBAL.LOGIN_ID
L.CREATION_DATE	SYSDATE
L.LAST_UPDATED_BY	FND_GLOBAL.LOGIN_ID
L.LAST_UPDATE_DATE	SYSDATE

Derived Columns	Source of Data, and Condition (if any)
L.ORDERED_QUANTITY, L.ORDER_QUANTITY_UOM	OAG Derivation from schedule level <QUANTITY qualifier="ORDERED"> tag

Workflow Event Setup

Detail	Value
Event Name	oracle.apps.ont.oi.po_inbound.create
Event Description	Order Management Generic Inbound Event
Subscription	R_OEOI_ORDER_IMPORT
Subscription Description	Oracle Order Management subscription to the event oracle.apps.ont.oi.po_inbound.create raised by the post process of the Process_PO XML mapping

Extension Tags

The following changes were made by Procurement as extensions on the XML documents sent to Order Management.

- \$ont/xml/oag72/oagis_entity_extensions.dtd

Add the following tags:

```
<!ENTITY % DATETIME.EXCHRATE DATE "DATETIME">
```

Change <!ENTITY % SEG_DATETIME_QUALIFIERS_EXTENSION "OTHER"> to
<!ENTITY % SEG_DATETIME_QUALIFIERS_EXTENSION "OTHER |
EXCHRATE DATE">

\$ont/xml/oag72/oagis_extensions.dtd

Add the following tags:

```
<!ELEMENT PAYMMETHOD (DESCRIPTN?, TERMID?)>
```

```
<!ELEMENT CREDITCRD (CARDID?, NAME?, (%DATETIME.EXPIRATION;?)>
```

```
<!ELEMENT STARTACTIVEDATE %STRDOM;>
```

```
<!ELEMENT ENDACTIVEDATE %STRDOM;>
```

<!ELEMENT CATEGORYID %STRDOM;>
 <!ELEMENT REVISIONNUM %STRDOM;>
 <!ELEMENT ATTACHMENT (TEXT?)>
 <!ELEMENT EXCHRATE %STRDOM;>
 <!ELEMENT CONFIRM %STRDOM;>
 <!ELEMENT PCARDHDR
 (MEMBERNAME?,PCARDNUM?,(%DATETIME.EXPIRATION;)?,PCARDBRAND
 ?)>
 <!ELEMENT MEMBERNAME %STRDOM;>
 <!ELEMENT PCARDNUM %STRDOM;>
 <!ELEMENT PCARDBRAND %STRDOM;>
 <!ELEMENT CUSTOMERNUM %STRDOM;>
 <!ELEMENT REQUESTOR %STRDOM;>
 <!ELEMENT CONTRACTPONUM %STRDOM;>
 <!ELEMENT CONTRACTPOLINENUM %STRDOM;>
 <!ELEMENT VENDORQUOTENUM %STRDOM;>
 <!ELEMENT LISTPRICE %STRDOM;>
 <!ELEMENT MARKETPRICE %STRDOM;>
 <!ELEMENT PRICENOTTOEXCEED %STRDOM;>
 <!ELEMENT NEGPRICE %STRDOM;>
 <!ELEMENT TAXABLE %STRDOM;>
 <!ELEMENT TXNREASONCODE %STRDOM;>
 <!ELEMENT TYPE1099 %STRDOM;>
 <!ELEMENT LINEORDERTYPE %STRDOM;>
 <!ELEMENT HAZRDUNNUM %STRDOM;>
 <!ELEMENT HAZRDUNDESC %STRDOM;>
 <!ELEMENT PRICEOVRRD %STRDOM;>
 <!ELEMENT DISTPROJECT
 (REQUESTOR?,DISTNUM?,PROJECTNUM?,PROJECTTYPE?,TASKNUM?,(%Q
 UANTITY.ORDERED;)?,CONVRATE,(%DATETIME.EXCHRATEDATE;)?,DESTTY
 PE?,DFFDISTRIBUTN?)>
 <!ELEMENT PROJECTNUM %STRDOM;>
 <!ELEMENT DISTNUM %STRDOM;>

```

<ELEMENT PROJECTTYPE %STRDOM;>

<ELEMENT TASKNUM %STRDOM;>

<ELEMENT CONVRATE %STRDOM;>

<ELEMENT DESTTYPE %STRDOM;>

<ELEMENT DFFPOHEADER
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<ELEMENT
DFFVENDORSITE(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,A
TTIBUTE5?,ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIB
UTE10?,ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?
,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE16?)>

<ELEMENT DFFVENDOR
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?,ATTRIBUTE10?,ATTRIBUT
E11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,ATTRIBUTE15?,ATTRIBUTE
16?)>

<ELEMENT DFFLINE
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?
,ATTRIBUTE10?,ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,
ATTRIBUTE15?,ATTRIBUTE16?)>

<ELEMENT DFFITEM
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?
,ATTRIBUTE10?,ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,
ATTRIBUTE15?,ATTRIBUTE16?)>

<ELEMENT KFFITEM
(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,ATTRIBUTE5?,ATTR
IBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?
,ATTRIBUTE10?,ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,
ATTRIBUTE15?,ATTRIBUTE16?,ATTRIBUTE17?,ATTRIBUTE18?,ATTRIBUTE19?,
ATTRIBUTE20?)>

<ELEMENT
DFFDISTRIBUTN(ATTRIBUTE1?,ATTRIBUTE2?,ATTRIBUTE3?,ATTRIBUTE4?,AT

```

TRIBUTE5?,ATTRIBUTE6?,ATTRIBUTE7?,ATTRIBUTE8?,ATTRIBUTE9?
 ,ATTRIBUTE10?,ATTRIBUTE11?,ATTRIBUTE12?,ATTRIBUTE13?,ATTRIBUTE14?,
 ATTRIBUTE15?,ATTRIBUTE16?)>
 <!ELEMENT ATTRIBUTE1 %STRDOM;>
 <!ELEMENT ATTRIBUTE2 %STRDOM;>
 <!ELEMENT ATTRIBUTE3 %STRDOM;>
 <!ELEMENT ATTRIBUTE4 %STRDOM;>
 <!ELEMENT ATTRIBUTE5 %STRDOM;>
 <!ELEMENT ATTRIBUTE6 %STRDOM;>
 <!ELEMENT ATTRIBUTE7 %STRDOM;>
 <!ELEMENT ATTRIBUTE8 %STRDOM;>
 <!ELEMENT ATTRIBUTE9 %STRDOM;>
 <!ELEMENT ATTRIBUTE10 %STRDOM;>
 <!ELEMENT ATTRIBUTE11 %STRDOM;>
 <!ELEMENT ATTRIBUTE12 %STRDOM;>
 <!ELEMENT ATTRIBUTE13 %STRDOM;>
 <!ELEMENT ATTRIBUTE14 %STRDOM;>
 <!ELEMENT ATTRIBUTE15 %STRDOM;>
 <!ELEMENT ATTRIBUTE16 %STRDOM;>
 <!ELEMENT ATTRIBUTE17 %STRDOM;>
 <!ELEMENT ATTRIBUTE18 %STRDOM;>
 <!ELEMENT ATTRIBUTE19 %STRDOM;>
 <!ELEMENT ATTRIBUTE20 %STRDOM;>
 <!ELEMENT TANDC %STRDOM;>
 <!ELEMENT GLOBALCONTRACT %STRDOM;>
 <!ELEMENT GLOBALCONTRACTLIN %STRDOM;>
 <!ELEMENT CONSIGNEDINV %STRDOM;>

Open Interface Tracking

Overview

The two main categories that comprise messages are Electronic Data Interchange (EDI)

flat files and Open Applications Group (OAG) XML. The Open Interface Tracking form provides a unified view of Order Management's EDI and XML messages.

You can obtain the history of messages or delete historical records using a purge procedure. The form provides you with the current status and a textual message pertaining to processing of that message, or if using Oracle Workflow, you can view the diagram. Similarly, for XML messages, you can view the XML document. You can determine the specific outbound message (if any) triggered by a particular inbound message and vice versa.

This functionality will impact Order Import and the EDI/XML transaction processing. Both Order Import of closed orders and HVOP are not supported.

Order Management's Business Event System (BES)-based mechanism for other products (such as the Supply Chain Trading Connector module) captures and records the transaction history as XML messages exchanged between trading partners. The same mechanism has been extended to include other electronic messages and to make this information available via the form.

Major Features

Search For EDI/XML Electronic Messages

You can query up historical data for electronic messages received and processed by Order Management based upon a variety of search criteria such as Transaction Type, Order Source, etc.

View XML Data For a Particular Message

The XML document may convey important information, so you can navigate to the corresponding XML for a particular message.

View Workflow for XML Messages

You can navigate to the corresponding workflow for Order Management's XML messages.

Provides a Visual Cue To Associate Related Inbound and Outbound Messages

You can optionally group related Inbound and Outbound messages for a particular order. For instance, you can visually associate the inbound Cancel PO XML message with its corresponding outbound Confirm BOD and Acknowledgment. Similarly, an 850 Purchase Order Inbound EDI messages is associated with the corresponding 855 Purchase Order Acknowledgment Outbound.

Purge Procedure

You can purge the tracking data using a concurrent program. This concurrent program can also be used to purge the interface or acknowledgment tables.

Business Scenarios and Process Flow

INBOUND

1. For EDI/XML, the inbound Electronic Message is consumed using the appropriate Gateway and the Order Management Open Interface tables are populated. A record is created for this message with the information that data has been entered into the interface tables.
2. If the Electronic Message is processed using a workflow (i.e. for XML messages), then the record is updated with information regarding the current workflow status as the flow proceeds.
3. If Order Import (EDI/XML or standard) is run on the data, the record for the message is updated with information regarding the completion of Order Import.

OUTBOUND

1. Outbound EDI/XML Electronic Messages are generated via a concurrent program or when a business event is raised or when certain pre-determined attributes of the order are updated. When the triggering condition occurs, a record is created for this message with the information that the outbound Electronic Message will be generated.
2. If the Electronic Message is processed using a workflow (i.e. for XML messages), then the record is updated with information regarding the current workflow status as the flow proceeds.
3. When the appropriate Gateway generates and transmits the outbound Electronic Message, the record is updated with the information that the message has been sent.

The messages are logged as follow: a business event (**oracle.apps.ont.oi.xml_int.status**) is raised, and a seeded workflow subscription executes and writes the event information to a table.

Messages

This table is for any form-related messages that are seeded.

Form Related Messages That Are Seeded

Code	Meaning
OE_OI_IFACE	&TRANSACTION - Data successfully entered into Open Interface Tables
OE_OI_IMPORT_MODE_SYNC	&TRANSACTION - Order Import will be run in synchronous mode
OE_OI_IMPORT_MODE_ASYNC	&TRANSACTION - Order Import will be run in asynchronous mode
OE_OI_IMPORT_SUCCESS_GEN	&TRANSACTION - Order Import successful
OE_OI_IMPORT_FAILURE	&TRANSACTION - Order Import failed
OE_OI_OUTBOUND_TRIGGERRED	OM is ready to send &TRANSACTION message
OE_OI_OUTBOUND_SETUP	&TRANSACTION - Setup Validation successful
OE_OI_OUTBOUND_SETUP_ERR	&TRANSACTION - Setup Validation failed

Profile Options**Profile Options**

Function	User Function Name	Description	Usage
ONT_EMINTEG_SOURCES	OM: Electronic Message Integration Event Sources	Order Sources enabled for Integration Business Event	Profile option to control Tracking

Electronic Message Status-Message Mapping

To provide a useful picture regarding the current processing status of an electronic message, the following messages will be written to the history table:

Inbound XML Messages

Status	Message
Active	&TRANSACTION - Data successfully entered into Open Interface Tables
Active	&TRANSACTION - Order Import will be run in synchronous mode
Active	&TRANSACTION - Order Import will be run in asynchronous mode
Success/Error	&TRANSACTION - Order Import successful/failed

Outbound XML Messages

Status	Message
Active	OM is ready to send &TRANSACTION message
Active/Error	&TRANSACTION - Setup Validation Successful/Failed
Success/Error	&TRANSACTION - message has been sent

Inbound EDI Messages

Status	Message
Active	&TRANSACTION - Data successfully entered into Open Interface Tables
Success/Error	&TRANSACTION - Order Import successful/failed

Outbound EDI Messages

Status	Message
Active	EDI Acknowledgment values derived
Success/Error	&TRANSACTION - message has been sent

Standard Order Import

Standard Order Import

Status	Message
Success/Error	&TRANSACTION - Order Import successful/failed

Event/Event Subscription

The following event **oracle.apps.ont.oi.xml_int.status** is seeded

The following event subscription to the above event will also be seeded:

Seeded workflow OEM/Open Interface Tracking

The workflow (item type code OEM, name OM: Open Interface Tracking) will be seeded. This workflow will write the data from the event **oracle.apps.ont.oi.xml_int.status** to the table.

Event Parameters & Population

Event Parameters

The event **oracle.apps.ont.oi.xml_int.status** will contain the following parameters:

Event Parameters

Parameter Name	Parameter Description	Comments
XMLG_INTERNAL_CONTR OL_NUMBER	Unique document number generated by XML Gateway for inbound messages	Overloaded to provide a unique id for EDI inbounds and Order Import as well
XMLG_MESSAGE_ID	Unique message id generated by XML gateway for outbound messages	
XMLG_INTERNAL_TXN_TY PE	Typically 'ONT' except for Confirm BOD which is owned by 'ECX'	
XMLG_INTERNAL_TXN_SU BTYPE	Identifies XML transaction type	Overloaded to also contain EDI and Order Import transaction type, namely '850', '860', '855', '865', 'GENERIC'
DOCUMENT_DIRECTION	'IN' or 'OUT'	
XMLG_DOCUMENT_ID	Unique ID provided by OM for outbound messages	Overloaded to provide a unique id for outbound EDI transactions
TRADING_PARTNER_TYPE	'C'	
TRADING_PARTNER_ID	TP Party ID	
TRADING_PARTNER_SITE	TP Party Site ID	
DOCUMENT_NO	Sales Order Number	
ORG_ID	Organization ID	
PARTNER_DOCUMENT_NO	Purchase Order Number	
DOCUMENT_REVISION_N O	Change Sequence	
ONT_DOC_STATUS	'ACTIVE', 'SUCCESS', 'ERROR'	

Parameter Name	Parameter Description	Comments
MESSAGE_TEXT	FND MESSAGE detailing what has occurred.	
WF_ITEM_TYPE	Workflow Info for XML processing	
WF_ITEM_KEY	Workflow Info for XML processing	
ORDER_SOURCE_ID	Order Source	
SOLD_TO_ORG_ID	Customer ID	
ORDER_TYPE_ID	Order Type ID	
CONC_REQUEST_ID	Processing Concurrent Request ID	
PROCESSING_STAGE	Code to identify why this event was raised. Sample values include 'INBOUND_IFACE', 'OUTBOUND_SENT'.	
SUBSCRIBER_LIST	Comma delimited list of product short names of the intended subscribers e.g. 'ONT,CLN'	
DOCUMENT_STATUS	'SUCCESS', 'ERROR'	
HEADER_ID	Order Header identifier	

Parameter Name	Parameter Description	Comments
ORDER_PROCESSED_FLAG	Flag to indicate whether data has been successfully resolved to an order in the system. This flag is needed because even if Order Import failed for XML transactions, it is possible that order number, order type id, and header id are still derived for the events raised by Acknowledge PO (e.g. if Order Import failed due to duplicate order source and orig sys document ref). Similarly, for 3A8, 3A9 inbounds, the order number, header id, and order type id are derived prior to running Order Import, but Order Import could subsequently fail. Thus, this flag is set to N if Order Import has not run or if Order Import failed for both inbound and outbound transactions to allow subscribers to identify the above cases.	

Not every parameter will be populated in all cases. The following table contains a Y if a particular parameter is populated

Populated Parameters

Parameter Name	INBOUND - Parameter Populated (Electronic Message Types)	OUTBOUND - Parameter Populated (Electronic Message Types)
XMLG_INTERNAL_CONTR OL_NUMBER	Y (XML, EDI, Order Import)	
XMLG_MESSAGE_ID		Y (XML, if document sent successfully)

Parameter Name	INBOUND - Parameter Populated (Electronic Message Types)	OUTBOUND - Parameter Populated (Electronic Message Types)
XMLG_INTERNAL_TXN_TY PE	Y (XML, EDI, Order Import)	Y (XML, EDI)
XMLG_INTERNAL_TXN_SU BTYPE	Y (XML, EDI, Order Import)	Y (XML, EDI)
DOCUMENT_DIRECTION		Y (XML, EDI)
XMLG_DOCUMENT_ID		Y (XML, EDI)
TRADING_PARTNER_TYPE		Y (XML, except CBOD)
TRADING_PARTNER_ID		Y (XML, except CBOD)
TRADING_PARTNER_SITE		Y (XML, except CBOD)
DOCUMENT_NO	Y (XML, EDI, Order Import) (if import succeeds)	Y (XML if import succeeds or sales order already exists, EDI)
ORG_ID	Y (XML, EDI, Order Import)	Y (XML, EDI)
PARTNER_DOCUMENT_NO	Y (XML, EDI, Order Import)	Y (XML, EDI)
DOCUMENT_REVISION_N O	Y (XML, EDI, Order Import)	Y (XML, EDI)
DOCUMENT_STATUS	Y (XML, EDI, Order Import)	Y (XML, EDI)
ONT_DOC_STATUS	Y (XML, EDI, Order Import)	Y (XML, EDI)
MESSAGE_TEXT	Y (XML, EDI, Order Import)	Y (XML, EDI)
WF_ITEM_TYPE	Y (XML)	Y (XML, EDI *)
WF_ITEM_KEY	Y (XML except when PROCESSING_STAGE is IMPORT_SUCCESS or IMPORT_FAILURE)	Y (XML, EDI *)

Parameter Name	INBOUND - Parameter Populated (Electronic Message Types)	OUTBOUND - Parameter Populated (Electronic Message Types)
ORDER_SOURCE_ID	Y (XML, EDI, Order Import)	Y (XML, EDI)
SOLD_TO_ORG_ID	Y (XML, EDI, Order Import)	Y (XML, EDI)
ORDER_TYPE_ID	Y (XML, EDI, Order Import) (if import succeeds)	Y (XML if import succeeds or sales order already exists, EDI)
CONC_REQUEST_ID	Y (XML, EDI, Order Import)	Y (XML if Conc Show SO)
PROCESSING_STAGE	Y (XML, EDI, Order Import)	Y (XML, EDI)
SUBSCRIBER_LIST	Y (XML, EDI, Order Import)	Y (XML, EDI)
RESPONSE_FLAG	Y (XML only for Change PO response)	

The following table maps the value of the event parameter PROCESSING_STAGE to the message text for that stage:

Processing_Stage

XML	EDI	Order Import	Processing Stage Description (CODE)	Sample Message Text
Y	Y		Data received (INBOUND_IFACE)	"Process PO – Data successfully entered into Open Interface tables" "850 POI – Data successfully entered into Open Interface tables"

Y				Import mode (PRE_IMPORT)	"Process PO – Order Import will be run in Asynchronous mode" "Process PO – Order Import will be run in Synchronous mode"
Y	Y		Y	Order Import result (IMPORT_SUCC ESS, IMPORT_FAILU RE)	"Process PO - Order Import successful" "Process PO – Order Import failed"
Y	Y			Outbound message triggered (OUTBOUND_T RIGGERED)	"OM is ready to send the Acknowledge PO message" "OM is ready to send the Show SO message"
Y	Y *			Outbound setup result (OUTBOUND_S ETUP)	"Acknowledge PO – Setup validation successful" "Acknowledge PO – Setup validation failed" "EDI Acknowledgmen t values derived"
Y	Y			Outbound sent (OUTBOUND_S ENT)	"Acknowledge PO message has been sent" "855 POAO message has been sent"

Note: For Outbound_Triggered and Outbound_Setup, the EDI support is only active when the Pack J EDI workflow is being used.

Oracle Order Management EDI Transactions

This chapter covers the following topics:

- Oracle Order Management
- Inbound Purchase Order (POI/850/ORDERS)
- Order Import
- Review and Resolve Order Management Open Interface Exceptions

Oracle Order Management

Please refer to the following guides for more information on setting up and using EDI Transactions in Oracle Order Management:

- *Oracle e-Commerce Gateway Implementation Manual*
- *Oracle e-Commerce Gateway User's Guide*
- Integration Repository for more information on the extensible EDI APIs: Purchase Order Acknowledgments Extension Columns API and Purchase Order Change Acknowledgments Extension Columns API.

The following table summarizes the EDI Transactions:

Release Management Transactions

Transaction Name	Direction	Tran. Code	ASC X12	EDIFACT
Purchase Order	Inbound	POI	850	ORDERS

Transaction Name	Direction	Tran. Code	ASC X12	EDIFACT
Purchase Order Change	Inbound	POCI	860	ORDCHG
Purchase Order Acknowledgement	Outbound	POAO	855	ORDRSP
Purchase Order Change Acknowledgement	Outbound	POCO	865	ORDRSP

Inbound Purchase Order (POI/850/ORDERS)

This transaction is the inbound purchase order from the customer's procurement application into the Oracle Order Management application. This transaction uses the Order Management Order Import open interface. Refer to the *Oracle Manufacturing APIs and Open Interface Manual* and the *Oracle e-Commerce Gateway Implementation Manual* for more details.

Trading Partner Link to Oracle e-Commerce Gateway

The EDI Location Code is the link between a Customer/Customer site in Oracle Applications and the trading partner site definition in Oracle e-Commerce Gateway. This enables Oracle e-Commerce Gateway to access the detailed data about the Customer or Customer site in the base Oracle Applications without maintaining the detail data in Oracle e-Commerce Gateway.

Refer to the Trading Partner chapter in the *Oracle e-Commerce Implementation Manual* for recommendations on selecting the correct trading partner EDI Location Code for the control record 0010 for the transaction in the transaction interface file.

Oracle e-Commerce Gateway Required Fields

The following is a list of the Oracle e-Commerce Gateway required fields. These fields are required to authenticate the trading partner and transaction. If the required data is not provided in the transaction, the Oracle e-Commerce Gateway import process fails the transaction. Then an exception message is displayed in the View Staged Documents window.

If the trading partner is valid and the transaction is enabled, the import process proceeds to validate the transaction using the user-defined column rules. If no process

or column rule exceptions are detected, the Oracle e-Commerce Gateway import program will write the transaction to the Order Management Open Interface tables to be processed by the Order Management Open Interface API.

Control Record Details

Oracle e-Commerce Gateway Column Name for Required Fields	Record Number	Position Number	Note
TEST_INDICATOR	0010	20	T or P
TP_DOCUMENT_ID	0010	30	POI
TP_TRANSLATOR_C ODE	0010	70	Translator identifier for this trading partner
TP_LOCATION_CO DE	0010	80	The EDI Location Code

Review and Resolve Oracle e-Commerce Gateway Exceptions

The View Staged Documents window enables you to review the Oracle e-Commerce Gateway transaction exceptions as specified by the process and column rules in the e-Commerce Gateway. To resolve Oracle e-Commerce Gateway exceptions, you can either correct the set up data in Oracle e-Commerce Gateway or Oracle Applications, or ask the Trading Partner to send a corrected transaction. Once the exceptions are identified and resolved, you can submit the transaction for reprocessing, ignore the exception during reprocessing, or delete the transaction. Select the preferred option in the View Staged Documents window.

Relevant Oracle Order Management Profiles and Setup Steps

The required Order Management setup steps are the same whether the order is entered on-line or through Order Import. The following is a sample list of the Order Management setup steps needed to process orders through the e-Commerce Gateway or entered on-line.

- Customer and Customer Sites must be defined in Order Management/ Oracle Receivables.
- Order Types must be defined in Order Management.

- Item and appropriate customer items must be defined in Oracle Inventory.

Oracle Order Management Defaulting Functionality

Oracle Order Management provides powerful Defaulting functionality that is used by the Order Import process, as well as by orders entered manually using the Order Management forms. If a column is left null in the interface tables, then the Defaulting Rules are invoked, to provide a value for those columns.

Order Import Open Interface Data Levels

The EDI standard transaction has three (3) key levels of data: Header, Line (item), and Schedules (Shipments).

The Order Import open interface tables have a two level structure where the order line and a single shipment are combined into one table entry for each combination. Order Import requires that line and shipment data be consolidated into the LINE level table.

The e-Commerce Gateway transaction accommodated the three level transactions so that the item data is written only once on Record 2000 (also records 2010/2020 if needed) followed by all the shipment (schedules) in records 3000-4900. The process will combine the item data to each set of its associated shipment data before writing the data to the Order Import LINE level table.

Original System Reference Data in all Order Import Tables

Order Import requires a set of Original System reference fields. These are derived by the e-Commerce Gateway transaction process if one is not found in the transaction.

The Original System reference fields are used to match change orders to original orders. Thus, these values must be unique within all orders for a customer and entered with the new orders.

The POI transaction program derives the header and line level Original System reference fields, if data is not supplied in the transaction when the transaction was created.

Original System Document Reference

Original System Document Reference is an identifier for the order within Order Management to guarantee its uniqueness within one operating unit.

Most often, it is the purchase order number, plus release number (if appropriate). A character such as the dash sign between each component improves readability.

The ORIG_SYS_DOCUMENT_REF is derived by the POI program or generated by a process such as the EDI Translator before the order is imported into the Order Import tables. It is not sent by the customer in the transaction as a field known as Original System Document Reference.

Order Import

Flags

There are several flags in the interface tables of Order Management that affect the Order Import processing. Valid values of these flags are Y, N, and null. Null means different things depending on the particular flag. The Order Import process sets the defaults if a code is not found in the transaction. These flags are: Booked, Cancelled, Closed, Error, Ready, Rejected.

Most of the flag data elements are not activated on the transaction, since the defaults are usually adequate. Their data elements are listed at the end of the activated data elements in the Transaction Definition window.

Order Import Open Interface Data

The Order Management Order Import Open Interface is used by the Inbound Purchase Order transaction. This validates the incoming data entered into the Order Management Open Interface tables by the Oracle e-Commerce Gateway import program or any other loading program.

The data for the Order Import tables have the following sources:

- Data in the transaction from the trading partner or determined by the EDI Translator.
- Data derived or hard codes by the POI Program given the trading partner or the presence of data in the transaction.
- Data derived by the Order Import process when it calls APIs to the Oracle tables.

Review and Resolve Order Management Open Interface Exceptions

At the completion of the Order Management Open Interface Order Import program, the orders with exception can be viewed in the Order Import window.

There are three ways to resolve Order Management Open Interface exceptions as follows:

Using the information given by the View Staging Documents window in the e-Commerce Gateway, correct the set up data in Oracle Applications.

Correct erroneous entries in the Order Management Order Import window.

Request the customer to send a corrected transaction.

If you chose to update Oracle Applications data or change the erroneous entries using the Order Import window in Order Management, you can resubmit the Order

Management Open Interface Order Import process to revalidate the transaction.

If you chose to have the customer send a corrected transaction, you must first delete the rejected Order data in the Order Management Open Interface tables using the Order Import window and then re-import the updated transaction using the Oracle e-Commerce Gateway.

Refer to the *Oracle Order Management User's Guide*, and the *Oracle Manufacturing APIs and Open Interface Manual* for more detail.

Oracle Order Management Open Interfaces and APIs

This chapter covers the following topics:

- Integrating Oracle Order Management Using Order Import
- Process Order Application Open Interface

Integrating Oracle Order Management Using Order Import

Overview

You must manually submit the Order Import Statistics concurrent program prior to submitting the Order Import concurrent program if you wish to optimize interface record processing.

Order Import, like the rest of Oracle Applications, uses the Cost Based Optimizer of the database for optimizing queries. The Cost Based Optimizer uses generated statistical information to optimize queries. The Order Import Statistics concurrent program gathers statistics that will be used by the cost based optimizer. This concurrent program should be run after data is populated into the interface tables.

See: Order Import Concurrent Program.

Order Import is an Order Management Open Interface that consists of open interface tables and a set of APIs. Order Import can import new, change, and completed sales orders or returns from other applications such as a legacy system. You can also import cancelled, closed and booked orders. The orders may come from any source such as EDI transactions that are processed by the Oracle e-Commerce Gateway or internal orders created for internal requisitions developed in Oracle Purchasing or returns.

Order Import features include validation and defaulting, processing constraint checks, applying and releasing of order holds, scheduling of shipments, then ultimately inserting, updating or deleting the orders in the base Order Management tables. Order Management checks all the data during the import process to ensure its validity within

Order Management. Valid transactions are then converted into orders with lines, reservations, payments, price adjustments, and sales credits in the base Order Management tables.

You can use the Order Import Correction window to examine the order and optionally correct data if it fails the import process. You can use the Error Message window to determine the error if your data failed to import.

Each time you run Order Import, Order Management produces a summary of information letting you know of the total number of orders that Order Import evaluates, and succeeded or failed.

Prerequisites and Set-Up

Before using this program to import orders, you should:

- Set up every aspect of Order Management that you want to use with imported orders, including customers, pricing, items, and bills.
- Define and enable your Order Import sources using the Order Import Sources window.
- Determine if you should submit the Order Import Statistics concurrent program.

Multi-Org Access Control in Order Import

The Operating Unit is added as a new parameter to the Order Import concurrent program. If you do not specify an Operating Unit then Order Import concurrent request will allow you to import Orders into the different Operating Units that you have access to.

If you do specify an Operating Unit, then the concurrent program will import Orders only for the selected Operating Unit.

If there is no Operating Unit specified on the interface records, Order Import attempts to process those records for the default Operating Unit. Order Import updates such records (with no Org Id) with the Org ID for the Operating Unit that they were processed for.

By default this parameter (Process Orders With No Org Specified) will be set to Y, to maintain backward compatibility. If you always populate the Operating Unit field in the interface table records then you should change it to N. It is recommended that you always specify a value for the Operating Unit (Org ID) when you populate the interface tables.

Order Management Parameters

The following Order Management parameters affects the operation of the Order Import program:

- *Item Validation Organization*--Determines the organization used for validating items

and bill of material structures.

- *Reservation Time Fence*--This system parameter controls automatic reservations during scheduling.
- *Schedule Line on Hold*-- This system parameter decides if Order Management scheduling should schedule lines that are on hold.

Profile Options

- *OM: Apply Automatic Attachments*--This profile option determines whether rule-based attachments are applied without user intervention.
- *OM: AutoSchedule*-- This profile decides whether the order line should be automatically scheduled or not.
- *OM: Authorized to Override ATP*-- This profile provides the authorization to users to override the scheduling results.
- *OM: Import Multiple Shipments*-- This profile option is used to import multiple shipments via Order Import. If this profile option is set to Yes, the `orig_sys_shipment_ref` will also be used in determining the uniqueness of the line record in combination with `order_source_id`, `orig_sys_document_ref` and `orig_sys_line_ref`.
- *OM: Unique Order Source, Orig Sys Document Ref Combination for Each Customer*-- This profile determines whether or not customer information will be utilized to distinguish orders with the same reference information.

Items and Bills

Order Management uses the same customer, item pricing, and bill attribute validation and logic for imported orders as for orders entered in the Sales Orders window.

You need to define items using Oracle Inventory for items to be orderable via Order Import. You also need to define bills of material in Oracle Bills of Material for models if you have any complex items that customers can order in various configurations.

Order Import provides the ability to import an item specified in the following supplier, customer or generic formats:

- Supplier Specific Internal Part number
- Customer Specific Item number
- Generic (depending on what you have set up in Oracle Inventory as cross-references):
 - CLEI (Common Language Equipment Identifier)

- EAN (European Article Number) code
- ISBN (International Standard Book Number)
- JAN (Japanese Article Number) code
- UPC (Universal Product code) code

Customer Relationships

Site Level Customers and Sites for Bill_To, Ship_To and Deliver_To will be validated based on the Customer Relationships profile.

This parameter has three setting:

Single Customer: Site Can only belong to the Sold to Customer. The Site level Customer cannot be different than the Sold to Customer.

Related Customers: Site Can only belong to the Sold to Customer or a Customer Related to the Sold To Customer . The Site level Customer can be different than the Sold to Customer but the relationship should exist between the site level customer and the Sold to Customer.

All Customers: Site Can only belong to the Sold to Customer or any other Customers. The Site level Customer can be different than the Sold to Customer and it is not required that relationship should exist between the site level customer and the Sold to Customer.

When checking for relationship we also check the appropriate usage in the Relationship also.

For example, customer "A" can be a valid related BillTO customer to Customer "B," only if the relationship exists and the bill_to_Flag is checked in the relationship.

Similarly for ShipTO and Deliver_to we check the Ship_To flag.

Add Customer

Order Management provides the capability to add a new customer, the address and contacts using Order Import. The table OE_CUSTOMER_INFO_IFACE_ALL needs to be populated for this.

Based on the data available in the OE_HEADERS_IFACE_ALL the data from the table OE_CUSTOMER_INFO_IFACE_ALL is processed to add a new customer. This table can also be used to import a new address only or contact information of an existing Customer.

The link between the header interface record and the customer interface record is made through the column CUSTOMER_INFO_REF of the table OE_CUSTOMER_INFO_IFACE_ALL and any one the columns mentioned below. The link depends on the CUSTOMER_INFO_TYPE_CODE. When the order import is used to create a new customer (new account/address/contact etc), the

CUSTOMER_INFO_TYPE_CODE should be either ('ACCOUNT', 'ADDRESS', or 'CONTACT') to denote the type of information in the record. The following table explains the relationship between the various header interface table columns and the column CUSTOMER_INFO_TYPE_CODE and the usage of the combination.

OE_HEADERS_IFACE_ALL

OE_HEADERS_IFACE_ALL	Usage	CUSTOMER_INFO_TYPE_CODE	Result
Orig_Sys_Customer_Ref		ACCOUNT	New Customer account will get created
Orig_Ship_Address_Ref	SHIP_TO	ADDRESS	New Ship to address will get created
Orig_Bill_Address_Ref	BILL_TO	ADDRESS	New Bill to address will get created
Orig_Deliver_Addresses_Ref	DELIVER_TO	ADDRESS	New Deliver to address will get created
Sold_to_Contact_Ref	SOLD_TO	CONTACT	New Sold to Contact will get created
Ship_to_Contact_Ref	SHIP_TO	CONTACT	New Ship to Contact will get created
Bill_to_Contact_Ref	BILL_TO	CONTACT	New Bill to Contact will get created
Deliver_to_Contact_Ref	DELIVER_TO	CONTACT	New Deliver to Contact will get created

The following profiles affect the add customer functionality.

- OM: Add Customer (Order Import)

This will decide whether the customer can be imported using order import. This must be set to "ALL" for importing new customer account and to 'Address and Contact only' for importing addresses and contacts alone. The eligible values for the profile are as follows:

N = None

P = Address and Contact only

Y = All

- OM: Email Required on New Customers

This will decide whether the records in the oe_customer_info_iface_all should have the email or not. If the profile is set to "Yes" then you must enter a not null value in the field EMAIL_ADDRESS of the table OE_CUSTOMER_INFO_IFACE_ALL.

- HZ: Generate Party Number = 'YES' or NULL

This will decide whether the records in the oe_customer_info_iface_all should have the party number or not. If the profile is set to Yes then you must enter a not null value in the field PARTY_NUMBER of the table OE_CUSTOMER_INFO_IFACE_ALL.

- HZ: Generate Contact Number = 'YES' or NULL.

If this is set to 'N' then Order Import program will generate the Contact Number. If this is set to 'YES' or NULL then it will be generated later while creating the Contact.

- HZ: Generate Party Site Number = 'YES' or NULL.

This will decide whether the records in the oe_customer_info_iface_all should have the site number or not. If the profile is set to Yes then you must enter a not null value in the field SITE_NUMBER of the table OE_CUSTOMER_INFO_IFACE_ALL..

The value of the profile option 'Automatic Customer Numbering' and 'Automatic Site Numbering' from the AR System Options are also important. If the flags are unchecked in the option, then the following fields of the table OE_CUSTOMER_INFO_IFACE_ALL must have a not null value.

CUSTOMER_NUMBER if Automatic Customer Numbering is unchecked

LOCATION_NUMBER if Automatic Site Numbering is unchecked

Order Import will respect the setting of the OM System Parameter – Customer Relationships (Setup=>Parameters) and create customer-to-customer relationships if needed and allowed. The parameter values will be interpreted as follows:

- Customer Relationships = 'Single Customer'

The Ship_To, Deliver_To and Bill_To addresses must belong to the Sold_To customer on the order. If an attempt is made to import an order with a Bill_To, Deliver_To or Ship_To from a different customer than the Sold_To, Order Import will raise an error and the order will not be imported.

- Customer Relationships = 'Related Customers'

If a new customer is added for a Ship_To, Bill_To, or Deliver_To address, the appropriate relationship will be created in the TCA (Trading Community

Architecture)

- Customer Relationships = 'All Customers'

A new customer can be added, but no relationship creation occurs.

Importing an order using a new Customer information will require you to add and also to remove some data from the headers interface table depending on the kind of information being imported. For example if a customer account alone is to be imported then the Orig_Sys_Customer_Ref needs to be populated and the sold_to_org/sold_to_org_id should be populated as null.

The some or all of the following data is to be populated in oe_headers_iface_all table,

- Orig_Sys_Customer_Ref
- Orig_Ship_Address_Ref
- Orig_Bill_Address_Ref
- Orig_Deliver_Address_Ref
- Sold_to_Contact_Ref
- Ship_to_Contact_Ref
- Bill_to_Contact_Ref
- Deliver_to_Contact_Ref

And some or all of the following data needs to be removed from the oe_headers_iface_all tables depending upon the information being imported.

- sold_to_org / sold_to_org_id
- invoice_to_org / invoice_to_org_id
- ship_to_org / ship_to_org_id

Against this the oe_custome_info_iface_all table needs to be populated. The number of records being populated will depend on what is being imported. For example while importing only the customer account only one record will be populated and while importing account, address and contacts three or more records will be populated.

If the customer addresses are to be different for ship, bill, and deliver, then three different records need to be populated. Whether an address is for ship , bill, or deliver, is decided by the following columns of the oe_customer_info_iface_all table-

- IS_SHIP_TO_ADDRESS,
- IS_BILL_TO_ADDRESS,

- IS_DELIVER_TO_ADDRESS

These take the values 'Y,' 'N' or null. So if the particular record needs to be only a bill to address, then the IS_BILL_TO_ADDRESS should be set as 'Y' and the rest as 'N' or null. If the same address needs to be Bill to , ship to and deliver to Address then the flag should be set as 'Y' against all the three columns.

The link between the Account record and the address or contact record is through the column PARENT_CUSTOMER_REF. This should be populated in the address or contact record with the CUSTOMER_INFO_REF of the account record. For example, if the CUSTOMER_INFO_REF in the record with the CUSTOMER_INFO_TYPE_CODE as 'ACCOUNT' is 'XXXX' then the PARENT_CUSTOMER_REF for the record with the CUSTOMER_INFO_TYPE_CODE as 'ADDRESS' or 'CONTACT' should be 'XXXX.'

Defaulting Rules

You can setup your defaulting rules which allow you to default columns in the same way as for orders entered online. You can pass the column value Null to Order Import if you want the defaulting rules to populate the column. However, if the column is defined as Not Null or Mandatory column and the defaulting rules fail to default the column, for any reason, Order Import displays an error message without importing the order.

Importing from External Systems

You can import orders with any external source defined in the Define Document Sequences window.

Note: You cannot specify a value in the DROP_SHIP column of the OE_HEADERS_IFACE_ALL. If you enter a value in this column, the Order Import concurrent program will fail.

Internal Sales Orders

Oracle Purchasing uses Order Import to transfer requisitions for internally sourced products to Order Management. Once imported, the internal sales orders are processed as regular sales orders.

Returns

Returns can be imported like a regular sales order. Order Management utilizes workflow activities to import returns.

Special Considerations for importing Return orders

Creation of a non-referenced RMA

To import a return order for a non referenced return, you must:

- Populate all required attributes for creating a return order

- Use an order category of RETURN or MIXED for the Order Header record.
- Populate all required attributes for creating a return order lines.
 - For Order Line Record, you cannot specify a value for the line category_code column. You need to populate the column ordered_quantity with a negative value.
 - Line_type_id is optional, provided a default line_type has been defined for the specified Order Type.)
 - Additionally, you will need to populate the column reason_code for all return lines. Valid values are those values defined for the Order Management QuickCode CREDIT_MEMO_REASON.

Note: If the Reason Code is not provided, Order Import errors out. However we do not currently validate whether Reason Code is entered correctly. Users need to enter Reason Code when they populate the interface table, instead of the *definition* of the reason. For example: if you enter 'No reason provided' (definition) as reason but the system is expecting 'Not provided' (code), that is why the reason was not properly translated. If you enter 'Reason Code' and try again that works as expected.

- You can send the ordered_quantity as a negative on the line record and the line will be treated/created as an RMA line.

Creation of a Referenced RMA (if you want to return an existing outbound line)

If you create a referenced RMA, you should copy the Header Record for the return from the referenced Order header record, modifying the Order Type to category RETURN or MIXED (order_type_id from oe_transaction_types_all).

For the Order Line record, populate the following attributes only:

line_category_code: RETURN

1. return_context: ORDER
2. return_attribute1: header_id from the referenced order.
3. return_attribute2: line_id from the referenced order line.
4. calculate_price_flag: Set it to P if you want to retain the original price, the flag to Y if you want to reprice the RMA line.
5. line_type_id: Assign a line_type_id from RMA line. Line_type_id is optional,

provided a default line_type has been defined for the specified Order Line Type.)

6. return_reason_code: Populate a reason code from lookup_type
CREDIT_MEMO_REASON
7. For sales credit info please populate the header_level/line level sales credits details from the referenced order.

Import Types

The following enable you to import orders of various kinds.

Configurations

Order Management enables you to import ATO and PTO configurations. For EDI orders, you can import valid and invalid configurations; however, you will not be able to book orders with invalid configurations. Top model line reference needs to be populated for all lines including the model.

Note: If the same option appears more than twice in the BOM of the model, the component_code is used in the interface table to resolve the ambiguity. If only the option line is sent without a class line, then the system will derive and populate the class lines in the order.

Changes

You can import changes to orders that have been imported by passing all changed data through Order Import. You can update or delete orders, order lines, price adjustments, and sales credits. You can use change sequence numbers to control the sequence of changes you want to make to orders. You can also split lines using Order Import.

Order Status

You can import new, booked, cancelled or closed orders. If an order is imported with an entry status of Booked (OE_HEADERS_IFACE_ALL.BOOKED_FLAG=Y) the result after import is that a Action Request of BOOK_ORDER is initiated. You may also pass the Action Request to BOOK_ORDER; both methods are supported.

Order Import ensures that all required fields for entry or booking are validated appropriately as the orders are imported. Order Import imports the order in as Entered and attempts to book it. If any of the required fields for a booked order are not supplied, Order Management retains the order in the Entered status and notifies you of the error.

Line Sets

You can import grouped order lines, called sets, based on certain common attributes for a new or existing order. You can also add a line to an existing set. You will need to provide the set ID or name in the Order Import tables. If that set already exists, the line

will be included in the set. However, if the set does not already exist, a new set will be created and the line will be added to the set. In addition, if any line attribute, which is also a set attribute, does not match with the set attribute value, the set attribute value will overwrite the line attribute.

Fulfillment Set Functionality Support for Order Import

Order Import functionality supports addition, removal, and moving lines to multiple fulfillment sets. This can be achieved by populating the Actions table with corresponding action code.

User Procedures

- Populate the Order Import Header Interface tables with Header Information
- Populate the Order Import Line Interface tables with Line Information
- There are three requirements Add the line to a Specified fulfillment Set (**ADD**) and Remove the line from specified fulfillment Set (**REMOVE**) and Move the line to new fulfillment set.
- Populate the Actions Interface table with Action code and Fulfillment set Name.

Consider the following example.

- There are two lines in table OE_ORDER_LINES_ALL with the lines in fulfillment sets as shown.

Example 1

Line_id	Fulfillment Set
1000	F1
2000	F2, F3

For adding, removing and moving lines from multiple fulfillment the following steps can achieve sets.

Populate Order Import Lines Interface Table.

Example 2

Orig_Sys_line_Ref	Operation Code
1000	UPDATE
2000	UPDATE

Populate Order Import Actions Interface Table.

Example 3

For Orig_sys_line_ref	Operation Code	Fulfillment Set Name
1000	ADD_FULFILLMENT_SET	F3
1000	REMOVE_FULFILLMENT_SET	F1
2000	ADD_FULFILLMENT_SET	F4
2000	REMOVE_FULFILLMENT_SET	F2
2000	ADD_FULFILLMENT_SET	F5

After Order import the lines in the Order Lines table OE_ORDER_LINES_ALL

Example 4

Line_Id	Fulfillment Set
1000	F3
2000	F3, F4, F5

Workflows

You can import an order within any valid order workflow activity. The order must be at the initial activity of Entered, Booked, or Closed. Orders imported using Order Import cannot be in the middle of a workflow activity.

Manual and Automatic Pricing

You can indicate whether you want to manually enter prices for imported orders or allow Order Management to automatically price the order. You can use automatic pricing or manual pricing for your imported orders. If you want to use automatic pricing, you should set the column `OE_LINES_INTERFACE.CALCULATE_PRICE_FLAG` to Y, and define all your pricing setup including discounts, promotions, surcharges, free goods, etc. in Oracle Pricing and Order Management. However, if you want to use the manual pricing, you should set the column `OE_LINES_INTERFACE.CALCULATE_PRICE_FLAG` to N. In this case, you should define all your discounts as line level, overridable, and not automatic.

Note: Order Import now allows you to import Coupons, Promotions, Surcharges, and Pricing Attributes.

Pricing Agreements

You can specify an agreement name if you want to order against a specific customer agreement for an order or order line.

Scheduling

Order Import enables you to reserve orders as they are imported, using the same rules as online order entry. If the scheduling request is unsuccessful on an imported order, the order import fails, and the scheduling exceptions can be viewed in the Error Messages of the Order Import Corrections window. You can use Schedule, Unschedule, Reserve or Unreserve as values for scheduling actions.

To override the line, Schedule Ship Date or Arrival Date has to be populated on the line record. The dates populated are based on the Ordered Date Type. If the Ordered Date Type is 'Ship' then populate the ship date otherwise populate the Arrival Date.

Validations

Process Order API

Process Order is the central application process interface (API) provided by Order Management to perform all common operations such as inserting, updating, deleting, and validating an order or order line. The API also performs the scheduling and returns a promise date. This API is called by Order Import.

Order Import passes one order, with all lines and other entities, at a time to the Process Order Interface, along with the operations that need to be completed on the order or line such as, inserting or updating an order or line. Errors at any line or entity level will cause the order to fail the importing of the entire order. In addition, Order Import processes only those orders and lines which are not rejected and do not have the `ERROR_FLAG` column set to Y from previous processes.

Attachments

Order Management applies any automatic attachments to imported orders that meet your automatic note criteria based on the setting of the *OM: Apply Automatic Attachments* profile option.

Credit Checking

Order Management performs credit checking on all imported orders or changes, according to the credit checking rules you have defined in Order Management.

Defaulting Rules

You can pass the column value Null to Order Import if you want the defaulting rules to populate the column. However, if the column is defined as Not Null or Mandatory column and the defaulting rules fail to default the column, for any reason, Order Import displays an error message without importing the order.

Holds and Releases

Order Management automatically applies all holds to imported orders and order lines that meet hold criteria. Order Import allows you to apply holds on imported orders for review, just as you would for orders entered through the Sales Orders window. You can also apply holds or release holds using the actions interface table.

Price Comparisons

Order Import performs a price comparison on your imported orders. For example, if you provide a selling price and also want the system to calculate a price, Order Import warns you of the differences, if any, between the two prices as discrepancies. The warning can be viewed in the Error Message window of the Order Import Corrections window.

If there is a difference between your selling price and the system calculated price, Order Import raises a warning of the difference. Order Import saves your customer-provided value for the selling price in a column on the order line table, so you can have visibility to what your customer sent in.

Note: You cannot copy or interface an order line having a price list with a currency code different from the existing or newly created order header's currency code. If you click the Error button, error messages are displayed.

Payment Term Comparison

Order Import performs payment term comparisons. If there is a difference between your payment terms, Order Import raises a warning of the difference. Order Import saves your customer-provided value for payment terms in a column on the order line table so that you can have visibility to what your customer sent in.

Processing Constraints

Order Import checks the processing constraints you have defined in Order Management to assure that any operation such as insert, update, cancel, split and delete are acceptable by your security standards. Order Import displays an error message if it encounters a processing constraint that has been violated.

Corrected Data

Once the data is corrected, the ERROR_FLAG for the record is updated to N. You can set the REJECT_FLAG to Y for headers and line in case your data cannot be corrected by using the Order Import Corrections window.

Order Import Statistics Concurrent Program

The Order Import Statistics concurrent program performs a table analysis of all interface tables related to Order Import for determining optimum record processing should the Order Import concurrent program be submitted. You can choose to submit this program (or not) prior to each submission of the Order Import concurrent program. If you normally process a similar number of interface records, you typically do not need to submit this program prior to submitting the Order Import concurrent program.

There are no parameters for the submission of the Order Import Statistics concurrent program. See: *Oracle Order Management User's Guide*, Order Import Statistics Concurrent Program

Request Submission

You can submit a request by selecting Order Import Request. You can run the Order Import process in the validation-only mode.

This mode allows the transaction to be validated against all the Order Management rules but not pass valid transactions to the base Order Management tables. If you choose you can run production transactions in validation-only mode for a preview of exceptions. Make necessary corrections to the transactions in the Order Import window, then choose the Validate button to perform a validation check. The validation-only mode may also facilitate testing of transactions through Order Import even in a production environment to take advantage of all the setup is the production environment.

Parameters

The Order Import program provides the following parameters:

- Operating Unit

By default Order Import processes records with no Org ID specified, when you run Order Import and specify your default Operating Unit or if the Operating Unit parameter is left blank. The records will be processed for your default Operating Unit. However Order Import is more efficient if you explicitly populate the Org ID on the interface table records and set the hidden parameter Process Orders with no

Org specified for the concurrent program to No (using the Application Developer responsibility).

If you always specify the Org ID on your interface table records then you should set the hidden parameter Process Orders with no Org specified for the concurrent program to No (using the Application Developer responsibility).

- Order Source

Choose a specific Order Import source that you have defined in the Order Import Sources window so that only records with that source are processed, or leave this parameter blank so that all enabled sources are processed for data existing in the interface tables.

- Order References

You can enter the System Document Reference if you want to run Order Import for a specific order.

- Validate Only (Yes/No)

Choose whether to validate only the data in the interface tables. If Yes, the order will be validated, but not imported into the base orders tables. The default value is No. In the Order Import Concurrent Program window, the "Instances" parameter is also displayed. This parameter determines the maximum number of child processes you wish Order Import to spawn.

- Validate Descriptive Flex fields (Yes/No)

If the value selected is Yes then the flexfields will be validated; if the value of the parameter is No then flexfields will not be validated.

Purge Open Interface Data Concurrent Program

The Purge Open Interface Data concurrent program allows you to delete records that are no longer required from the interface tables.

Parameters

The Order Import program provides the following parameters:

- View Name (required)

Choose a view to be purged. When you choose a particular view, any child views are automatically purged. For instance, if you select Headers Interface, all the child records of the deleted Headers Interface records from the Lines Interface will also be purged. Note that this concurrent program can also be used to purge Acknowledgment or Open Interface Tracking data.

- Customer Number

You can optionally specify a Customer Number so that only records with the

Sold_To_Org_Id or Sold_To_Org corresponding to the Customer Number are deleted.

- Order Source

You can optionally specify an Order Import source that you have defined in the Order Import Sources window so that only records with that source are deleted.

- Order Reference To/Order Reference From

You can optionally enter a System Document Reference range so that only records with order references within that range are deleted.

Order Import Corrections Window

The Order Import Corrections window consists of the Find and Summary windows. The Find window allows you to find orders to be imported based on certain attributes such as Operating Unit ID, Request ID, Order Source, Original System Document Reference, Sold To Org ID, and Change Sequence. The Operating Unit ID field shows your default operating unit. If you specify a value for Operating Unit, your search will be restricted to the specified operating unit.

The Summary window displays order headers, lines, sales credits, price adjustments, lot serials, payments, reservations and action requests information. You can enable or disable the display of columns in the window using the folder functionality

The Order Import window displays all orders or selected orders based on the criteria given in the Find window. You can modify the orders here. The orders that have errors display in red.

You can insert, update, and delete the orders and lines in the interface tables. You can update one or multiple orders or lines at the same time through the Summary window. You can also mark an order or a line to be rejected by setting the REJECTED flag. There are separate windows for the header and line level data. These windows have related fields grouped as tabs.

Buttons

- *Lines*: Displays line level information for orders.
- *Discounts*: Displays discount information for orders.
- *Validate*: Validates the data but does not import it. Only the selected orders will be validated and performed online.
- *Import*: Imports the orders. The data is validated before being imported. If an error is encountered while importing, the order will be rejected and the error messages can be viewed by choosing the Errors button. Only the selected orders will be imported and the import is performed online. If an order is successfully imported, it also gets deleted from the interface tables. If you attempt to re-query the window,

you will not be able to view that order in the Order Import Corrections window.

- *Errors*: Displays all the errors encountered while importing. The error messages are stored context sensitive. If you choose the Errors button from the Order Headers region, all the errors for that order are displayed. If you choose the Errors button from the Lines region, all the errors are displayed for that line. If you encountered errors while importing orders, you can also fix these errors in the window and try importing the order again. You can navigate from the Errors window to the Order Headers or Lines region where the error has occurred.
- *Actions*: Displays order actions for orders.
- *Sales Credits*: Displays sales credit information for orders.
- *Add Customer*: To add new Customer and Address during the Import process.
- *Pricing Attributes*: To add Pricing Attributes for Order/Line.
- *Payments*: To add payments details for orders.

Processing Results

Each time you run Order Import, Order Management automatically generates an Order Import processing results summary log which identifies the total number of successful and failed imported orders.

Oracle Order Management Interface Tables and Column Descriptions

Order Import uses the following interface tables during processing:

OE_CUSTOMER_INFO_IFACE_ALL

OE_HEADERS_IFACE_ALL

OE_LINES_IFACE_ALL

OE_PRICE_ADJS_IFACE_ALL

OE_PRICE_ATTTS_IFACE_ALL

OE_CREDITS_IFACE_ALL

OE_LOTSERIALS_IFACE_ALL

OE_RESERVTTNS_IFACE_ALL

OE_ACTIONS_IFACE_ALL

ONT.OE_PAYMENTS_IFACE_ALL

A table listing for each interface table is provided below.

OE_CUSTOMER_INFO_IFACE_ALL

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CUSTOMER_INFO_REF	VARCHAR2(50)	Yes			
CURRENT_CUSTOMER_NUMBER	NUMBER	Yes, if Customer_info_Type_Code is ADDRESS or CONTACT and the corresponding customer exists already either Current_Customer_Number or Current_Customer_Id should be populated. It must not be populated if the corresponding customer is being created simultaneously.			
CURRENT_CUSTOMER_ID	NUMBER	Yes, if Customer_info_Type_Code is ADDRESS or CONTACT and the corresponding customer exists already either Current_Customer_Number or Current_Customer_Id should be populated. It must not be populated if the corresponding customer is being created simultaneously.			

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CUSTOMER_TYPE	VARCHAR2(30)	No (Default value of ORGANIZATION will be used)			
CUSTOMER_INFO_TYPE_CODE	VARCHAR2(10)	Yes			
PARENT_CUSTOMER_REF	VARCHAR2(50)	Yes, if the corresponding customer is being created simultaneously			
ORGANIZATION_NAME	VARCHAR2(30)	Yes, if Customer_Info_Type_Code is ACCOUNT and Customer_Type is ORGANIZATION			
PERSON_FIRST_NAME	VARCHAR2(150)	Yes, if Customer_Type_Code is ADDRESS and Customer_Type is PERSON			
PERSON_MIDDLE_NAME	VARCHAR2(60)				
PERSON_LAST_NAME	VARCHAR2(150)	Yes, if Customer_Type is PERSON			
PERSON_NAME_SUFFIX	VARCHAR2(30)				
PERSON_TITLE	VARCHAR2(60)				

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CUSTOMER_NUMBER	NUMBER	Populate only if Customer_info_Type_Code is ACCOUNT and automatic numbering is not set to occur in Accounts Receivable.			
EMAIL_ADDRESS	VARCHAR2(2000)	Yes, if the Order Management profile option OM: Email Required On New Customer is set to Yes and Customer_Info_Type_Code is ACCOUNT			
PARTY_NUMBER	VARCHAR2(30)	Yes, if Customer_Info_Type_Code is ACCOUNT and the customer should be added to an existing party, populate this field with that party_number. Must be left null to create a new party for the new customer.			
PHONE_COUNTRY_CODE	VARCHAR2(10)				
PHONE_AREA_CODE	VARCHAR2(10)				
PHONE_NUMBER	VARCHAR2(40)				
PHONE_EXTENSION	VARCHAR2(20)				

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
COUNTRY	VARCHAR2(60)	Yes, if Customer_Info_Type _Code is ADDRESS			
ADDRESS1... ADDRESS4	VARCHAR2(240)	Yes, if Customer_Info_Type _Code is ADDRESS			
CITY	VARCHAR2(60)	Yes, if Customer_Info_Type _Code is ADDRESS			
POSTAL_CODE	VARCHAR2(60)	Yes, if Customer_Info_Type _Code is ADDRESS			
STATE	VARCHAR2(60)	Yes, if Customer_Info_Type _Code is ADDRESS			
PROVINCE	VARCHAR2(60)				
COUNTY	VARCHAR2(60)				
IS_SHIP_TO_ADDRESS	VARCHAR2(1)	Yes, if Customer_Info_Type _Code is ADDRESS and entering Ship To Address			
IS_BILL_TO_ADDRESS	VARCHAR2(1)	Yes, if Customer_Info_Type _Code is ADDRESS and entering Bill To Address			
IS_DELIVER_TO_ADDRESSES	VARCHAR2(1)	Yes, if Customer_Info_Type _Code is ADDRESS and entering Deliver To Address			

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SITE_NUMBE R	VARCHAR2(80)	Yes, if Customer_Info_Type _Code is ADDRESS and the customer should be added to an existing site, populate this field with that site_number.			
LOCATION_ NUMBER	VARCHAR2(40)				
NEW_PARTY _ID	NUMBER	internal use only			
NEW_PARTY _NUMBER	NUMBER	internal use only			
NEW_ACCO UNT_ID	NUMBER	internal use only			
NEW_ACCO UNT_NUMB ER	NUMBER	internal use only			
NEW_CONT ACT_ID	NUMBER	internal use only			
NEW_ADDR ESS_ID_SHIP	NUMBER	internal use only			
NEW_ADDR ESS_ID_BILL	NUMBER	internal use only			
NEW_ADDR ESS_ID_DELI VER	NUMBER	internal use only			
ATTRIBUTE_ CATEGORY	VARCHAR2(30)				

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ATTRIBUTE1.. 24	VARCHAR2(150)				
GLOBAL_tribute_category	VARCHAR2(30)				
GLOBAL_tribute1.. GLOBAL_tribute20	VARCHAR2(150)				
CREATION_DATE	DATE				
CREATED_BY	NUMBER				
LAST_UPDATE_DATE	DATE				
LAST_UPDATE_BY	NUMBER				
LAST_UPDATE_LOGIN	NUMBER				
REQUEST_ID	NUMBER				
ERROR_FLAG	VARCHAR2(1)				
REJECTED_FLAG	VARCHAR2(1)				
ORG_ID	NUMBER				

Table information:

OE_HEADERS_IFACE_ALL

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDER_SOURCE_ID	NUMBER		REQUIRED			
ORIGINAL_DOCUMENT_REF	VARCHAR2(50)		REQUIRED			
CHANGE_SEQUENCE	VARCHAR2(50)					X
CHANGE_REQUEST_CODE	VARCHAR2(30)					X
ORDER_SOURCE	VARCHAR2(30)		C			
ORG_ID	NUMBER				X	
HEADER_ID	NUMBER				X	
ORDER_NUMBER	NUMBER				X	
VERSION_NUMBER	NUMBER					X
ORDERED_DATE	DATE				X	
ORDER_CATEGORY	VARCHAR2(30)					

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDER_TY PE_ID	NUMBER		C			
ORDER_TY PE	VARCHAR 2(30)		C			
PRICE_LIST _ID	NUMBER			C		
PRICE_LIST	VARCHAR 2(30)			C		
CONVERSI ON_RATE	NUMBER		C			
CONVERSI ON_RATE_ DATE	DATE		C			
CONVERSI ON_TYPE_ CODE	VARCHAR 2(30)		C			
CONVERSI ON_TYPE	VARCHAR 2(30)		C			
TRANSACT IONAL_CU RR_CODE	VARCHAR 2(15)				X	
TRANSACT IONAL_CU RR	VARCHAR 2(30)				X	
SALESREP_ ID	NUMBER			X		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SALESREP	VARCHAR2(30)			X		
SALES_CH ANNEL_CODE	VARCHAR2(30)					X
RETURN_REASON_CODE	VARCHAR2(30)		C			
TAX_POINT_CODE	VARCHAR2(30)		For future use only			
TAX_POINT	VARCHAR2(30)					X
TAX_EXEMPT_FLAG	VARCHAR2(30)					X
TAX_EXEMPT_NUMBER	VARCHAR2(50)					X
TAX_EXEMPT_REASON_CODE	VARCHAR2(30)		C			
TAX_EXEMPT_REASON	VARCHAR2(30)		C			
AGREEMENT_ID	NUMBER					X
AGREEMENT	VARCHAR2(50)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
INVOICING_RULE_ID	NUMBER			X		
INVOICING_RULE	VARCHAR2(30)			X		
ACCOUNTING_RULE_ID	NUMBER			X		
ACCOUNTING_RULE	VARCHAR2(30)			X		
PAYMENT_TERM_ID	NUMBER			X		
PAYMENT_TERM	VARCHAR2(30)			X		
DEMAND_CLASS_CODE	VARCHAR2(30)					X
DEMAND_CLASS	VARCHAR2(30)					X
SHIPMENT_PRIORITY_CODE	VARCHAR2(30)					X
SHIPMENT_PRIORITY	VARCHAR2(30)					X
SHIPPING_METHOD_CODE	VARCHAR2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIPPING_METHOD	VARCHAR2(30)					X
FREIGHT_CARRIER_CODE	VARCHAR2(30)					X
FREIGHT_TERMS_CODE	VARCHAR2(30)					X
FREIGHT_TERMS	VARCHAR2(30)					X
FOB_POINT_CODE	VARCHAR2(30)					X
FOB_POINT	VARCHAR2(30)					X
PARTIAL_SHIPMENTS_ALLOWED	VARCHAR2(1)					X
SHIP_TOLERANCE_ABOVE	NUMBER					X
SHIP_TOLERANCE_BELOW	NUMBER					X
SHIPPING_INSTRUCTIONS	VARCHAR2(240)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
PACKING_INSTRUCTIONS	VARCHAR2(240)					X
ORDER_DATE_TYPE_CODE	VARCHAR2(30)					X
EARLIEST_SCHEDULE_LIMIT	NUMBER				X	
LATEST_SCHEDULE_LIMIT	NUMBER				X	
CUSTOMER_PO_NUMBER	VARCHAR2(50)					X
CUSTOMER_PAYMENT_TERM_ID						
CUSTOMER_PAYMENT_TERM						
PAYMENT_TYPE_CODE	VARCHAR2(30)					X
PAYMENT_AMOUNT	NUMBER					X
CHECK_NUMBER	VARCHAR2(50)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CREDIT_C ARD_CODE	VARCHAR 2(30)					X
CREDIT_C ARD_HOL DER_NAME	VARCHAR 2(50)					X
CREDIT_C ARD_NUM BER	VARCHAR 2(50)					X
CREDIT_C ARD_EXPIRATION_DATE	DATE					X
CREDIT_C ARD_APPR OVAL_CODE	VARCHAR 2(50)					X
SOLD_FROM M_ORG_ID	NUMBER			C		
SOLD_FROM M_ORG	VARCHAR 2(30)			C		
SOLD_TO_ ORG_ID	NUMBER			C		
SOLD_TO_ ORG	VARCHAR 2(30)			C		
SOLD_TO_ PARTY_ID	NUMBER					

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SOLD_TO_PARTY_NUMBER	VARCHAR2(30)					
SHIP_FROM_ORG_ID	NUMBER					X
SHIP_FROM_ORG	VARCHAR2(30)					X
SHIP_TO_ORG_ID	NUMBER					X
SHIP_TO_ORG	VARCHAR2(30)					X
INVOICE_TO_ORG_ID	NUMBER					X
INVOICE_TO_ORG	VARCHAR2(30)					X
DELIVER_TO_ORG_ID	NUMBER					X
DELIVER_TO_ORG	VARCHAR2(30)				X	
DELIVER_TO_ORG_CUSTOMER_NUMBER	VARCHAR2(30)				X	
DELIVER_TO_ORG_CUSTOMER	VARCHAR2(30)				X	

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SOLD_TO_CONTACT_ID	NUMBER					X
SOLD_TO_CONTACT	VARCHAR2(30)					X
SHIP_TO_CONTACT_ID	NUMBER					X
SHIP_TO_CONTACT	VARCHAR2(30)					X
INVOICE_TO_CONTACT_ID	NUMBER					X
INVOICE_TO_CONTACT	VARCHAR2(30)					X
DELIVER_TO_CONTACT_ID	NUMBER					X
DELIVER_TO_CONTACT	VARCHAR2(30)					X
CUSTOMER_ID	NUMBER					X
CUSTOMER_NAME	VARCHAR2(30)			C		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIPMENT_PRIORITY_CODE_INT	VARCHAR 2(30)					X
SHIP_TO_ADDRESS1	VARCHAR 2(30)			C		
SHIP_TO_ADDRESS2	VARCHAR 2(30)			C		
SHIP_TO_ADDRESS3	VARCHAR 2(30)			C		
SHIP_TO_ADDRESS4	VARCHAR 2(30)			C		
SHIP_TO_CITY	VARCHAR 2(30)			C		
SHIP_TO_CONTACT_FIRST_NAME	VARCHAR 2(30)					X
SHIP_TO_CONTACT_LAST_NAME	VARCHAR 2(30)					X
SHIP_TO_COUNTY	VARCHAR 2(30)			C		
SHIP_TO_CUSTOMER	VARCHAR 2(30)			C		
SHIP_TO_CUSTOMER_NUMBER	VARCHAR 2(30)			C		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIP_TO_POSTAL_CODE	VARCHAR2(30)			C		
SHIP_TO_PROVINCE	VARCHAR2(30)			C		
SHIP_TO_SITELINE_INT	VARCHAR2(30)				X	
SHIP_TO_STATE	VARCHAR2(30)			C		
SHIP_TO_COUNTRY	VARCHAR2(30)			C		
INVOICE_ADDRESS1	VARCHAR2(35)			C		
INVOICE_ADDRESS2	VARCHAR2(35)			C		
INVOICE_ADDRESS3	VARCHAR2(35)			C		
INVOICE_ADDRESS4	VARCHAR2(35)			C		
INVOICE_CITY	VARCHAR2(30)			C		
INVOICE_COUNTRY	VARCHAR2(20)			C		
INVOICE_COUNTY	VARCHAR2(25)			C		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
INVOICE_CUSTOMER	VARCHAR 2(60)			C		
INVOICE_CUSTOMER_NUMBER	VARCHAR 2(30)			C		
INVOICE_POSTAL_CODE	VARCHAR 2(15)			C		
INVOICE_PROVINCE_INTERNET	VARCHAR 2(30)			C		
INVOICE_SITE	VARCHAR 2(30)				X	
INVOICE_SITE_CODE	VARCHAR 2(30)				X	
INVOICE_STATUTE	VARCHAR 2(30)			C		
INVOICE_TRANSACTION_CONTACT_FIRST_NAME	VARCHAR 2(30)					X
INVOICE_TRANSACTION_CONTACT_LAST_NAME	VARCHAR 2(30)					X
ORDERED_BY_CONTACT_FIRST_NAME	VARCHAR 2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDERED_BY_CONTRACT_LAST_NAME	VARCHAR2(30)					X
DROP_SHIP_FLAG	VARCHAR2(1)					X
BOOKED_FLAG	VARCHAR2(1)					X
CLOSED_FLAG	VARCHAR2(1)					X
CANCELLED_FLAG	VARCHAR2(1)					X
REJECTED_FLAG	VARCHAR2(1)					X
CONTEXT	VARCHAR2(30)					X
ATTRIBUTE1..20	VARCHAR2(240)					X
HEADER_PO_CONTEXT	VARCHAR2(30)					X
PO_ATTRIBUTE1..20	VARCHAR2(240)					X
PO_REVISION_DATE	DATE					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
GLOBAL_A TTRIBUTE_ CATEGORY	VARCHAR 2(30)					X
GLOBAL_A TTRIBUTE1. .20	VARCHAR 2(240)					X
CREATED_ BY	NUMBER		REQUIRED			
CREATION_ _DATE	DATE		REQUIRED			
LAST_UPD ATED_BY	NUMBER		REQUIRED			
LAST_UPD ATE_DATE	DATE		REQUIRED			
LAST_UPD ATE_LOGI N	NUMBER					X
PROGRAM_ _APPLICAT ION_ID	NUMBER					X
PROGRAM_ _ID	NUMBER					X
PROGRAM_ _UPDATE_ DATE	DATE					X
REQUEST_I D	NUMBER				X	

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
REQUEST_DATE	DATE				X	
SUBMISSION_DATE_TIME	DATE					X
OPERATION_CODE	VARCHAR2(30)		REQUIRED			
ERROR_FLAG	VARCHAR2(1)					X
READY_FLAG	VARCHAR2(1)					X
STATUS_FLAG	VARCHAR2(1)					X
FORCE_APPLY_FLAG	VARCHAR2(1)					X
CHANGE_REASON	VARCHAR2(30)					X
CHANGE_COMMENTS	VARCHAR2(200)					X
TP_CONTEXT	VARCHAR2(30)					X
TP_ATTRIBUTES1..20	VARCHAR2(240)					X
BLANKET_NUMBER	NUMBER					

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORIG_SYS_CUSTOMER_REF	Varchar2(50)		Yes, if the customer is being created as Sold To for the order being imported			
ORIG_SHIP_ADDRESS_REF	Varchar2(50)		Yes, if Ship_To Address is being added for a customer.			
ORIG_BILL_ADDRESS_REF	Varchar2(50)		Yes, if Bill_To Address is being added for a customer			
ORIG_DELIVER_ADDRESS_REF	Varchar2(50)		Yes, if Deliver_To Address is being added for a customer			

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SOLD_TO_CONTACT_REF	Varchar2(50)		Yes, if Sold_To_Contact is being added for a customer			
SHIP_TO_CONTACT_REF	Varchar2(50)		Yes, if Ship_To_Contact is being added for a customer			
BILL_TO_CONTACT_REF	Varchar2(50)		Yes, if Bill_To_Contact is being added for a customer			
DELIVER_TO_CONTACT_REF	Varchar2(50)		Yes, if Deliver_To_Contact is being added for a customer			
XML_MESSAGE_ID	Varchar2(30)					X
XML_TRANSACTION_TYPE_CODE	varchar2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
TRANSACTION_PHASE_CODE	Varchar2(30)					X
SALES_DOCUMENT_NAME	Varchar2(240)					X
QUOTE_NUMBER	Number					X
QUOTE_DATE	Date					X
USER_STATUS_CODE	Varchar2(30)					X
EXPIRATION_DATE	Date					X
END_CUSTOMER_NAME	VARCHAR2(360)					X
END_CUSTOMER_NUMBER	VARCHAR2(50)					X
END_CUSTOMER_PARTY_NUMBER	VARCHAR2(30)					X
END_CUSTOMER_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
END_CUST OMER_PARTY_ID	NUMBER					X
END_CUST OMER_ORG_CONTACT_ID	NUMBER					X
END_CUST OMER_CONTACT_ID	NUMBER					X
END_CUST OMER_CONTACT	VARCHAR 2(360)					X
END_CUST OMER_LOCATION	VARCHAR 2(240)					X
END_CUST OMER_SITE_USE_ID	NUMBER					X
END_CUST OMER_PARTY_SITE_ID	NUMBER					X
END_CUST OMER_PARTY_SITE_USE_ID	NUMBER					X
END_CUST OMER_ADDRESS1	VARCHAR 2(240)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
END_CUST OMER_AD DRESS2	VARCHAR 2(240)					X
END_CUST OMER_AD DRESS3	VARCHAR 2(240)					X
END_CUST OMER_AD DRESS4	VARCHAR 2(240)					X
END_CUST OMER_CIT Y	VARCHAR 2(60)					X
END_CUST OMER_STA TE	VARCHAR 2(60)					X
END_CUST OMER_POS TAL_CODE	VARCHAR 2(60)					X
END_CUST OMER_CO UNTRY	VARCHAR 2(60)					X
END_CUST OMER_PRO VINCE	VARCHAR 2(60)					X
END_CUST OMER_CO UNTY	VARCHAR 2(60)					X
IB_OWNER	VARCHAR 2(60)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
IB_OWNER_CODE	VARCHAR 2(30)					X
IB_CURRENT_LOCATION	VARCHAR 2(60)					X
IB_CURRENT_LOCATION_CODE	VARCHAR 2(30)					X
IB_INSTALLATION_LOCATION	VARCHAR 2(60)					X
IB_INSTALLATION_LOCATION_CODE	VARCHAR 2(30)					X
SOLD_TO_PARTY_ID	NUMBER					X
SOLD_TO_ORG_CONTACT_ID	NUMBER					X
SHIP_TO_PARTY_ID	NUMBER					X
SHIP_TO_PARTY_SITE_ID	NUMBER					X
SHIP_TO_PARTY_SITE_USE_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
DELIVER_T O_PARTY_ID	NUMBER					X
DELIVER_T O_PARTY_ SITE_ID	NUMBER					X
DELIVER_T O_PARTY_ SITE_USE_ID	NUMBER					X
INVOICE_T O_PARTY_ID	NUMBER					X
INVOICE_T O_PARTY_ SITE_ID	NUMBER					X
INVOICE_T O_PARTY_ SITE_USE_ID	NUMBER					X
SHIP_TO_ORG_CONT ACT_ID	NUMBER					X
DELIVER_T O_ORG_CONTACT_ID	NUMBER					X
INVOICE_T O_ORG_CONTACT_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SOLD_TO_PARTY_NUMBER	VARCHAR2(30)					X
SHIP_TO_PARTY_NUMBER	VARCHAR2(30)					X
INVOICE_TO_PARTY_NUMBER	VARCHAR2(30)					X
DELIVER_TO_PARTY_NUMBER	VARCHAR2(30)					X

OE_HEADERS_IFACE_ALL Derived Values

- TRANSACTIONAL_CURR_CODE = FND_CURRENCIES.CURRENCY_CODE
- SOLD_FROM_ORG_ID = HR_ALL_ORGANIZATION_UNITS.ORGANIZATION_ID
- ACCOUNTING_RULE_ID = RA_RULES.RULE_ID
- INVOICING_RULE_ID = RA_RULES.RULE_ID
- SALESREP_ID = RA_SALESREPS_ALL.SALESREP_ID
- SALESREP = RA_SALESREPS_ALL.NAME
- PAYMENT_TERM_ID = RA_TERMS_B.TERM_ID
- CUSTOMER_PAYMENT_TERM_ID = RA_TERMS_B.TERM_ID
- PAYMENT_TERM = RA_TERMS_TL.NAME
- CUSTOMER_PAYMENT_TERM = RA_TERMS_TL.NAME

- AGREEMENT_ID = OE_AGREEMENTS_B.AGREEMENT_ID
- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- HEADER_ID = OE_ORDER_HEADERS_ALL.HEADER_ID
- PRICE_LIST_ID = QP_LIST_HEADERS_TL.LIST_HEADER_ID
- PRICE_LIST = QP_LIST_HEADERS_TL.NAME

OE_HEADERS_IFACE_ALL Conditional Settings

Column Name	Conditional Setting requirement
ORDER_SOURCE_ID ORDER_SOURCE	Condition is that either one of the columns should be populated
ORDER_TYPE_ID ORDER_TYPE	Condition is that either one of the columns should be populated
CONVERSION_RATE CONVERSION_RATE_DATE	Condition is that either one of the columns should be populated
CONVERSION_TYPE_CODE CONVERSION_TYPE	Condition is that either one of the columns should be populated
TAX_EXEMPT_REASON_CODE TAX_EXEMPT_REASON	Condition is that either one of the columns should be populated
RETURN_REASON_CODE	Required for returns only
PRICE_LIST_ID PRICE_LIST	Condition is that either one of the columns should be populated
SOLD_FROM_ORG_ID SOLD_FROM_ORG	Condition is that either one of the columns should be populated

Column Name	Conditional Setting requirement
SOLD_TO_ORG_ID	Condition is that Sold_To_Org_ID and Customer_Name are required.
SOLD_TO_ORG	
CUSTOMER_NAME	
SHIP_TO_ADDRESS1..4	These columns or <i>Ship_to_Org_id</i> should be provided
SHIP_TO_CITY	
SHIP_TO_COUNTY	
SHIP_TO_CUSTOMER	
SHIP_TO_CUSTOMER_NUMBER	
SHIP_TO_POSTAL_CODE	
SHIP_TO_PROVINCE	
SHIP_TO_SITE_INT	
SHIP_TO_STATE	
SHIP_TO_COUNTRY	
INVOICE_ADDRESS1..4	These columns or <i>Invoice_To_Org_id</i> should be provided.
INVOICE_CITY	
INVOICE_COUNTRY	
INVOICE_COUNTY	
INVOICE_CUSTOMER	
INVOICE_CUSTOMER_NUMBER	
INVOICE_POSTAL_CODE	
INVOICE_PROVINCE_INT	
INVOICE_STATE	

Column Name	Conditional Setting requirement
DELIVER_TO_ADDRESS1..4	These columns or <i>Deliver_To_Org_id</i> should be provided.
DELIVER_TO_CITY	
DELIVER_TO_COUNTRY	
DELIVER_TO_COUNTY	
DELIVER_TO_CUSTOMER	
DELIVER_TO_CUSTOMER_NUMBER	
DELIVER_TO_POSTAL_CODE	
DELIVER_TO_PROVINCE_INT	
DELIVER_TO_STATE	

Note: Attribute columns are for customer use only and will NEVER be used by Oracle Development as a means to enable any type of system functionality.

Table:

OE_LINES_IFACE_ALL

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDER_SOURCE_ID	NUMBER		REQUIRED			
ORIG_SYS_DOCUMENT_REF	VARCHAR2(50)		REQUIRED			
ORIG_SYS_LINE_REF	VARCHAR2(50)		REQUIRED			

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORIG_SYS_SHIPMENT_REF	VARCHAR2(50)		REQUIRED			
CHANGE_SEQUENCE	VARCHAR2(50)					X
CHANGE_REQUEST_CODE	VARCHAR2(30)					X
ORG_ID	NUMBER				X	
LINE_NUMBER	NUMBER				X	
SHIPMENT_NUMBER	NUMBER				X	
LINE_ID	NUMBER				X	
SPLIT_FROM_LINE_ID	NUMBER					X
LINE_TYPE_ID	NUMBER				X	
LINE_TYPE	VARCHAR2(30)				X	
ITEM_TYPE_CODE	VARCHAR2(30)				X	X
INVENTORY_ITEM_ID	NUMBER		C			

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
INVENTORY_ITEM	VARCHAR2(30)		REQUIRED			
TOP_MODEL_LINE_REF	VARCHAR2(50)		C			
LINK_TO_LINE_REF	VARCHAR2(50)		C			
EXPLOSION_DATE	DATE				X	
ATO_LINE_ID	NUMBER				X	
COMPONENT_SEQUENCE_ID	NUMBER				X	
COMPONENT_CODE	VARCHAR2(50)				X	X
SORT_ORDER	VARCHAR2(240)				X	
MODEL_GROUP_NUMBER	NUMBER				X	
OPTION_NUMBER	NUMBER				X	
OPTION_FLAG	VARCHAR2(1)					X

Column Name	Type	Description	Required (C = Condition ally Required)	Condition ally Required for Booking	Derived	Optional
SHIP_MO DEL_COM PLETE_FL AG	VARCHAR 2(1)				X	
SOURCE_T YPE_COD E	VARCHAR 2(30)					X
SCHEDUL E_STATUS _CODE	VARCHAR 2(30)			C		
SCHEDUL E_SHIP_D ATE	DATE			C		
SCHEDUL E_ARRIVA L_DATE	DATE			C		
ACTUAL_ ARRIVAL_ DATE	DATE					X
REQUEST_ DATE	DATE		REQUIRE D			
PROMISE_ DATE	DATE			C		X
SCHEDUL E_DATE	DATE					X
DELIVERY _LEAD_TI ME	NUMBER		REQUIRE D			

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
DELIVERY_ID	NUMBER		REQUIRED			
ORDERED_QUANTITY	NUMBER		REQUIRED			
ORDER_QUANTITY_UOM	VARCHAR2(3)		REQUIRED			
SHIPPING_QUANTITY	NUMBER		C			
SHIPPING_QUANTITY_UOM	VARCHAR2(3)		C			
SHIPPED_QUANTITY	NUMBER		C			
CANCELLED_QUANTITY	NUMBER		C			
FULFILLED_QUANTITY	NUMBER		C			
PRICING_QUANTITY	NUMBER		C			
PRICING_QUANTITY_UOM	VARCHAR2(3)		C			

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SOLD_FR OM_ORG_ ID	NUMBER			C		
SOLD_FR OM_ORG	VARCHAR 2(30)			C		
SOLD_TO_ ORG_ID	NUMBER			C		
SOLD_TO_ ORG	VARCHAR 2(30)			C		
SHIP_FRO M_ORG_I D	NUMBER					X
SHIP_FRO M_ORG	VARCHAR 2(30)					X
SHIP_TO_ ORG_ID	NUMBER					X
SHIP_TO_ ORG	VARCHAR 2(30)					X
DELIVER_ TO_ORG_I D	NUMBER				X	
DELIVER_ TO_ORG	VARCHAR 2(30)				X	
INVOICE_ TO_ORG_I D	NUMBER			C		
INVOICE_ TO_ORG	VARCHAR 2(30)			C		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIP_TO_ADDRESS1	VARCHAR 2(30)			C		
SHIP_TO_ADDRESS2	VARCHAR 2(30)			C		
SHIP_TO_ADDRESS3	VARCHAR 2(30)			C		
SHIP_TO_ADDRESS4	VARCHAR 2(30)			C		
SHIP_TO_CITY	VARCHAR 2(30)			C		
SHIP_TO_COUNTY	VARCHAR 2(30)			C		
SHIP_TO_STATE	VARCHAR 2(30)			C		
SHIP_TO_POSTAL_CODE	VARCHAR 2(30)			C		
SHIP_TO_COUNTRY	VARCHAR 2(30)			C		
SHIP_TO_CONTACT_FIRST_NAME	VARCHAR 2(30)					X
SHIP_TO_CONTACT_LAST_NAME	VARCHAR 2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIP_TO_CONTACT_JOB_TITLE	VARCHAR 2(30)					X
SHIP_TO_CONTACT_AREA_CODE1	VARCHAR 2(10)					X
SHIP_TO_CONTACT_AREA_CODE2	VARCHAR 2(10)					X
SHIP_TO_CONTACT_AREA_CODE3	VARCHAR 2(10)					X
SHIP_TO_CONTACT_ID	NUMBER					X
SHIP_TO_CONTACT	VARCHAR 2(30)					X
DELIVER_TO_CONTACT_ID	NUMBER					X
DELIVER_TO_CONTACT_ACT	VARCHAR 2(30)					X
INVOICE_TO_CONTACT_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
INVOICE_ TO_CONT ACT	VARCHAR 2(30)					X
DROP_SHI P_FLAG	VARCHAR 2(1)					X
VEH_CUS_ ITEM_CU M_KEY_ID	NUMBER					X
CUST_PRO DUCTION _SEQ_NU M	NUMBER					X
LOAD_SE Q_NUMBE R	NUMBER					X
OVER_SHI P_REASO N_CODE	VARCHAR 2(30)					X
OVER_SHI P_RESOLV ED_FLAG	VARCHAR 2(1)					X
AUTHORI ZED_TO_S HIP_FLAG	VARCHAR 2(1)					X
SHIP_TOL ERANCE_ ABOVE	NUMBER					X
SHIP_TOL ERANCE_ BELOW	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIP_SET_ID	NUMBER					X
SHIP_SET_NAME	VARCHAR2(30)					X
ARRIVAL_SET_ID	NUMBER					X
ARRIVAL_SET_NAME	VARCHAR2(30)					X
INVOICE_SET_ID	NUMBER					X
INVOICE_SET_NAME	VARCHAR2(30)					X
FULFILLMENT_SET_ID	NUMBER					X
FULFILLMENT_SET_NAME	VARCHAR2(30)					X
PRICE_LIST_ID	NUMBER			C		
PRICE_LIST	VARCHAR2(30)			C		
PRICING_DATE	DATE			C		
UNIT_LIST_PRICE	NUMBER			C		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
UNIT_LIST_PRICE_PQTY	NUMBER					
UNIT_SEL_LING_PRICE	NUMBER			C		
UNIT_SEL_LING_PRICE_PQTY	NUMBER					
CALCULATE_PRICE_FLAG	VARCHAR2(1)					X
TAX_CODE	VARCHAR2(50)					X
TAX	VARCHAR2(50)					X
TAX_VALUE	NUMBER					X
TAX_DATE	DATE					X
TAX_POINT_CODE	VARCHAR2(30)					For future use
TAX_POINT	VARCHAR2(30)					X
TAX_EXEMPT_FLAG	VARCHAR2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
TAX_EXE MPT_NUMBER	VARCHAR 2(50)					X
TAX_EXE MPT_REASSON_CODE	VARCHAR 2(30)					X
TAX_EXE MPT_REASSON	VARCHAR 2(30)					X
AGREEMENT_ID	NUMBER					X
AGREEMENT	VARCHAR 2(30)					X
INVOICING_RULE_ID	NUMBER			C		
INVOICING_RULE	VARCHAR 2(30)			C		
ACCOUNTING_RULE_ID	NUMBER			C		
ACCOUNTING_RULE	VARCHAR 2(30)			C		
PAYMENT_TERM_ID	NUMBER			C		
PAYMENT_TERM	VARCHAR 2(30)			C		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
DEMAND_CLASS_CODE	VARCHAR2(30)					X
DEMAND_CLASS	VARCHAR2(30)			C		X
SHIPMENT_PRIORITY_CODE	VARCHAR2(30)					X
SHIPMENT_PRIORITY	VARCHAR2(30)					X
SHIPPING_METHOD_CODE	VARCHAR2(30)					X
SHIPPING_METHOD	VARCHAR2(30)					X
SHIPPING_INSTRUCTIONS	VARCHAR2(240)					X
PACKING_INSTRUCTIONS	VARCHAR2(240)					X
FREIGHT_CARRIER_CODE	VARCHAR2(30)					X
FREIGHT_TERMS_CODE	VARCHAR2(30)					X

Column Name	Type	Description	Required (C = Condition ally Required)	Condition ally Required for Booking	Derived	Optional
FREIGHT_ TERMS	VARCHAR 2(30)					X
FOB_POIN T_CODE	VARCHAR 2(30)					X
FOB_POIN T	VARCHAR 2(30)					X
SALESREP _ID	NUMBER				X	
SALESREP	VARCHAR 2(30)			X		
RETURN_ REASON_ CODE	VARCHAR 2(30)		C			
REFEREN CE_TYPE	VARCHAR 2(30)		C			
REFEREN CE_HEAD ER_ID	NUMBER		C			
REFEREN CE_HEAD ER	VARCHAR 2(30)		C			
REFEREN CE_LINE_I D	NUMBER		C			
REFEREN CE_LINE	VARCHAR 2(30)		C			

Column Name	Type	Description	Required (C = Condition ally Required)	Condition ally Required for Booking	Derived	Optional
CREDIT_I NVOICE_L INE_ID	NUMBER					X
CUSTOMER_PO_NUMBER	VARCHAR 2(50)					X
CUSTOMER_LINE_NUMBER	VARCHAR 2(50)					X
CUSTOMER_SHIPMENT_NUMBER	VARCHAR 2(50)					X
CUSTOMER_ITEM_ID	NUMBER					X
CUSTOMER_ITEM_ID_TYPE	VARCHAR 2(30)					X
CUSTOMER_ITEM_NAME	VARCHAR 2(200)					X
CUSTOMER_ITEM_REVISION	VARCHAR 2(50)					X
CUSTOMER_ITEM_NET_PRICE	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CUSTOMER_PAYMENTS_TERM_ID						
CUSTOMER_PAYMENTS_TERM						
DEMAND_BUCKET_TYPE_CODE	VARCHAR2(30)					X
DEMAND_BUCKET_TYPE	VARCHAR2(50)					X
SCHEDULE_ITEM_DETAILS	VARCHAR2(30)					X
DEMAND_STREAM	VARCHAR2(30)					X
CUSTOMER_DOCK_CODE	VARCHAR2(30)					X
CUSTOMER_DOCK	VARCHAR2(50)					X
CUSTOMER_JOB	VARCHAR2(50)					X
CUSTOMER_PRODUCT_LINE	VARCHAR2(50)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CUST_MO DEL_SER AL_NUMB ER	VARCHAR 2(50)					X
PROJECT_I D	NUMBER					X
PROJECT	VARCHAR 2(30)					X
TASK_ID	NUMBER					X
TASK	VARCHAR 2(30)					X
END_ITE M_UNIT_ NUMBER	VARCHAR 2(30)					X
ITEM_REV ISION	VARCHAR 2(3)					X
SERVICE_ DURATIO N	NUMBER		C			
SERVICE_ START_D ATE	DATE		C			
SERVICE_ END_DAT E	DATE		C			
SERVICE_ COTERMI NATE_FL AG	VARCHAR 2(1)		C			

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
UNIT_SEL LING_PERCENT	NUMBER				X	
UNIT_LIST _PERCENT	NUMBER				X	
UNIT_PER CENT_BASE PRICE	NUMBER				X	
SERVICE_ NUMBER	NUMBER				X	
SERVICED _LINE_ID	NUMBER				X	
FULFILLED_FLAG	VARCHAR 2(1)					X
CLOSED_FLAG	VARCHAR 2(1)					X
CANCELLED_FLAG	VARCHAR 2(1)					X
REJECTED_FLAG	VARCHAR 2(1)					X
CONTRACT_PO_NUMBER	VARCHAR 2(150)					X
LINE_PO_CONTEXT	VARCHAR 2(30)					X
ATTRIBUTE1..20	VARCHAR 2(240)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
INDUSTRY_CONTEXT	VARCHAR2(30)					X
INDUSTRY_ATTRIBUTE1..30	VARCHAR2(240)					X
PRICING_CONTEXT	VARCHAR2(150)					X
PRICING_ATTRIBUTE1..10	VARCHAR2(240)					X
PRICING_ATTRIBUTE10	VARCHAR2(240)					X
GLOBAL_ATTRIBUTE_CATEGORY	VARCHAR2(30)					X
GLOBAL_ATTRIBUTE1..20	VARCHAR2(240)					X
RETURN_ATTRIBUTE1..20	VARCHAR2(240)					X
INVENTORY_ITEM_SEGMENT_1...20	VARCHAR2(240)					X
SERVICE_CONTEXT	VARCHAR2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SERVICE_ATTRIBUTES1..20	VARCHAR2(240)					X
CREATED_BY	NUMBER		REQUIRED			
CREATION_DATE	DATE		REQUIRED			
LAST_UPDATED_BY	NUMBER		REQUIRED			
LAST_UPDATE_DATE	DATE					X
LAST_UPDATE_LOGIC	NUMBER					X
PROGRAM_APPLICATION_ID	NUMBER					X
PROGRAM_ID	NUMBER					X
PROGRAM_UPDATE_DATE	DATE					X
REQUEST_ID	NUMBER		REQUIRED			
OPERATION_CODE	VARCHAR2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ERROR_FLAG	VARCHAR 2(1)					X
STATUS_FLAG	VARCHAR 2(1)					X
CHANGE_REASON	VARCHAR 2(30)					X
CHANGE_COMMENTS	VARCHAR 2(2000)					X
CONFIG_HEADER_ID	NUMBER				X	X
CONFIG_REVENUE_NBR	NUMBER				X	X
CONFIGURATION_ID	NUMBER				X	X
SERVICE_TRANSACTION_REASON_CODE	VARCHAR 2(30)					X
SERVICE_TRANSACTION_COMMENTS	VARCHAR 2(2000)					X
BLANKET_NUMBER	NUMBER					
BLANKET_LINE_NUMBER	NUMBER					

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
OVERRIDE _ATP_DATA E_CODE	Varchar2(30)		No	No	No	Y Comment: Based on the Ordered Date Type, either of the dates should be populated to override the line.
LATE_DE MAND_PENALTY_FACTOR	Number		No	No	No	Y Values should be greater than 0.
SHIP_TO_CUSTOMER_NAME	Varchar2(360)					
SHIP_TO_CUSTOMER_NUMBER	Varchar2(30)					
INVOICE_TO_CUSTOMER_NAME	Varchar2(360)					
INVOICE_TO_CUSTOMER_NUMBER	Varchar2(30)					

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
DELIVER_TO_CUSTOMER_NAME	Varchar2(360)					
DELIVER_TO_CUSTOMER_NUMBER	Varchar2(30)					
DELIVER_TO_ADDRESS1	Varchar2(240)					
DELIVER_TO_ADDRESS2	Varchar2(240)					
DELIVER_TO_ADDRESS3	Varchar2(240)					
DELIVER_TO_ADDRESS4	Varchar2(240)					
DELIVER_TO_CITY	Varchar2(60)					
DELIVER_TO_COUNTRY	Varchar2(60)					
DELIVER_TO_COUNTRY	Varchar2(60)					
DELIVER_TO_STATE	Varchar2(60)					

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
DELIVER_TO_PROVINCE	Varchar2(60)					
DELIVER_TO_POSTAL_CODE	Varchar2(60)					
ORIG_SHIP_ADDRESS_REF	Varchar2(50)		Yes, if	Ship_To_Address is being added for the customer at the line level		
ORIG_BILL_ADDRESS_REF	Varchar2(50)		Yes, if	Bill_To_Address is being added for the customer at the line level		
ORIG_DELIVER_ADDRESS_REF	Varchar2(50)		Yes, if	Deliver_To_Address is being added for the customer at the line level		

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIP_TO_CONTACT_REF	Varchar2(50)		Yes, if Ship_To_Contact is being added for the customer at the line level			
BILL_TO_CONTACT_REF	Varchar2(50)		Yes, if Bill_To_Contact is being added for the customer at the line level			
DELIVER_TO_CONTACT_REF	Varchar2(50)		Yes, if Deliver_To_Contact is being added for the customer at the line level			
XML_TRANSACTION_TYPE_CODE						
SHIP_TO_PARTY_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SHIP_TO_PARTY_SITE_ID	NUMBER					X
SHIP_TO_PARTY_SITE_USE_ID	NUMBER					X
DELIVER_TO_PARTY_ID	NUMBER					X
DELIVER_TO_PARTY_SITE_ID	NUMBER					X
DELIVER_TO_PARTY_SITE_USE_ID	NUMBER					X
INVOICE_TO_PARTY_ID	NUMBER					X
INVOICE_TO_PARTY_SITE_ID	NUMBER					X
INVOICE_TO_PARTY_SITE_USE_ID	NUMBER					X
SHIP_TO_ORG_CONTACT_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
DELIVER_ TO_ORG_ CONTACT_ _ID	NUMBER					X
INVOICE_ TO_ORG_ CONTACT_ _ID	NUMBER					X
SHIP_TO_ PARTY_N UMBER	VARCHAR 2(30)					X
INVOICE_ TO_PART Y_NUMBE R	VARCHAR 2(30)					X
DELIVER_ TO_PART Y_NUMBE R	VARCHAR 2(30)					X
END_CUS TOMER_N AME	VARCHAR 2(360)					X
END_CUS TOMER_N UMBER	VARCHAR 2(50)					X
END_CUS TOMER_P ARTY_NU MBER	VARCHAR 2(30)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
END_CUSTOMER_ID	NUMBER					X
END_CUSTOMER_PARTY_ID	NUMBER					X
END_CUSTOMER_ORG_CONTACT_ID	NUMBER					X
END_CUSTOMER_CONTACT_ID	NUMBER					X
END_CUSTOMER_CONTACT	VARCHAR2(360)					X
END_CUSTOMER_LOCATION	VARCHAR2(240)					X
END_CUSTOMER_SITE_ID	NUMBER					X
END_CUSTOMER_PARTY_SITE_ID	NUMBER					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
END_CUSTOMER_PARTY_USE_ID	NUMBER					X
END_CUSTOMER_ADDRESS1	VARCHAR2(240)					X
END_CUSTOMER_ADDRESS2	VARCHAR2(240)					X
END_CUSTOMER_ADDRESS3	VARCHAR2(240)					X
END_CUSTOMER_ADDRESS4	VARCHAR2(240)					X
END_CUSTOMER_CITY	VARCHAR2(60)					X
END_CUSTOMER_STATE	VARCHAR2(60)					X
END_CUSTOMER_POSTAL_CODE	VARCHAR2(60)					X
END_CUSTOMER_COUNTRY	VARCHAR2(60)					X

Column Name	Type	Description	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
END_CUSTOMER_PROVINCE	VARCHAR2(60)					X
END_CUSTOMER_COUNTY	VARCHAR2(60)					X
IB_OWNER	VARCHAR2(60)					X
IB_CURRENT_LOCATION	VARCHAR2(60)					X
IB_INSTALL_LOCATION	VARCHAR2(60)					X
IB_OWNER_CODE	Varchar2(30)	Install Base Owner Identifier				
IB_INSTALL_LOCATION_CODE	Varchar2(30)	Install Location Identifier				
IB_CURRENT_LOCATION_CODE	Varchar2(30)	Install Current Location Identifier				

OE_LINES_IFACE_ALL Derived Values

- AGREEMENT_ID = OE_AGREEMENTS_TL.AGREEMENT_ID
- SHIP_FROM_ORG_ID =
HR_ALL_ORGANIZATION_UNITS.ORGANIZATION_ID

- SOLD_FROM_ORG_ID =
HR_ALL_ORGANIZATION_UNITS.ORGANIZATION_ID
- ACCOUNTING_RULE_ID = RA_RULES.RULE_ID
- INVOICING_RULE_ID = RA_RULES.RULE_ID
- SALESREP_ID = RA_SALESREPS_ALL.SALESREP_ID
- PRICE_LIST_ID = QP_LIST_HEADERS_TL.LIST_HEADER_ID
- PAYMENT_TERM_ID = RA_TERMS_B.TERM_ID
- CUSTOMER_PAYMENT_TERM_ID = RA_TERMS_B.TERM_ID
- SALESREP = RA_SALESREPS_ALL.NAME
- CUSTOMER_PAYMENT_TERM = RA_TERMS_TL.NAME
- PAYMENT_TERM = RA_TERMS_TL.NAME
- PRICE_LIST = QP_LIST_HEADERS_TL.NAME
- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DOCUMENT_REF =
- OE_HEADERS_IFACE_ALL.ORIG_SYS_DOCUMENT_REF
- LINK_TO_LINE_REF = OE_LINES_IFACE_ALL.ORIG_SYS_LINE_REF
- TOP_MODEL_LINE_REF = OE_LINES_IFACE_ALL.ORIG_SYS_LINE_REF

OE_LINES_IFACE_ALL Conditional Settings

Column Name	Conditional Setting requirement
INVENTORY_ITEM_ID INVENTORY_ITEM	Condition is that either column should be populated
TOP_MODEL_LINE_REF LINK_TO_LINE_REF	Required for model items only

Column Name	Conditional Setting requirement
SHIPPING_QUANTITY SHIPPING_QUANTITY_UOM	Condition is that both columns should be populated, if populated since they are not required.
PRICING_QUANTITY PRICING_QUANTITY_UOM	Condition is that both columns should be populated, if populated since they are not required.
RETURN_REASON_CODE	Only required for returns.
REFERENCE_TYPE REFERENCE_HEADER_ID REFERENCE_HEADER REFERENCE_LINE_ID REFERENCE_LINE	Reference_Type should be there to populate either reference_header_id/reference_header and reference_line_id/reference_line (either id or value column should be populated; not both)
SERVICE_DURATION SERVICE_START_DATE SERVICE_END_DATE SERVICE_COTERMINATE_FLAG	Should be populated only for service items
SOLD_FROM_ORG_ID SOLD_FROM_ORG	Condition is that either column should be populated
SOLD_TO_ORG_ID SOLD_TO_ORG	Condition is that either column should be populated
INVOICE_TO_ORG_ID INVOICE_TO_ORG	Condition is that either column should be populated

Column Name	Conditional Setting requirement
SHIP_TO_ADDRESS1..4	These columns or Ship_to_Org_id should be present.
SHIP_TO_CITY	
SHIP_TO_COUNTY	
SHIP_TO_STATE	
SHIP_TO_POSTAL_CODE	
SHIP_TO_COUNTRY	
PRICE_LIST_ID	Condition is that either column should be populated
PRICE_LIST	
ACCOUNTING_RULE_ID	Condition is that either column should be populated
ACCOUNTING_RULE	
PAYMENT_TERM_ID	Condition is that either column should be populated
PAYMENT_TERM	
DEMAND_CLASS_CODE	Condition is that either column should be populated, if populated since they are not required.
DEMAND_CLASS	
SHIP_TO_CUSTOMER_NAME	
SHIP_TO_CUSTOMER_NUMBER	
INVOICE_TO_CUSTOMER_NAME	
INVOICE_TO_CUSTOMER_NUMBER	
DELIVER_TO_CUSTOMER_NAME	
DELIVER_TO_CUSTOMER_NUMBER	

Column Name	Conditional Setting requirement
DELIVER_TO_ADDRESS1	
DELIVER_TO_ADDRESS2	
DELIVER_TO_ADDRESS3	
DELIVER_TO_ADDRESS4	
DELIVER_TO_CITY	
DELIVER_TO_COUNTY	
DELIVER_TO_COUNTRY	
DELIVER_TO_STATE	
DELIVER_TO_PROVINCE	
DELIVER_TO_POSTAL_CODE	

Note: Attribute columns are for customer use only and will NEVER be used by Oracle Development as a means to enable any type of system functionality.

OE_PRICE_ADJS_IFACE_ALL

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
AC_CONTEXT	VARCHAR2(150)				
AC_ATTRIBUTE1...15	VARCHAR2(240)				
ATTRIBUTE1...15	VARCHAR2(240)				X
ADJUSTED_AMOUNT	VARCHAR2()				X

Column Name	Type	Required (C = Conditionall y Required)	Conditionall y Required for Booking	Derived	Optional
ADJUSTED_ AMOUNT_P ER_PQTY	VARCHAR2()				X
APPLIED_FL AG	VARCHAR2(1)				X
ARITHMETI C_OPERATO R	VARCHAR2()				X
AUTOMATI C_FLAG	VARCHAR2(1)				X
CHANGE_R EQUEST_CO DE	VARCHAR2(30)				X
CHANGE_SE QUENCE	VARCHAR2(50)				X
CHANGE_R EASON_CO DE	VARCHAR2(30)				
CHANGE_R EASON_TEX T	VARCHAR2(2000)				
CHARGE_SU BTYPE_COD E	VARCHAR2(30)				
CHARGE_TY PE_CODE	VARCHAR2(30)				
CONTEXT	VARCHAR2(30)				X

Column Name	Type	Required (C = Conditionall y Required)	Conditionall y Required for Booking	Derived	Optional
COST_ID	NUMBER				
CREATED_BY	NUMBER	REQUIRED			
CREATION_DATE	DATE	REQUIRED			
CREDIT_OR_CHARGE_FLAG	VARCHAR2(1)				
DISCOUNT_ID	NUMBER				
DISCOUNT_LINE_ID	NUMBER				
DISCOUNT_NAME	VARCHAR2(30)				
ERROR_FLAG	VARCHAR2(1)				X
ESTIMATED_FLAG	VARCHAR2(1)				
INC_IN_SALES_PERFORMANCE	VARCHAR2(1)				
INCLUDE_ON_RETURNS_FLAG	VARCHAR2(1)				
INTERFACE_STATUS	VARCHAR2(1000)				

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
INVOICED_F LAG	VARCHAR2(1)				
LAST_UPDA TE_DATE	DATE	REQUIRED			
LAST_UPDA TE_LOGIN	NUMBER				X
LAST_UPDA TED_BY	NUMBER	REQUIRED			
LIST_HEADE R_ID	NUMBER				
LIST_LINE_I D	NUMBER				
LIST_LINE_ NUMBER	NUMBER				
LIST_LINE_T YPE_CODE	VARCHAR2(30)				
LIST_NAME	VARCHAR2(240)				
MODIFIED_F ROM	VARCHAR2(240)				
MODIFIED_ TO	VARCHAR2(240)				
MODIFIER_ MECHANIS M_TYPE_CO DE	VARCHAR2(30)				

Column Name	Type	Required (C = Conditionall y Required)	Conditionall y Required for Booking	Derived	Optional
MODIFIER_ NAME	VARCHAR2(240)				
OPERAND	NUMBER				
OPERAND_P ER_PQTY	NUMBER				
OPERATION _CODE	VARCHAR2(30)	REQUIRED			
ORDER_SOU RCE_ID	NUMBER				
ORG_ID	NUMBER			X	
ORIG_SYS_D ISCOUNT_R EF	VARCHAR2(50)	REQUIRED			
ORIG_SYS_D OCUMENT_ REF	VARCHAR2(50)	REQUIRED			
ORIG_SYS_LI NE_REF	VARCHAR2(50)	Conditional or optional (Required only if line level adjustment/cr edit)			
ORIG_SYS_S HIPMENT_R EF	VARCHAR2(50)	OPTIONAL			
PARENT_AD JUSTMENT_I D	NUMBER				

Column Name	Type	Required (C = Conditionall y Required)	Conditionall y Required for Booking	Derived	Optional
PERCENT	NUMBER				X
PRICING_PH ASE_ID	NUMBER				
PROGRAM_ APPLICATION_ ID	NUMBER				X
PROGRAM_I D	NUMBER				X
PROGRAM_ UPDATE_DATE	DATE				X
REQUEST_ID	NUMBER				X
SOLD_TO_O RG	VARCHAR2(360)				X
SOLD_TO_O RG_ID	NUMBER				X
STATUS_FL AG	VARCHAR2(1)				X
TAX_CODE	VARCHAR2(50)				
UPDATE_AL LOWED	VARCHAR2(1)				
UPDATED_F LAG	VARCHAR2(1)				
VERSION_N UMBER	VARCHAR2(30)				

OE_PRICE_ADJS_IFACE_ALL Derived Values

- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DISCOUNT_REF
=OE_ORDER_HEADERS_ALL.ORIG_SYS_DOCUMENT_REF
- ORIG_SYS_LINE_REF = OE_ORDER_LINES_ALL.ORIG_SYS_LINE_REF
- ORIG_SYS_SHIPMENT_REF =
OE_ORDER_LINES_ALL.ORIG_SYS_SHIPMENT_REF
- LIST_HEADER_ID = QP_LIST_HEADERS_B.LIST_HEADER_ID
- LIST_NAME = QP_LIST_HEADERS_TL.NAME
- LIST_LINE_ID = QP_LIST_LINES.LIST_LINE_ID

OE_PRICE_ATTS_IFACE_ALL

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ATTRIBUTE1.. 15	VARCHAR2(240)				X
CHANGE_RE QUEST_CODE	VARCHAR2(30)				X
CHANGE_SE QUENCE	VARCHAR2(50)				X
CONTEXT	VARCHAR2(30)				X
CREATED_BY	NUMBER	REQUIRED			
CREATION_DATE	DATE	REQUIRED			

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CREDIT_OR_CHARGE_FL AG	VARCHAR2(1)				
ERROR_FLAG	VARCHAR2(1)				X
FLEX_TITLE					
INTERFACE_STATUS	VARCHAR2(1000)				
LAST_UPDATE_DATE	DATE	REQUIRED			
LAST_UPDATE_LOGIN	NUMBER				X
LAST_UPDATED_BY	NUMBER	REQUIRED			
OPERATION_CODE	VARCHAR2(30)	REQUIRED			
ORDER_SOURCE_ID					
ORG_ID	NUMBER			X	
ORIG_SYS_ATT_REF	VARCHAR2(50)	REQUIRED			
ORIG_SYS_DOCUMENT_REF	VARCHAR2(50)	REQUIRED			

Column Name	Type	Required (C = Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORIG_SYS_LINE_REF	VARCHAR2(50)	Conditional or optional (Required only if line level adjustment/credit)			
ORIG_SYS_SHIPMENT_REF	VARCHAR2(50)	OPTIONAL			
PRICING_ATTRIBUTES1...100					
PROGRAM_APPLICATION_ID	NUMBER				X
PROGRAM_ID	NUMBER				X
PROGRAM_UPDATE_DATE	DATE				X
REQUEST_ID	NUMBER				X
SOLD_TO_ORG	VARCHAR2(360)				X
SOLD_TO_ORG_ID	NUMBER				X
STATUS_FLAG	VARCHAR2(1)				X

OE_PRICE_ATTS_IFACE_ALL Derived Values

- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DOCUMENT_REF =

OE_ORDER_HEADERS_ALL.Orig_Sys_Document_Ref

- ORIG_SYS_LINE_REF = OE_ORDER_LINES_ALL.Orig_Sys_Line_Ref
- ORIG_SYS_SHIPMENT_REF =
OE_ORDER_LINES_ALL.Orig_Sys_Shipment_Ref

OE_CREDITS_IFACE_ALL

Column Name	Type	Required (C= Conditionall y Required)	Conditionall y Required for Booking	Derived	Optional
ORDER_SO URCE_ID	NUMBER	REQUIRED			
ORIG_SYS_ DOCUMENT_ T_REF	VARCHAR2 (50)	REQUIRED			
ORIG_SYS_L INE_REF	VARCHAR2 (50)	Conditional or optional (Required only if line level adjustment/c redit)			
ORIG_SYS_S HIPMENT_ REF	VARCHAR2 (50)	OPTIONAL			
ORIG_SYS_ CREDIT_RE F	VARCHAR2 (50)	REQUIRED			
CHANGE_S EQUENCE	VARCHAR2 (50)				X
CHANGE_R EQUEST_C ODE	VARCHAR2 (30)				X
ORG_ID	NUMBER			X	

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
SALESREP_ID	NUMBER			X	
SALESREP	VARCHAR2 (30)			X	
SALES_CRE DIT_TYPE_ID	NUMBER	C			
SALES_CRE DIT_TYPE	VARCHAR2 (30)	C			
SOLD_TO_ORG	VARCHAR2 (360)				X
SOLD_TO_ORG_ID	NUMBER				X
QUOTA_FLAG	VARCHAR2 (1)			X	
PERCENT	NUMBER	REQUIRED			
CONTEXT	VARCHAR2 (30)				X
ATTRIBUTE 1..15	VARCHAR2 (240)				X
CREATED_BY	NUMBER			X	
CREATION_DATE	DATE			X	
LAST_UPDATED_BY	NUMBER			X	

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
LAST_UPDATE_DATE	DATE			X	
LAST_UPDATE_LOGIN	NUMBER			X	
PROGRAM_APPLICATION_ID	NUMBER				X
PROGRAM_ID	NUMBER				X
PROGRAM_UPDATE_DATE	DATE				X
REQUEST_ID	NUMBER				X
OPERATION_CODE	VARCHAR2 (30)	REQUIRED			
ERROR_FLAG	VARCHAR2 (1)				X
STATUS_FLAG	VARCHAR2 (1)				X

Conditional Settings:

OE_CREDITS_IFACE_ALL Conditional Settings

Column Name	Conditional Setting requirement
SALES_CREDIT_TYPE_ID & SALES_CREDIT_TYPE	Condition is that either one these columns should be populated

OE_CREDITS_IFACE_ALL Derived Values

- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DOCUMENT_REF =
OE_HEADERS_IFACE_ALL.ORIG_SYS_DOCUMENT_REF
- ORIG_SYS_LINE_REF = OE_LINES_IFACE_ALL.ORIG_SYS_LINE_REF
- ORIG_SYS_SHIPMENT_REF =
OE_LINES_IFACE_ALL.ORIG_SYS_SHIPMENT_REF

OE_LOTSERIALS_IFACE_ALL

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDER_SOURCE_ID	NUMBER	REQUIRED			
ORIG_SYS_DOCUMENT_REF	VARCHAR2(50)	REQUIRED			
ORIG_SYS_LINE_REF	VARCHAR2(50)	Conditional or optional (Required only if line level adjustment/credit)			
ORIG_SYS_SHIPMENT_REF	VARCHAR2(50)	OPTIONAL			X

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORIG_SYS_LOT_SERIAL_REF	VARCHAR2(50)	REQUIRED			
CHANGE_SEQUENCE	VARCHAR2(50)				X
CHANGE_REQUEST_CODE	VARCHAR2(30)				X
ORG_ID	NUMBER			X	
LOT_NUMBER	NUMBER	C			
FROM_SERIAL_NUMBER	VARCHAR2(30)	C			
TO_SERIAL_NUMBER	VARCHAR2(30)	C			
QUANTITY	NUMBER	REQUIRED			
SOLD_TO_ORG	VARCHAR2(360)				X
SOLD_TO_ORG_ID	NUMBER				X
CONTEXT	VARCHAR2(30)				X
ATTRIBUTE1..15	VARCHAR2(240)				X

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CREATED_BY	NUMBER	REQUIRED			
CREATION_DATE	DATE	REQUIRED			
LAST_UPDATED_BY	NUMBER	REQUIRED			
LAST_UPDATE_DATE	DATE	REQUIRED			
LAST_UPDATE_LOGIN	NUMBER	REQUIRED			
PROGRAM_APPLICATION_ID	NUMBER				X
PROGRAM_ID	NUMBER				X
PROGRAM_UPDATE_DATE	DATE				X
REQUEST_ID	NUMBER				X
OPERATION_CODE	VARCHAR2(30)	REQUIRED			
ERROR_FLAG	VARCHAR2(1)				X
STATUS_FLAG	VARCHAR2(1)				X

OE_LOTSERIALS_IFACE_ALL derived Values

- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DOCUMENT_REF =
OE_ORDER_LINES_ALL.ORIG_SYS_DOCUMENT_REF
- ORIG_SYS_LINE_REF = OE_LINES_IFACE_ALL.ORIG_SYS_LINE_REF
- ORIG_SYS_SHIPMENT_REF =
OE_LINES_IFACE_ALL.ORIG_SYS_SHIPMENT_REF

OE_RESERVATIONS_IFACE_ALL

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDER_SOURCE_ID	NUMBER	REQUIRED			
ORIG_SYS_DOCUMENT_REF	VARCHAR2(50)	REQUIRED			
ORIG_SYS_LINE_REF	VARCHAR2(50)	REQUIRED			
ORIG_SYS_SHIPMENT_REF	VARCHAR2(50)				X
ORIG_SYS_RESERVATION_REF	VARCHAR2(50)	REQUIRED			
CHANGE_SEQUENCE	VARCHAR2(50)				X
ORG_ID	NUMBER			X	
INVENTORY_ITEM_ID	NUMBER	REQUIRED			

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
REVISION	VARCHAR2(3)				X
LOT_NUMB ER_ID	NUMBER				X
LOT_NUMB ER	VARCHAR2(30)				X
SOLD_TO_ ORG	VARCHAR2(360)				X
SOLD_TO_ ORG_ID	NUMBER				X
SUBINVENT ORY_ID	NUMBER				
SUBINVENT ORY_CODE	VARCHAR2(10)				X
LOCATOR_I D	NUMBER				X
QUANTITY	NUMBER	REQUIRED			
ATTRIBUTE _CATEGOR Y	VARCHAR2(30)				X
ATTRIBUTE 1..15	VARCHAR2(240)				X
OPERATIO N_CODE	VARCHAR2(30)	REQUIRED			
REQUEST_I D	NUMBER				X

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ERROR_FLAG	VARCHAR2(1)				X

OE_RESERVTSN_IFACE_ALL Derived Values

- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DOCUMENT_REF =
OE_ORDER_HEADERS_ALL.ORIG_SYS_DOCUMENT_REF
- ORIG_SYS_LINE_REF = OE_ORDER_LINES_ALL.ORIG_SYS_LINE_REF
- ORIG_SYS_SHIPMENT_REF =
OE_ORDER_LINES_ALL.ORIG_SYS_SHIPMENT_REF
- INVENTORY_ITEM_ID = MTL_SYSTEM_ITEMS_B.INVENTORY_ITEM_ID

OE_ACTIONS_IFACE_ALL

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
ORDER_SOURCE_ID	NUMBER				X
ORIG_SYS_DOCUMENT_REF	VARCHAR2 (50)	REQUIRED			
ORIG_SYS_LINE_REF	VARCHAR2 (50)				X
ORIG_SYS_SHIPMENT_REF	VARCHAR2 (50)				X

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
CHANGE_SEQUENCE	VARCHAR2 (50)				X
ORG_ID	NUMBER			X	
HOLD_ID	NUMBER				X
HOLD_TYPE_CODE	VARCHAR2 (1)				X
HOLD_TYPE_ID	NUMBER				X
HOLD_UNTIL_DATE	DATE				X
RELEASE_REASON_CODE	VARCHAR2 (30)				X
SOLD_TO_ORG	VARCHAR2 (360)				X
SOLD_TO_ORG_ID	NUMBER				X
COMMENTS	VARCHAR2 (240)				X
CONTEXT	VARCHAR2 (240)				X
ATTRIBUTE1..15	VARCHAR2 (240)				X
REQUEST_ID	NUMBER				X

Column Name	Type	Required (C= Conditionally Required)	Conditionally Required for Booking	Derived	Optional
OPERATION_CODE	VARCHAR2 (30)	REQUIRED			
ERROR_FLAG	VARCHAR2 (1)				X
STATUS_FLAG	VARCHAR2 (1)				X
CHAR_PARAMETER1	VARCHAR2 (2000)				
CHAR_PARAMETER2	VARCHAR2 (240)				
CHAR_PARAMETER3	VARCHAR2 (240)				
CHAR_PARAMETER4	VARCHAR2 (240)				
CHAR_PARAMETER5	VARCHAR2 (240)				
DATE_PARAMETER1	DATE				
DATE_PARAMETER2	DATE				
DATE_PARAMETER3	DATE				
DATE_PARAMETER4	DATE				
DATE_PARAMETER5	DATE				

OE_ACTIONS_IFACE_ALL Derived Values

- ORDER_SOURCE_ID = OE_ORDER_SOURCES.ORDER_SOURCE_ID
- ORIG_SYS_DOCUMENT_REF =
OE_HEADERS_IFACE_ALL.ORIG_SYS_DOCUMENT_REF
- ORIG_SYS_LINE_REF = OE_LINES_IFACE_ALL.ORIG_SYS_LINE_REF
- ORIG_SYS_SHIPMENT_REF =
OE_LINES_IFACE_ALL.ORIG_SYS_SHIPMENT_REF

ONT.OE_PAYMENTS_IFACE_ALL

This is a multi-org table for payment records open interface. This table stores payment information that is imported from a feeder system into Oracle Order Management using Order Import. You can import a single payment record at the order level using the existing oe_headers_iface_all.

Note: Note on CREDIT_CARD_APPROVAL_CODE/DATE: An order can be created with the approval code and date interfaced using order import. Order Management (OM) will not authorize such order. Similarly, for data updates on approval code and date, OM will not validate the data.

ONT.OE_PAYMENTS_IFACE_ALL Table

Name	Data Type	Length	Mandatory	Comments
ORDER_SOURCE_ID	NUMBER	NA	No	Order Source Id
TRXN_EXTENSION_ID	NUMBER	22	No	Transaction Extension ID
INSTRUMENT_SECURITY_CODE	NUMBER	10	No	Instrument Security Code
REQUEST_ID	NUMBER	NA	No	REQUEST ID
ORIG_SYS_DOCUMENT_REF	VARCHAR2	50	No	Original system document reference

ORIG_SYS_LINE_REF	VARCHAR2	50	No	Original system line reference
ORIG_SYS_SHIPMENT_REF	VARCHAR2	50	No	Original system shipment reference
ORIG_SYS_PAYMENT_REF	VARCHAR2	50	No	Original system payment reference
CHANGE_SEQUENCE	VARCHAR2	50	No	Change sequence
CHANGE_REQUEST_CODE	VARCHAR2	30	No	Change Request Code
ORG_ID	NUMBER	NA	No	Organization ID
SOLD_TO_ORG_ID	NUMBER	NA	No	Foreign key to HZ_CUST_ACCOUNTS.CUST_ACCOUNT_ID
SOLD_TO_ORG	VARCHAR2	360	No	NA
PAYMENT_TYPE_CODE	VARCHAR2	30	Yes	Lookup type: OE_PAYMENT_TYPE.
PAYMENT_NUMBER	NUMBER	NA	No	For Insert actions, this number will be system generated.
COMMITMENT	VARCHAR2	240	No	Commitment name: description of a contractual guarantee with a customer for future purchase.

PAYMENT_TRX_ID	NUMBER	NA	No	Commitment ID, Customer Bank Account ID or NULL.
PAYMENT_METHODOD	VARCHAR2	30	No	AR Payment Method name
RECEIPT_METHOD_OD_ID	NUMBER	NA	No	Foreign key reference to AR_Receipt_Method.
PAYMENT_COLLECTION_EVENT	VARCHAR2	30	No	Lookup type: OE_PAYMENT_COLLECTION_EVENT. Values are Prepay or Invoice.
PAYMENT_SET_ID	NUMBER	NA	No	Foreign key reference to AR. Unique ID to identify receipts created for an order.
PREPAID_AMOUNT	NUMBER	NA	No	Receipts created in AR.
CREDIT_CARD_NUMBER	VARCHAR2	80	No	A holder for credit card number if payment type code is Credit Card. If ap_bank_account_id already exists, do not populate this column and use the payment_trx_id column instead.

CREDIT_CARD_ HOLDER_NAME	VARCHAR2	80	No	A holder for credit card holder name if payment type code is Credit Card. If ap_bank_account_id already exists, do not populate this column and use the payment_trx_id column instead.
CREDIT_CARD_ EXPIRATION_DATE	DATE	NA	No	A holder for credit card expiration date if payment type code is Credit Card. If ap_bank_account_id already exists, do not populate this column and use the payment_trx_id column instead.
CREDIT_CARD_ CODE	VARCHAR2	80	No	Lookup type: Credit_Card
CREDIT_CARD_ APPROVAL_CODE	VARCHAR2	80	No	A holder for credit card approval code if payment type code is Credit Card.
CREDIT_CARD_ APPROVAL_DATE	DATE	NA	No	A holder for credit card approval date if payment type code is Credit Card.

CHECK_NUMBER	NUMBER	NA	No	A holder for check number if payment type code is Check.
PAYMENT_AMOUNT	NUMBER	NA	No	A holder for payment amount input by user, or system calculated from payment_percent; for example, amount written on check, commitment amount to be promised, or prepayment amount to be collected.
PAYMENT_PERCENTAGE	NUMBER	NA	No	Value is 1 to 100. Do not enter this column if Payment Amount has been populated.
DEFER_PAYMENT_PROCESSING_FLAG	VARCHAR2	1	No	Set this flag to 'Y' if you want to process payment offline. This applies to both prepayment and credit card authorization.
CREATION_DATE	DATE	NA	No	Standard Who column
CREATED_BY	NUMBER	NA	No	Standard Who column
LAST_UPDATE_DATE	DATE	NA	No	Standard Who column

LAST_UPDATE D_BY	NUMBER	NA	No	Standard Who column
LAST_UPDATE _LOGIN	NUMBER	NA	No	Standard Who column
REQUEST_ID	NUMBER	NA	No	Concurrent request that last changed row
PROGRAM_AP PLICATION_ID	NUMBER	NA	No	Application of concurrent program that last updated row.
PROGRAM_ID	NUMBER	NA	No	Concurrent program that last updated row.
PROGRAM_UP DATED_DATE	DATE	NA	No	Last date changed by concurrent program
CONTEXT	VARCHAR2	30	No	Descriptive flex-field context
ATTRIBUTE1	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE2	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE3	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE4	VARCHAR2	240	No	Descriptive flex-field segment

ATTRIBUTE5	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE6	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE7	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE8	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE9	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE10	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE11	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE12	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE13	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE14	VARCHAR2	240	No	Descriptive flex-field segment
ATTRIBUTE15	VARCHAR2	240	No	Descriptive flex-field segment

OPERATION_C ODE	VARCHAR2	30	No	Operation Code
STATUS_FLAG	VARCHAR2	1	No	Status Flag
INTERFACE_ST ATUS	VARCHAR2	1000	No	Interface Status
ERROR_FLAG	VARCHAR2	1	No	Error Flag

Indexes

OE_PAYMENTS_IFACE_N1: Request_ID

OE_PAYMENTS_IFACE_N2: Order_Source_Id,

Orig_Sys_Document_Ref, Orig_Sys_Line_Ref, Orig_sys_shipment_ref,
orig_sys_payment_ref

Primary key: N2+Change Sequence

Process Order Application Open Interface

The Sales Order has been modeled as a business object that Oracle Order Management owns. The Sales Order business object comprises of several entities, namely, Header, Header Sales Credits, Header Price Adjustments, Header Pricing Attributes, Header Adjustment Attributes, Header Adjustment Associations, Lines, Line Sales Credits, Line Price Adjustments, Line Pricing Attributes, Line Adjustment Attributes, Line Adjustment Associations, and Line Lot Serial Numbers.

The Process Order Application Program Interface (API) is designed as the mechanism through which all data manipulation (inserts, updates and deletes) may be performed on the Sales Order business object entities and their attributes, in a consistent manner. Besides these, certain other action requests such as applying holds, attachments, booking etc. can also be processed using the Sales Order API. Business logic in the API not only takes care of updates to the attributes but also make calls to other functions depending on the changes to attribute values.

The importance of using the Process Order API for all data manipulation to the Sales Order Business object cannot be over stressed. It must be understood that by using the Process Order API we not only avoid duplication of business logic in many functions but also move towards the distributed solution approach.

Process Order API Features - Operations on the Sales Orders Object

The Process Order API can be used to create, update or delete the following entities that the sales order business object consists of.

Process Orders Entities and Associated Tables

Entity	Table Name
Order Header	OE_ORDER_HEADERS_ALL
Order Price Adjustments	OE_PRICE_ADJUSTMENTS
Order Sales Credits	OE_SALES_CREDITS
Order Line	OE_ORDER_LINES_ALL
Order Pricing Attributes	OE_ORDER_PRICE_ATTRIBS
Order Adjustment Attributes	OE_PRICE_ADJ_ATTRIBS
Order Adjustment Associations	OE_PRICE_ADJ ASSOCS
Line Sales Credits	OE_SALES_CREDITS
Line Price Adjustments	OE_PRICE_ADJUSTMENTS
Line Pricing Attributes	OE_ORDER_PRICE_ATTRIBS
Line Adjustment Attributes	OE_PRICE_ADJ_ATTRIBS
Line Adjustment Associations	OE_PRICE_ADJ ASSOCS
Lot Serial Numbers	OE_LOT_SERIAL_NUMBERS

Passing Parameters By Values

Process Order API provides users the capability of passing the attributes on the order entities by their display values instead of their internal identifiers (IDs or codes).

For example, to specify the customer on the order, the user can either pass in the customer ID on the header record, `p_header_rec.sold_to_org_id` or send in the display name of the customer on the header value record, `p_header_val_rec.sold_to_org`.

The values are internally resolved into the identifiers for all the entity records passed to process order. For the value fields that could not be resolved, error messages are posted to the OM message stack and none of the records are processed any further.

If both the value and the identifier fields are populated for the same attribute, then information messages are posted to the OM message stack for such attributes. The

identifier field takes preference and further processing is based on this field.

Multi-Org Access Control

Process Order, Lock Order and Get Order APIs are enhanced to accept Operating Unit as an optional input parameter so that you can specify the operating unit in which to process the order. If an Operating Unit value is not passed explicitly to the API, then the API uses the default Operating Unit derived from the security/Operating Unit profile options. If no default Operating Unit is specified, an error will be raised. These APIs also check if the current responsibility has access to the Operating Unit. If not, an error is raised.

When calling Process Order, Lock Order or Get Order APIs, if you already know the Operating Unit name or org_id, you may pass it so that the API doesn't need to derive the OU from the profile options.

If you are enabling Multi-Org Access Control in your application:

In addition to calling FND_GLOBAL.APPS_INITIALIZE(), call the procedure mo_global.init('ONT') procedure, before calling the Public API. This ensures that the multi-org access control context is appropriately set. If this is not done, then an error will be displayed as the logic that checks for OU access will fail.

If you pass in an org_id / operating unit name and you have access to that Operating Unit, the org context is set accordingly. If you do not pass in an org_id / operating unit name, then the call is processed for the default Operating Unit. If a default cannot be derived the call fails.

If you are NOT enabling Multi-Org Access Control in your application:

The existing Process Order , Lock Order or Get Order API calls will work as before, and will be validated and processed for the MO: Operating Unit for the applications context set for that call.

Note: The MOAC validation API determines whether MOAC is enabled or not based on whether MO Security Profile is set.

Though Order Management APIs do not seed any error messages, the Multi-Org API is called and raises one of the following messages depending on the nature of the failure:

- FND_MO_NOINIT_SP_PUB_API: Raised if the Multi-Org initialization is not done.
- FND_MO_INVALID_OU_API: Raised if you pass an org_id that is invalid.
- FND_MO_INVALID_OU_PUB_API: Raised if you do not pass an org_id and it cannot be derived.

Pricing an Order/Line

Pricing in process order API can be controlled using flag calculate_price_flag on order line record. When set to Y (Yes), the process order API fetches the list price and applies

adjustments and charges. When set to N, the process order API would fetch a list price if the list price is not passed and no adjustments would be applied. When set to P, all the modifiers which are associated with phases having override freeze flag set to Y are applied. That mainly includes freight charges.

You may use the Process Order Interface for order repricing, provided the order data you are updating has an change that will trigger repricing. Once Process Order is invoked and program logic determines an attribute that can trigger order repricing has been updated, a call to the pricing engine is made to reprice the order.

The following order attributes (database columns) can trigger Process Order to reprice an order or order line:

- agreement_id
- cust_po_number
- inventory_item_id
- invoice_to_org_id
- ordered_item_id
- ordered_item
- line_category_code
- line_type_id
- ordered_quantity
- ordered_quantity_uom
- preferred_grade
- payment_term_id
- price_list_id
- pricing_date
- request_date
- ship_to_org_id
- sold_to_org_id
- service_start_date

- service_end_date
- service_duration
- service_period

Along with the preceding attributes, a pricing call will be invoked when a change occurs in the following attributes (from OEXULINB.pls):

- Cancelled_quantity on the sales order line.
- Ordered_quantity2 (Secondary order quantity): Ability to price by secondary order quantity.
- Unit_selling_price on the sales order line is cleared.

You can also choose to update order pricing via the action *Price Order* from the Order Organizer or Sales Orders window.

Note: You cannot interface an order line having a price list with a currency code different from the existing or newly created order header's currency code. An error message will be displayed in the Process Messages window.

Scheduling/Reservations

Process order API can be used to perform scheduling actions on order lines. Scheduling actions include: schedule, unschedule, reserve and unreserve.

The schedule_action_code field provided on the order line record (line_rec_type) can be used to provide the action which needs to be performed on the order line.

Reservations can also be performed by passing the reserved_quantity on the order line record. You do not need to send in the schedule action if reserved_quantity is passed.

Process Order API will also automatically schedule or re-schedule the lines if the schedule_ship_date or schedule_arrival_date field is passed or updated respectively.

Alternatively, you can just set the profile option *OM: AutoSchedule* to Yes and all standard lines will be automatically scheduled as they are created. This holds true only if the lines are not part of any set.

Return Lines

Process order can be used to create and update return lines also. To create a return line, you can either pass in the line category of *RETURN* on the order line record and the line type would default to the inbound line type of the order type. Alternatively, you can also provide a line type of the type *Return* on the order line record.

Additionally, if you want to specify a reference for the return line, you can pass in the return flexfields (return_context, return_attribute1-2).

Column Return_Context can have the following values to determine the reference type:

- Sales Order
- Customer PO
- Invoice
- Serial Number

Return_Attribute1... Return_Attribute2 can have the following values depending on the reference type:

- Sales Order
 - return_Attribute1: Header ID
 - return_Attribute2: Line ID
- Customer PO
 - return_Attribute1: Header ID
 - return_Attribute2: Line ID
- Invoice
 - return_Attribute1: Invoice Header ID
 - return_Attribute2: Invoice Line ID
- Serial Number
 - return_Attribute1: Inventory Item ID
 - return_Attribute2: Serial Number

Special Considerations for using Process Order for return orders

Creation of a non-referenced RMA: If you wish to process orders for a return order for a non referenced return, you must

- populate all required attributes for creating a return order
- use an order category of RETURN or MIXED for the Order Header record.
- populate all required attributes for creating a return order lines.
 - For Order Line Record, the line category should be set to RETURN, and specify a RETURN LINE TYPE within column line_type_id.

- Line_type_id is optional, provided a default line_type has been defined for the specified Order Type.)
- Additionally, you will need to populate the column reason_code for all return lines. Valid values are those values defined for the Order Management quickcode *CREDIT_MEMO_REASON*.

Creation of a Referenced RMA (if you want to return an existing outbound line): If you create a referenced RMA, you should copy the Header Record for the return from the referenced Order header record, modifying the Order Type to category RETURN or MIXED (order_type_id from oe_transaction_types_all).

For the Order Line record, populate the following attributes only:

1. Line_category_code: RETURN
2. return_context: ORDER
3. return_attribute1: header_id from the referenced order.
4. return_attribute2: line_id from the referenced order line.
5. calculate_price_flag: Set it to P if you want to retain the original price, the flag to Y if you want to reprice the RMA line.
6. line_type_id: Assign a line_type_id from RMA line. Line_type_id is optional, provided a default line_type has been defined for the specified Order Line Type.)
7. return_reason_code: Populate a reason code from lookup_type = *CREDIT_MEMO_REASON*.
8. Sales credit details: populate the header_level/line level sales credits from the referenced order.

Capturing Customer Acceptance

To capture customer acceptance using Process Order, you need to use Action Requests. You can attach these requests at the order level or at the line level. For each acceptance or rejection, create a request record in the action request table. If the ID of the order or line for which this action is requested on is not available (order or order line is being created at the same time), assign the entity_index field with the index for the entity record.

An example of how Process Order API is called to record Customer Acceptance:

Create acceptance records for order line id = 100. Quantity on the order = 10. Full acceptance of quantity 10.

```

-- ACTION REQUEST RECORD for acceptance

-- line level action
l_request_rec.entity := G_ENTITY_LINE;
-- line id
l_request_rec.entity_id := 100;
-- action requested
l_request_rec.request_type := G_ACCEPT_FULFILLMENT;

-- parameters
-- acceptance type code
l_request_rec.param1 := 'Fulfillment';
-- quantity
l_request_rec.param2 := 10;
-- customer signature
l_request_rec.param4 := 'John Smith';

-- customer signature date
l_request_rec.date_param2 := '20-DEC-2004';
-- inserting request record into action request table
l_action_request_tbl := l_request_rec;

-- CALL TO PROCESS ORDER
OE_Order_PVT.Process_Order(
.....
-- enabling validation of the request record parameters
p_validation_level => FND_API.G_VALID_LEVEL_FULL
.....
-- action request table
x_action_request_tbl => l_action_request_tbl
.....
-- OUT variables
.....
);

TYPE Request_Rec_Type IS RECORD
(

    Entity_code          Varchar2(30) := NULL,
    Entity_id            Number := NULL,
    Entity_index          Number := NULL,
    request_type          Varchar2(30) := NULL,

    return_status         VARCHAR2(1)    := FND_API.G_MISS_CHAR,

    param1                Varchar2(2000) := NULL,
    param2                Varchar2(240)  := NULL,
    param3                Varchar2(240)  := NULL,
    param4                Varchar2(240)  := NULL,

    .
    .
    .
    date_param1           DATE := NULL,
    date_param2           DATE := NULL,
    .
    .
    .
)

```

Acceptance Fulfillment

Lines waiting for fulfillment acceptance will be accepted for revenue recognition based

on the parameters sent in on the request record:

Parameter	Description
request_type	OE_GLOBALS.G_ACCEPT_FULFILLMENT
entity_code	OE_GLOBALS.G_ENTITY_HEADER for accepting all eligible lines waiting for fulfillment acceptance in an order
param1	Acceptance Comments
param2	Acceptance Customer Signature
param3	Acceptance Reference Document
param4	Implicit acceptance
date_param1	Acceptance date

Rejection Fulfillment

Lines waiting for fulfillment acceptance will be rejected for revenue recognition based on the parameters sent in on the request record:

Parameter	Description
request_type	OE_GLOBALS.G_ACCEPT_FULFILLMENT
entity_code	OE_GLOBALS.G_ENTITY_HEADER for accepting all eligible lines waiting for fulfillment acceptance in an order
param1	Acceptance Comments
param2	Acceptance Customer Signature
param3	Acceptance Reference Document
param4	Implicit rejection
date_param1	Rejection date

Holds/Releases

The existing hold sources are evaluated and if they are applicable, holds are applied and released automatically on the orders or order lines when created/updated via the process order API.

For e.g. if there is a hold source defined for the customer ABC, all orders for that customer are placed on hold as they are entered.

Attachments

If the profile option *OM: Apply Automatic Attachments* is set to *Yes* and if attachment rules are applicable, attachments are automatically applied to the order header or order lines when they are created via the process order API.

Note: Attachments are NOT automatically deleted or re-applied if the order or line is updated.

The caller can also send in an explicit action request to apply attachments. Please refer to the section Action Requests for more details.

Sets

Process order can be used to add or delete order lines from ship sets, arrival sets or fulfillment sets. Action Requests can be used to add or remove lines from fulfillment sets. Please refer to Process Order and Action Requests sections for more details.

User can add a line into a new set by passing set name (ship_set/arrival_set/fulfillment_set) on the line record or user can use set id (ship_set_id/arrival_set_id/fulfillment_set_id) to add to an existing set.

Splits

Process order can be used to split an existing order line into shipments. User can perform splits by calling process order with both the line records, one to reduce the quantity on the existing line record with the action of SPLIT and another to create the new line with the remaining quantity and a reference to the line that it was split from. For e.g. if user wants to split a line with original quantity 10 into 6 and 4 he must populate the table in the following manner. In the first update call to process order user cannot update any other attributes other than ordered quantity.

```
line_tbl(1).split_action_code := 'SPLIT'
```

```
line_tbl(1).split_by := 'USER'
```

```
line_tbl(1).ordered_quantity := 6
```

```
line_tbl(1).operation = oe_globals.g_opr_update
```

```

line_tbl(2).split_action_code := 'SPLIT'

line_tbl(2).split_by := 'USER'

line_tbl(2).ordered_quantity := 4

line_tbl(2).operation = oe_globals.g_opr_create

line_tbl(2).split_from_line_id := line_tbl(1).line_id

```

User can also optionally pass in the change reason and change comments on the original line record that is being updated.

Cancellation

Process order can be used to cancel orders and order lines. User should update the cancelled flag to Y on the header record if user intends to cancel the entire order. Cancellation on the line is performed by reducing the ordered quantity on the line record. User has to supply change reason and optionally change comments on the line record.

Tax Calculation

Process order will check whether the transaction type on the order line is taxable or whether user has specified that tax should be calculated (set the tax exempt flag to Required). The calculated tax is always an estimated value and is internally stored as a price adjustment.

Freight and Special Charges

Freight and special charges can be setup in pricing as modifiers. When the order or line is priced, these charges get applied on the order or line. These do not affect the unit selling price and are also stored as price adjustments.

Users can also specify the freight and special charges to be applied by setting up the price adjustment records appropriately. Such price adjustments should have the list_line_type_code parameter set to *CHARGE*.

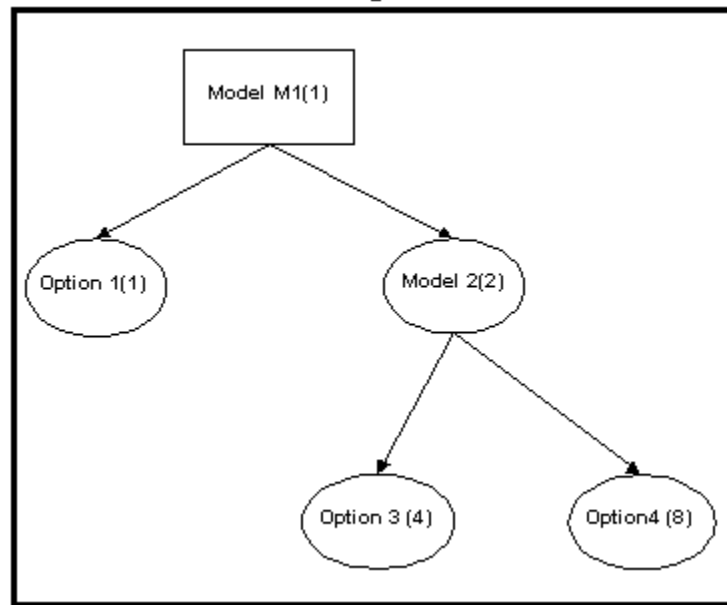
Cascading

Process Order will automatically cascade changes to other dependent entities. For example,

- cascade quantity changes from model lines to all options lines
- if a set identifying attribute is changed on one line of the set, the change will be automatically applied to all lines on the set.

The processing sequence is very important when considering automatic cascading. Process Order will go by the following order when processing the line entity. First, all

standard lines and model lines are processed. These are followed by processing of requests in their arrival order. The importance of declaring the processing sequence may be depicted by the following example. Consider that a model M1 has O1 and M2 as two of its options. M2 has O2 and O3 as two options defined for it. Graphically represented, it would look like this.



Let us assume that the following requests are received by Process Order API:

- Request 1: Update the quantity of Option4 to 20.
- Request 2: Update the quantity of Model2 to 4.

If requests are processed in the order that they were received, the following would occur:

Option 4 would be updated to 20 assuming that the new ratio between Model2 and Option4 (1:10) falls within a valid range. Next Model2 would be updated to 4 (assuming again that ratio of Model1 to Model2 is valid). That would then result in quantity cascading, thereby updating Option3 to 8 and Option4 to 40. So the resulting quantities are:

Model2 = 4, Option3 = 8, Option4 = 40.

If processing sequence of requests is based on item type (models first, classes next, options after that etc.), the following would be the sequence in which the requests would be processed:

Request 2 would be processed first, with quantity of Model2 to be updated to 4. Next quantity of Option4 would be updated to 20 (assuming ratios fall in the right range). So the resulting quantities are:

Model2 = 4, Option3 = 8 and Option4 = 20.

The above example highlights the differences with processing lines (termed in the example as requests) in the sequential arrival order versus processing based on item types.

As stated earlier, process order API will process Standard Lines and top level Model lines first., then option classes and option lines and service lines in the end. For lines in the same category, the requests would be processed in the order that they are received.

Booking an Order

The following order and line level columns and fields are required for booking an order:

Order Header

- The following fields are required at the order header level:
 - Order_Number
 - Sold_to_Org
 - Invoice_to_Org_Id
 - Price_list
 - Tax_Exempt_Flag
 - Salesrep_Id (Salesperson ID)
 - Ordered_Date
- The following fields are required for regular or mixed orders, but not for return orders:
 - Ship_to_Org
 - Payment_Term_Id
- If Conversion_Type_Code = User, then Conversion_Rate and Conversion_Rate_Date are required.
- For checks based on Payment Type: If Payment_Type_Code is not null, then the Payment_Amount is required. The Check_Number is required if the payment_type is check .

Line

- Required fields:
 - Order_Number

- Sold_to_Org
- Invoice_to_Org_Id
- Price_list
- Tax_Exempt_Flag
- Salesrep_Id (Salesperson ID)
- Ordered_Date
- Inventory_Item_Id
- Ordered_Quantity
- Ordered_Quantity_UOM

The following fields are required for regular or mixed orders, but not for return orders:

- Ship_to_Org
- Payment_Term_Id
- Items that are not included items or configurable items--for example, Item_Type_Code is INCLUDED or CONFIG, Price_List , Unit_Selling_Price , and Unit_List_Price--are required.
- Warehouse, Schedule Date, and Tax_Date are required on return lines.
- If Tax_Exempt_Flag = R (Required), then the Tax_Code is needed.
- For Service Item Type (Item_Type_Code = SERVICE):
If the Service_Coterminate_Flag = Y (yes) and Service_Reference_Type_Code is Customer Product, then the following fields are required: Service_Start_Date, Service_End_Date, Service_Duration, and Service_Period.

Functional Overview - Public Procedures

Process Order API provides the following public procedures for operations and queries on the entities for the sales order object.

- Process_Order
- This is the main procedure used to create, update and delete the entities on the sales order. This procedure can also be used to perform other actions via the records or

explicit action requests.

Please refer to the next section on the details of the processing.

- **Get_Order**

This procedure queries and returns all records belonging to a single sales order object.

- **Lock_Order**

Locks the entities of a sales order object. Use the `get_order` procedure to retrieve the entity records and then call `lock_order` with the records to be locked.

Processing the Sales Order Business Object

Process order can only process entities belonging to one sales order in one i.e. in a single call, it can accept one header record and a table of header adjustment, header sales credits, lines, line adjustments, line sales credit and line lot serial numbers records for processing. The entities are processed in the following sequence:

1. Process Header Record
2. Process Header Adjustments
3. Process Header Pricing Attributes
4. Process Header Adjustment Attributes
5. Process Header Adjustment Associations
6. Process Header Sales Credits
7. Check Entity context to make sure all lines belong to one header
8. Process Lines
9. Process Line Adjustments
10. Process Line Pricing Attributes
11. Process Line Adjustment Attributes
12. Process Line Adjustment Associations
13. Process Line Sales Credits
14. Process Line Lot Serial Numbers

15. Perform Cross Entity Logic for Sales Order business object

The procedure `process_order` performs the following actions for each entity on the order object:

Attribute Level Security Check

If operation on the record is `UPDATE` or `CREATE`, then for all the attributes that have changed between the old and the new record, constraints are evaluated to check if the user is allowed to change the attribute. An error is raised if there was at least one attribute that failed security check and the record is not processed further.

Attribute Validation

All the attributes that are passed in by the caller on the entity record are validated. Errors are posted to the OM message stack for all attributes that are invalid. An error is raised at the end if at least one attribute failed validation and the record is not processed further.

Clear Dependent Attributes

If operation is `UPDATE`, then the fields dependent on the updated fields are cleared (or set to `MISSING` values). This is done so that the dependent fields are re-defaulted. For e.g. if the customer on the order is being updated, then the contacts, ship to and bill to on the order should be re-defaulted.

Note: If the user is also trying to update the dependent fields in the same call to the process order API, then the fields will not be cleared and the user-specified value will be used instead.

Defaulting

Defaulting occurs for each missing attribute.

Attributes that are not explicitly passed by the user and therefore, retain the values on the initialized record are defined as `MISSING` attributes. For e.g. all the number fields on the entity records are initialized to the value `FND_API.G_MISS_NUM` and thus, all number fields with this value are `MISSING` attributes.

- Check security if new default value is different from old value and if there is a valid constraint, then an error is raised and abort processing of this record.
- Validate the default value if not null. If new default value is `NOT` valid, then set the attribute to `NULL`.
- If default value is valid, clear the dependent attributes (set to `MISSING` values).

Note: If the user has specified the values of these dependent attributes in the same call to process order, then the user-specified values will take preference and will not be cleared.

- Re-default all the missing attributes (dependent attributes) till all attributes have a value.

Record Validation

There are three main validation steps:

1. Check for Required Attributes. Validate that all the required attributes have been specified on the entity and even if one required attribute is missing, raise an error and quit. For e.g. inventory item is a required field on the order line entity.
2. Check for Conditionally Required Attributes. Validate that all attributes that are required based on the status of the entity have been specified and if at least one is not specified, raise an error and quit. For e.g. for a booked line, ship to location is required.
3. Cross-Attribute Validation. Validate all those attributes that are dependent on the value of another attribute. At the end of the validation, if at least one attribute is not valid, then raise an error and quit. For e.g. verify that the ship to is valid for the customer.

Entity Security Check

Entity level security check (constraints that are not attribute-specific) is done once again before the entity record is posted to database as defaulting may have changed the values of some attributes for e.g. ship set may have defaulted onto the line and there is a constraint against inserting a line into a ship set where all lines in the ship set have been invoiced.

Any constraints that may have been setup for DELETE operation on this entity are also evaluated here.

Database Write

Write the changes to the database.

Cross Record Logic

After all the records have been processed for one entity, other cross record logic like validations, automatic application of holds etc. is done. For e.g. after all records are processed for the entity *order sales credit*, validate that the total percent adds up to 100.

Cross Entity Logic

After all the entities have been processed, cross entity logic is performed for the business object as a whole.

Note: In addition to the above a check is made to validate entity context for the line to ensure all lines belong to same header.

Start Processing Record


```

if operation          if operation

DELETE                CREATE or UPDATE

Attribute Security Check

error

Attribute Validation

if operation          if operation          error

CREATE                UPDATE

Clear Dependent Attributes

Set the error

status on the record

Defaulting

errorRollback all changes.

Exit with error status

Record Validation

error

Entity Security Check

Database Write

Process

Next

Record Record Processed

Successfully

error

All records processed

Cross Record Logic error

```

All entities processed

Process Action Requests

Exit with Success Status

Setting Up the Process Order Procedure

Before using the API, set up and activate the following parameters:

- Operating Unit. For more information please refer to the section on Multi-Org Access Control, page 7-112.
- Version number
- Initialize message list
- Initialize the IN record parameters to be interfaced to missing values
- Setup the parameters on the records to be interfaced

Parameter Descriptions

The following chart describes all parameters used by the public API OE_ORDER_PUB.PROCESS_ORDER. All of the inbound and outbound parameters are listed. The OUT parameters have to be assigned when calling the API and are therefore marked as Required fields. Additional information on these parameters follows.

OE_ORDER_PUB.PROCESS_ORDER

Parameter	Usage	Type	Required	Optional	Description	Default
P_Operating_Unit	IN	VARCHAR 2		X	Operating Unit Name	
P_org_id	IN	NUMBER		X	Internal id for Operating Unit	

Parameter	Usage	Type	Required	Optional	Description	Default
p_api_version_number	IN	NUMBER	X		Used to compare the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible	
p_init_msg_list	IN	Varchar2		X	Requests that the API initialize the message list on your behalf.	FND_API.G_FALSE
p_return_values	IN	Varchar2		X	Requests that the API send back the values on your behalf.	FND_API.G_FALSE

Parameter	Usage	Type	Required	Optional	Description	Default
x_return_status	OUT	Varchar2	X		<p>Returns the status, indicates whether the request was processed successfully or no</p> <p>Success: FND_API. G_RET_STS_SUCCESS</p> <p>Error: FND_API. G_RET_STS_ERROR</p> <p>Unexpected Error: FND_API. G_RET_STS_UNEXP_ERROR</p>	

Parameter	Usage	Type	Required	Optional	Description	Default
x_msg_count	OUT	NUMBER	X		<p>Indicates number of error messages API has encountered.</p> <p>If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call OE_MSG_PUB.GET. This api can be called with the following parameter values:</p> <p>p_msg_index => 1</p> <p>p_encoded => F</p> <p>p_data => 1_message</p> <p>p_msg_index_out => 1_msg_index_out</p> <p>where 1_message and</p>	

Parameter	Usage	Type	Required	Optional	Description	Default
					1_msg_in ex_out should be local variables of types Varchar2(2 000 and Number respectivel y	
x_msg_data	OUT	Varchar2	X		Displays error message text. If the x_msg_coun t is equal to 1, then this contains the actual message	
p_header_rec	IN	PL/SQL Record		X	Use this parameter to send in operations on the order header entity	G_MISS_H EADER_R EC
p_old_header_rec	IN	PL/SQL Record		X	Use this parameter to send in the old record for the order header entity.	G_MISS_H EADER_R EC

Parameter	Usage	Type	Required	Optional	Description	Default
p_header_val_rec	IN	PL/SQL Record		X	Use this parameter to send in the display values for fields on the order header entity.	G_MISS_HEADER_VAL_REC
p_old_header_val_rec	IN	PL/SQL Record		X	Use this parameter to send in display values for the fields on the old record for the order header entity.	G_MISS_HEADER_VAL_REC
p_header_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order price adjustment entities	G_MISS_HEADER_ADJ_TBL
p_old_header_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the order price adjustment entities.	G_MISS_HEADER_ADJ_TB

Parameter	Usage	Type	Required	Optional	Description	Default
p_header_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the order price adjustment entities	G_MISS_HEADER_ADJ_VAL_TBL
p_old_header_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in display values for the fields on the old records for the order price adjustment entities.	G_MISS_HEADER_ADJ_VAL_TBL
p_header_price_attr_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order pricing Attributes entities	G_MISS_HEADER_PRICE_ATTR_TBL
p_old_header_price_attr_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the order pricing Attributes entities	G_MISS_HEADER_PRICE_ATTR_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_header_adj_att_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order Adjustment Attributes entities.	G_MISS_HEADER_ADJ_ATT_TBL
p_old_header_adj_att_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the order Adjustment Attributes entities.	G_MISS_HEADER_ADJ_ATT_TBL
p_header_adj_assoc_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order Adjustment Associations entities.	G_MISS_HEADER_ADJ_ASSOC_TBL
p_old_header_adj_assoc_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the order Adjustment Associations entities.	G_MISS_HEADER_ADJ_ASSOC_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_header_scredit_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order sales credits.	G_MISS_HEADER_SCREDIT_TBL
p_old_header_scredit_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old record for the order sales credits.	G_MISS_HEADER_SCREDIT_TBL
p_header_scredit_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the order sales credits.	G_MISS_HEADER_SCREDIT_VAL_TBL
p_old_header_scredit_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in display values for the fields on the old record for order sales credits.	G_MISS_HEADER_SCREDIT_VAL_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_Header_Payment_tbl	IN					Header_Payment_Tbl_Type := G_MISS_HEADER_PAYMENT_TBL
p_old_Header_Payment_tbl	IN					Header_Payment_Tbl_Type := G_MISS_HEADER_Payment_TBL
p_Header_Payment_val_tbl	IN					Header_Payment_Val_Tbl_Type := G_MISS_HEADER_Payment_VAL_TBL
p_old_Header_Payment_val_tbl	IN					
p_line_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order lines.	G_MISS_LINE_TBL
p_old_line_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the order lines.	G_MISS_LINE_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_line_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the order lines.	G_MISS_LI NE_VAL_TB L
p_old_line_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in display values for the fields on the old records for order lines.	G_MISS_LI NE_VAL_TB L
p_line_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the line price adjustment entities.	G_MISS_LI NE_ADJ_TB L
p_old_line_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the line price adjustment entities.	G_MISS_LI NE_ADJ_TB L

Parameter	Usage	Type	Required	Optional	Description	Default
p_line_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the line price adjustment entities.	G_MISS_LI NE_ADJ_V AL_TBL
p_old_line_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in display values for the fields on the old records for the line price adjustment entities.	G_MISS_LI NE_ADJ_V AL_TBL
p_line_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the Line pricing Attributes entities.	G_MISS_LI NE_PRICE _ATT_TBL
p_old_line_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the Line pricing Attributes entities.	G_MISS_LI NE_PRICE _ATT_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_line_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the Line Adjustment Attributes entities.	G_MISS_LINE_ADJ_ATT_TBL
p_old_line_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the Line Adjustment Attributes entities.	G_MISS_LINE_ADJ_ATT_TBL
p_line_price_att_tbl	IN	PL/SQL Table		X		
p_old_line_price_att_tbl	IN	PL/SQL Table		X		
p_line_adj_att_tbl	IN	PL/SQL Table		X		
p_old_line_adj_att_tbl	IN	PL/SQL Table		X		
p_line_adj_assoc_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the Line Adjustment Association entities.	G_MISS_LINE_ADJ_ASSOC_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_old_line_adj_assoc_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old records for the Line Adjustment Associations entities.	G_MISS_LI NE_ADJ_A SSOC_TBL
p_line_credit_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the line sales credits.	G_MISS_LI NE_SCRE DIT_TBL
p_old_line_scredit_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old record for the line sales credits.	G_MISS_LI NE_SCRE DIT_TBL
p_line_scredit_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the line sales credits	G_MISS_LI NE_SCRE DIT_VAL_ TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_old_line _scredit_val tbl	IN	PL/SQL Table		X	Use this parameter to send in display values for the fields on the old record for line sales credits.	G_MISS_LI NE_SCRE DIT_VAL_ TBL
p_Line_Pa yment_tbl	IN					Line_Paym ent_Tbl_Ty pe := G_MISS_H EADER_P AYMENT_ TBL
p_old_Line _Payment_ tbl	IN					Line_Paym ent_Tbl_Ty pe := G_MISS_H EADER_Pa yment_TBL
p_Line_pa yment_val_ tbl	IN					Line_Paym ent_Val_Tb l_Type := G_MISS_LI NE_Payme nt_VAL_T BL
p_old_Line _Payment_ val_tbl	IN					
p_lot_serial _tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the lot serials.	G_MISS_L OT_SERIA L_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_old_lot_serial_tbl	IN	PL/SQL Table		X	Use this parameter to send in the old record for the lot serials.	G_MISS_LOT_SERIAL_TBL
p_lot_serial_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the lot serials.	G_MISS_LOT_SERIAL_VAL_TBL
p_old_lot_serial_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in display values for the fields on the old record for lot serials.	G_MISS_LOT_SERIAL_VAL_TBL
p_action_request_tbl	IN	PL/SQL Table		X	Use this to send in requests to process other actions on the order	G_MISS_REQUEST_TBL
x_header_rec	OUT	PL/SQL Table	X		Returns the processed order header record.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_val_rec	OUT	PL/SQL Table	X		Returns the values for the processed order header record for the sales order if p_return_values was set to FND_API.G_TRUE	
x_header_adj_tbl	OUT	PL/SQL Table	X		Returns the records for the processed price adjustments	
x_header_adj_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed header price adjustments if p_return_values was set to FND_API.G_TRUE	
x_header_pricing_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header pricing attributes	

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_adj_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header adjustment attributes	
x_header_adj_assoc_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header adjustment associations	
x_header_sales_credit_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header adjustment associations	
x_header_sales_credit_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed header sales credits if p_return_values was set to FND_API.G_TRUE	
x_line_tbl	OUT	PL/SQL Table	X		Returns the processed order lines.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_line_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed order lines if p_return_values was set to FND_API.G_TRUE	
x_line_adj_tbl	OUT	PL/SQL Table	X		Returns the processed line price adjustments for the sales order	
x_line_adj_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed line price adjustments for the sales order if p_return_values was set to FND_API.G_TRUE	
x_line_price_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed line adjustment attributes	

Parameter	Usage	Type	Required	Optional	Description	Default
x_line_adj_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed line adjustment associations	
x_line_adj_assoc_tbl	OUT	PL/SQL Table	X		Returns the records for the processed line adjustment associations	
x_line_credit_tbl	OUT	PL/SQL Table	X		Returns the processed sales credits.	
x_line_credit_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed sales credits for the sales order if p_return_values was set to FND_API.G_TRUE	
x_lot_serial_tbl	OUT	PL/SQL Table	X		Returns the processed lot serials.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_lot_serial_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed lot serials if p_return_values was set to FND_API.G_TRUE	
x_action_request_tbl	OUT	PL/SQL Table	X		Returns the status for each of the action requests that were passed to process order.	

Setting Up the Get_Order Procedure

Before using the API, set up and activate the following parameters:

- Operating Unit. For more information please refer to the section on Multi-Org Access Control, page 7-112.
- Version number
- Initialize message list
- Pass the header ID of the order to be queried

Parameter Descriptions

The following chart describes all parameters used by the public API OE_ORDER_PUB.GET_ORDER. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

OE_ORDER_PUB.GET_ORDER

Parameter	Usage	Type	Required	Optional	Description	Default
P_Operating_Unit	IN	VARCHAR2		X	Operating Unit Name	
P_org_id	IN	NUMBER		X	Internal id for Operating Unit	
p_api_version_number	IN	NUMBER	X		Used to compare the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.	
p_init_msg_list	IN	Varchar2		X	Requests that the API initialize the message list on your behalf.	FND_API.G_FALSE

Parameter	Usage	Type	Required	Optional	Description	Default
p_return_values	IN	Varchar2		X	Requests that the API send back the values on your behalf.	FND_API.G_FALSE
p_commit	IN	Varchar2		X	Requests that the API update information for you after it completes its function.	FND_API.G_FALSE

Parameter	Usage	Type	Required	Optional	Description	Default
x_return_status	OUT	Varchar2		X	<p>Returns the status, indicates whether the request was processed successfully or not.</p> <p>Valid values include:</p> <p><i>Success:</i> FND_API. G_RET_STS_SUCCESS</p> <p><i>Error:</i> FND_API. G_RET_STS_ERROR</p> <p><i>Unexpected Error:</i> FND_API. G_RET_STS_UNEXP_ERROR</p>	

Parameter	Usage	Type	Required	Optional	Description	Default
x_msg_count	OUT	NUMBER		X	<p>Indicates number of error messages API has encountered.</p> <p>If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call OE_MSG_PUB.GET. This api can be called with the following parameter values:</p> <p>p_msg_index => 1</p> <p>p_encoded => F</p> <p>p_data => 1_message</p> <p>p_msg_index_out => 1_msg_index_out</p> <p>where 1_message and</p>	

Parameter	Usage	Type	Required	Optional	Description	Default
					1_msg_in ex_out should be local variables of types Varchar2(2 000) and Number respectivel y.	
x_msg_data	OUT	Varchar2			Displays error message text. If the x_msg_coun t is equal to 1, then this contains the actual message.	
p_header_id	IN	NUMBER	X		Parameter to identify the sales order that is to be queried.	FND_API. G_MISS_N UM
p_header	IN	Varchar2		X		
x_header_rec	OUT	PL/SQL Table	X		Returns the queried order header record for the sales order.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_val_rec	OUT	PL/SQL Table	X		Returns the values for the queried order header record for the sales order if p_return_values was set to FND_API.G_TRUE	
x_header_adj_tbl	OUT	PL/SQL Table	X		Returns the queried header price adjustments for the sales order.	
x_header_adj_val_tbl	OUT	PL/SQL Table	X		Returns the values for the queried header price adjustments for the sales order if p_return_values was set to FND_API.G_TRUE	
x_header_purchase_att_tbl	OUT	PL/SQL Table	X			
x_header_adj_att_tbl	OUT	PL/SQL Table	X			

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_adj_assoc_tbl	OUT	PL/SQL Table	X			
x_header_sales_credit_tbl	OUT	PL/SQL Table	X		Returns the queried header sales credits for the sales order.	
x_header_sales_credit_val_tbl	OUT	PL/SQL Table	X		Returns the values for the queried header sales credits for the sales order if p_return_values was set to FND_API.G_TRUE	
x_Header_Payment_tbl	OUT	Header_Payment_Tbl_Type				
x_Header_Payment_val_tbl	OUT	Header_Payment_Val_Tbl_Type				
x_line_tbl	OUT	PL/SQL Table	X		Returns the queried order lines for the sales order.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_line_val_tbl	OUT	PL/SQL Table	X		Returns the values for the queried order lines for the sales order if p_return_values was set to FND_API.G_TRUE	
x_line_adj_tbl	OUT	PL/SQL Table	X		Returns the queried line price adjustments for the sales order	
x_line_adj_val_tbl	OUT	PL/SQL Table	X		Returns the values for the queried line price adjustments for the sales order if p_return_values was set to FND_API.G_TRUE	
x_line_price_att_tbl	OUT	PL/SQL Table	X			
x_line_adj_att_tbl	OUT	PL/SQL Table	X			
x_line_adj_assoc_tbl	OUT	PL/SQL Table	X			

Parameter	Usage	Type	Required	Optional	Description	Default
x_line_scredit_tbl	OUT	PL/SQL Table	X		Returns the queried line sales credits for the sales order.	
x_line_scredit_val_tbl	OUT	PL/SQL Table	X		Returns the values for the queried line sales credits for the sales order if p_return_values was set to FND_API.G_TRUE	
x_Line_Payment_tbl	OUT	OUT Line_Payment_Tbl_Type				
x_Line_Payment_val_tbl	OUT	OUT Line_Payment_Val_Tbl_Type				
x_lot_serial_tbl	OUT	PL/SQL Table	X		Returns the queried lot serials for the sales order.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_lot_serial_val_tbl	OUT	PL/SQL Table	X		Returns the values for the queried lot serials for the sales order if p_return_values was set to FND_API.G_TRUE	

Setting Up the Lock_Order Procedure

Before using the API, set up and activate the following parameters:

- Operating Unit. For more information please refer to the section on Multi-Org Access Control, page 7-112.
- Version number
- Initialize message list
- Query all the entities for the sales order using the Get_Order procedure.
- Set the operation parameter to OE_GLOBALS.G_OPR_LOCK only on the entity records to be locked in this call.

Parameter Descriptions

The following chart describes all parameters used by the public API OE_ORDER_PUB.LOCK_ORDER. All of the inbound and outbound parameters are listed. The OUT parameters have to be assigned when calling the API and are therefore marked as Required fields. Additional information on these parameters follows.

OE_ORDER_PUB.LOCK_ORDER

Parameter	Usage	Type	Required	Optional	Description	Default
P_Operating_Unit	IN	VARCHAR2		X	Operating Unit Name	
P_org_id	IN	NUMBER		X	Internal id for Operating Unit	
p_api_version_number	IN	NUMBER	X		Used to compare the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.	
p_init_msg_list	IN	Varchar2		X	Requests that the API initialize the message list on your behalf.	FND_API.G_FALSE

Parameter	Usage	Type	Required	Optional	Description	Default
p_return_values	IN	Varchar2		X	Requests that the API send back the values on your behalf.	FND_API. G_FALSE
x_return_status	OUT	Varchar2	X		<p>Returns the status, indicates whether the request was processed successfully or not.</p> <p>Valid values include:</p> <p><i>Success:</i> FND_API. G_RET_STATUS_SUCCESS</p> <p><i>Error:</i> FND_API. G_RET_STATUS_ERROR</p> <p><i>Unexpected Error:</i> FND_API. G_RET_STATUS_UNEXPECTED_ERROR</p>	

Parameter	Usage	Type	Required	Optional	Description	Default
x_msg_count	OUT	NUMBER	X		<p>Indicates number of error messages API has encountered.</p> <p>If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call OE_MSG_PUB.GET. This api can be called with the following parameter values:</p> <p>p_msg_index => 1</p> <p>p_encoded => F</p> <p>p_data => 1_message</p> <p>p_msg_index_out => 1_msg_index_out</p> <p>where 1_message and</p>	

Parameter	Usage	Type	Required	Optional	Description	Default
					1_msg_in ex_out should be local variables of types Varchar2(2 000) and Number respectivel y.	
x_msg_data	OUT	Varchar2	X		Displays error message text. If the x_msg_coun t is equal to 1, then this contains the actual message.	
p_header_rec	IN	PL/SQL Record		X	Use this parameter to send in operations on the order header entity.	G_MISS_H EADER_R EC
p_header_val_rec	IN	PL/SQL Record		X	Use this parameter to send in the display values for fields on the order header entity.	G_MISS_H EADER_V AL_REC

Parameter	Usage	Type	Required	Optional	Description	Default
p_header_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order price adjustment entities.	G_MISS_HEADER_ADJ_TBL
p_header_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the order price adjustment entities.	G_MISS_HEADER_ADJ_VAL_TBL
p_header_price_attributes_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order pricing Attributes entities.	G_MISS_HEADER_PRICE_ATTRIBUTES_TBL
p_header_adj_attributes_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order Adjustment Attributes entities.	G_MISS_HEADER_ADJ_ATTRIBUTES_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_header_adj_assoc_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order Adjustment Associations entities.	G_MISS_HEADER_ADJ_ASSOC_TBL
p_header_sales_credit_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order sales credits.	G_MISS_HEADER_SALES_CREDIT_TBL
p_header_sales_credit_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the order sales credits.	G_MISS_HEADER_SALES_CREDIT_VAL_TBL
p_line_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the order lines.	G_MISS_LINE_TBL
p_line_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the order lines.	G_MISS_LINE_VAL_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_line_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the line price adjustment entities.	G_MISS_LI NE_ADJ_TBL
p_line_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the line price adjustment entities.	G_MISS_LI NE_ADJ_VAL_TBL
p_line_adj_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the Line pricing Attributes entities.	G_MISS_LI NE_PRICE_ATT_TBL
p_line_adj_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the Line Adjustment Attributes entities.	G_MISS_LI NE_ADJ_ATT_TBL

Parameter	Usage	Type	Required	Optional	Description	Default
p_line_adj_ assoc_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the Line Adjustment Association entities.	G_MISS_LI NE_ADJ_A SSOC_TBL
p_line_scre dit_tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the line sales credits.	G_MISS_LI NE_SCRE DIT_TBL
p_line_scre dit_val_tbl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the line sales credits.	G_MISS_LI NE_SCRE DIT_VAL_ TBL
p_lot_serial_ tbl	IN	PL/SQL Table		X	Use this parameter to send in operations on the lot serials.	G_MISS_L OT_SERIA L_TBL
p_lot_lot_s erial_val_t bl	IN	PL/SQL Table		X	Use this parameter to send in the display values for fields on the lot serials.	G_MISS_L OT_SERIA L_VAL_TB L

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_rec	OUT	PL/SQL Table	X		Returns the processed order header record.	
x_header_val_rec	OUT	PL/SQL Table	X		Returns the values for the processed order header record for the sales order if p_return_values was set to FND_API.G_TRUE	
x_header_adj_tbl	OUT	PL/SQL Table	X		Returns the records for the processed price adjustments	
x_header_adj_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed header price adjustments if p_return_values was set to FND_API.G_TRUE	

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_p rice_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header pricing attributes	
x_header_a dj_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header adjustment attributes	
x_header_a dj_assoc_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header adjustment associations	
x_header_s credit_tbl	OUT	PL/SQL Table	X		Returns the records for the processed header adjustment associations	

Parameter	Usage	Type	Required	Optional	Description	Default
x_header_s credit_val_ tbl	OUT	PL/SQL Table	X		Returns the values for the processed header sales credits if p_return_v alues was set to FND_API.G_TRUE	
x_line_tbl	OUT	PL/SQL Table	X		Returns the processed order lines.	
x_line_val_ tbl	OUT	PL/SQL Table	X		Returns the values for the processed order lines if p_return_v alues was set to FND_API.G_TRUE	
x_line_adj_ tbl	OUT	PL/SQL Table	X		Returns the processed line price adjustments for the sales order	

Parameter	Usage	Type	Required	Optional	Description	Default
x_line_adj_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed line price adjustments for the sales order if p_return_values was set to FND_API.G_TRUE	
x_line_price_att_tbl	OUT	PL/SQL Table	X		Returns the records for the processed line adjustment attributes	
x_line_adj_att_tbl	OUT	PL/SQL Table	X			
x_line_adj_assoc_tbl	OUT	PL/SQL Table	X		Returns the records for the processed line adjustment associations	
x_line_credit_tbl	OUT	PL/SQL Table	X		Returns the processed sales credits.	

Parameter	Usage	Type	Required	Optional	Description	Default
x_line_credit_val_tbl	OUT	PL/SQL Table	X	X	Returns the values for the processed sales credits for the sales order if p_return_values was set to FND_API.G_TRUE	
x_lot_serial_tbl	OUT	PL/SQL Table	X		Returns the processed lot serials.	
x_lot_serial_val_tbl	OUT	PL/SQL Table	X		Returns the values for the processed lot serials if p_return_values was set to FND_API.G_TRUE	

PL/SQL Record Structures

All the columns on the underlying order management tables are available on the record types that can be passed to the process order API. However, callers cannot provide values for certain columns via the public process order API. Such columns are marked as *Derived* columns as these are derived internally and updated by the API. If user provides values for these fields, it could result in data corruption.

For each column (if the record type is not a value record type, val_rec_type), the following information has been documented:

- **Datatype:** data type for this field
- **Required:** X if required at entry, C if conditionally required

- **Required at Booking:** X if required at booking, C if required at booking based on some other conditions.
- **Derived:** For internal use, users cannot update these fields.
- **Optional:** if users can either provide values for these fields or could be defaulted.

The value record structures have all fields that can be optionally passed instead of the identifier. If the identifier field is not passed when required, the value field can be passed instead and the value will be resolved into the identifier.

These record and table structures are defined in the specifications of the process order API, OE_ORDER_PUB.

Header_Rec_Type

Please refer to the table below.

- Please note that columns attribute1 - attribute15 are currently defined as VARCHAR2(240), but only the first 150 characters are passed to Oracle Receivables during Invoice Interface. These columns contain Additional Header Information Descriptive Flexfield data

OE_ORDER_HEADERS_ALL

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
accounting_rule_id		NUMBER				X
agreement_id		NUMBER		C - if order type requires		X
attribute1...		VARCHAR2(240)				X
attribute20						
booked_flag		VARCHAR2(1)			X	

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
cancelled_flag		VARCHAR2(1)				X
context		VARCHAR2(30)				X
conversion_rate		NUMBER		C - if conversion type is <i>User</i>		X
conversion_rate_date		DATE		C - if conversion type is <i>User</i>		X
conversion_type_code		VARCHAR2(30)		C - if order currency not same as SOB currency		X
customer_preference_set_code		VARCHAR2(30)				X
created_by		NUMBER			X	
creation_date		DATE			X	
cust_po_number		VARCHAR2(50)		C - if order type requires		X

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
deliver_to_contact_id		NUMBER				X
deliver_to_org_id		NUMBER				X
demand_class_code		VARCHAR2(30)				X
earliest_schedule_limit		NUMBER			X	
expiration_date		DATE			X	
fob_point_code		VARCHAR2(30)				X
freight_carrier_code		VARCHAR2(30)			X	
freight_terms_code		VARCHAR2(30)				X
global_attribute1...20		VARCHAR2(240)				X
global_attribute_category		VARCHAR2(30)				X
tp_context		VARCHAR2(30)				X
tp_attribute1....		VARCHAR2(240)				X
tp_attribute20						

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
header_id		NUMBER	C - for UPDATE & DELETE operations			X
invoice_to_contact_id		NUMBER				X
invoice_to_org_id		NUMBER		X		
invoicing_rule_id		NUMBER				X
last_updated_by		NUMBER			X	
last_update_date		DATE			X	
last_update_login		NUMBER			X	
latest_schedule_limit		NUMBER				X
open_flag		VARCHAR2(1)			X	
order_category_code		VARCHAR2(30)			X	
ordered_date		DATE		X		X
order_date_type_code		VARCHAR2(30)				X

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
order_number		NUMBER	C - for manual order types			X
order_source_id		NUMBER				X
order_type_id		NUMBER	X			
org_id		NUMBER			X	
orig_sys_document_ref		VARCHAR2(50)				X
partial_shipments_allowed		VARCHAR2(1)			X	
payment_term_id		NUMBER		C - for order lines, not return lines		X
price_list_id		NUMBER		X		X
pricing_date		DATE	X			X
program_application_id		NUMBER				X
program_id		NUMBER				X

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
program_update_date		DATE				X
request_date		DATE				X
request_id		NUMBER				X
return_reason_code		VARCHAR2(30)				X
salesrep_id		NUMBER		X		X
sales_channel_code		VARCHAR2(30)				X
shipment_priority_code		VARCHAR2(30)				X
shipping_method_code		VARCHAR2(30)				X
ship_from_org_id		NUMBER		C - for return lines		
ship_tolerance_above		NUMBER				X
ship_tolerance_below		NUMBER				X
ship_to_contact_id		NUMBER				X

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
ship_to_org_id		NUMBER		C - for order lines, not return lines		
sold_from_org_id		NUMBER		X		
sold_to_contact_id		NUMBER				X
sold_to_org_id		NUMBER		X		X
source_document_id		NUMBER				X
source_document_type_id		NUMBER				X
tax_exempt_flag		VARCHAR2(30)		X		X
tax_exempt_number		VARCHAR2(50)				X
tax_exempt_reason_code		VARCHAR2(30)	C - if tax exempt flag is <i>Exempt</i>			X
tax_point_code		VARCHAR2(30)				for future use
transactional_curr_code		VARCHAR2(3)	X			

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
version_number		NUMBER			X	
return_status		VARCHAR2(1)			X	
db_flag		VARCHAR2(1)			X	
operation		VARCHAR2(30)	X			
first_ack_code		VARCHAR2(30)			X	
first_ack_date		DATE			X	
last_ack_code		VARCHAR2(30)			X	
last_ack_date		DATE			X	
change_reason		VARCHAR2(30)	C -if constraints setup requires reason			X
change_comments		VARCHAR2(2000)				X
change_sequence		VARCHAR2(50)			X	
change_request_code		VARCHAR2(30)			X	
ready_flag		VARCHAR2(1)			X	

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
status_flag		VARCHAR2(1)			X	
force_apply_flag		VARCHAR2(1)			X	
drop_ship_flag		VARCHAR2(1)			X	
customer_payment_term_id		Obsolete	Obsolete	Obsolete	Obsolete	Obsolete
payment_type_code		VARCHAR2(30)				X
payment_amount		NUMBER		C - if payment type other than Credit Card		
check_number		VARCHAR2(50)		C - if payment type is <i>Check</i>		
credit_card_code		VARCHAR2(80)				X
credit_card_holder_name		VARCHAR2(80)		C - if payment type is <i>Credit Card</i>		

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
credit_card_number		VARCHAR2(80)		C - if payment type <i>Credit Card</i>		
credit_card_expiration_date		DATE		C - if payment type <i>Credit Card</i>		
credit_card_approval_code		VARCHAR2(80)			X	X
credit_card_approval_date		DATE			X	
shipping_instructions		VARCHAR2(2000)				X
packing_instructions		VARCHAR2(2000)				X
flow_status_code		VARCHAR2(30)			X	
booked_date		DATE			X	
marketing_source_code_id		NUMBER				X
xml_message_id		NUMBER				X

Parameter	Description	Datatype	Required	Required at Booking	Derived	Optional
xml_transaction_type_code		VARCHAR2(30)				X
IB_OWNER	Install Base Owner Identifier	Varchar2(60)				
IB_OWNER_CODE		Varchar2(60)				
IB_INSTALL_LOCATION	Install Location Identifier	Varchar2(60)				
IB_INSTALL_LOCATION		Varchar2(60)				
IB_CURRENT_LOCATION	Install Current Location Identifier	Varchar2(60)				
IB_CURRENT_LOCATION		Varchar2(60)				
END_CUSTOMER_ID	Customer Id of End Customer	Number				
END_CUSTOMER_CONTACT_ID	Contact Id of End Customer Contact	Number				
END_CUSTOMER_SITE_ID	Site Use Id of End Customer Site	Number				

TYPE Header_Tbl_Type IS TABLE OF Header_Rec_Type
INDEX BY BINARY_INTEGER;
Header_Rec_Type

Header_Rec_Type

Parameter	Value
ship_to_customer_id	NUMBER
invoice_to_customer_id	NUMBER
deliver_to_customer_id	NUMBER
IB_OWNER	VARCHAR2(60)
IB_INSTALLED_AT_LOCATION	VARCHAR2(60)
IB_CURRENT_LOCATION	VARCHAR2(60)
END_CUSTOMER_ID	NUMBER
END_CUSTOMER_CONTACT_ID	NUMBER
END_CUSTOMER_SITE_USE_ID	NUMBER
sold_to_party_id	NUMBER
sold_to_org_contact_id	NUMBER
ship_to_party_id	NUMBER
ship_to_party_site_id	NUMBER
ship_to_party_site_use_id	NUMBER
deliver_to_party_id	NUMBER
deliver_to_party_site_id	NUMBER
deliver_to_party_site_use_id	NUMBER

Parameter	Value
ship_to_customer_id	NUMBER
invoice_to_party_id	NUMBER
invoice_to_party_site_id	NUMBER
invoice_to_party_site_use_id	NUMBER
ship_to_customer_party_id	NUMBER
deliver_to_customer_party_id	NUMBER
invoice_to_customer_party_id	NUMBER
ship_to_org_contact_id	NUMBER
deliver_to_org_contact_id	NUMBER
invoice_to_org_contact_id	NUMBER
sold_to_party_number	varchar2(30)
ship_to_party_number	varchar2(30)
invoice_to_party_number	varchar2(30)
deliver_to_party_number	varchar2(30)

Header_Val_Rec_Type

For column descriptions, please refer to the Order Management e-TRM for the table:
Header_Val_Rec_Type

Parameter accounting_rule
agreement
conversion_type

Parameter accounting_rule

deliver_to_address1

deliver_to_address2

deliver_to_address3

deliver_to_address4

deliver_to_contact

deliver_to_location

deliver_to_org

demand_class

fob_point

freight_terms

invoice_to_address1

invoice_to_address2

invoice_to_address3

invoice_to_address4

invoice_to_contact

invoice_to_location

invoice_to_org

invoicing_rule

order_source

order_type

Parameter accounting_rule

payment_term

price_list

return_reason

salesrep

shipment_priority

ship_from_address1

ship_from_address2

ship_from_address3

ship_from_address4

ship_from_location

ship_from_org

ship_to_address1

ship_to_address2

ship_to_address3

ship_to_address4

ship_to_contact

ship_to_location

ship_to_org

sold_to_contact

sold_to_org

Parameter accounting_rule

sold_from_org

tax_exempt

tax_exempt_reason

tax_point

customer_payment_term - *obsolete*

payment_type

credit_card

status

freight_carrier

shipping_method

order_date_type

customer_number

sales_channel

ship_to_customer_name VARCHAR2(360)

invoice_to_customer_name VARCHAR2(360)

ship_to_customer_number VARCHAR2(50)

invoice_to_customer_number VARCHAR2(50)

deliver_to_customer_number VARCHAR2(50)

deliver_to_customer_name VARCHAR2(360)

end_customer_name VARCHAR2(360)

Parameter accounting_rule

end_customer_number VARCHAR2(50)

end_customer_contact VARCHAR2(360)

end_cust_contact_last_name VARCHAR2(240)

end_cust_contact_first_name VARCHAR2(240)

end_customer_site_address1 VARCHAR2(240)

end_customer_site_address2 VARCHAR2(240)

end_customer_site_address3 VARCHAR2(240)

end_customer_site_address4 VARCHAR2(240)

end_customer_site_state VARCHAR2(240)

end_customer_site_country VARCHAR2(240)

end_customer_site_location VARCHAR2(240)

end_customer_site_zip VARCHAR2(240)

end_customer_site_county VARCHAR2(240)

end_customer_site_province VARCHAR2(240)

end_customer_site_city VARCHAR2(240)

end_customer_site_postal_code VARCHAR2(240)

ib_owner_dsp VARCHAR2(60)

ib_installed_at_location_dsp VARCHAR2(60)

ib_current_location_dsp VARCHAR2(60)

TYPE Header_Val_Tbl_Type IS TABLE OF Header_Val_Rec_Type
INDEX BY BINARY_INTEGER;

Header_Adj_Rec_Type

For column descriptions, please refer to the Order Management e-TRM for the table
OE_PRICE_ADJUSTMENTS

OE_PRICE_ADJUSTMENTS

Parameter	Datatype	Required	Derived	Optional
attribute1 - attribute15	VARCHAR2(240)			
automatic_flag	VARCHAR2(1)	X	qp_list_lines	
context	VARCHAR2(30)			X
created_by	NUMBER		X	
creation_date	DATE		X	
discount_id	NUMBER		X	
discount_line_id	NUMBER		X	
header_id	NUMBER	X		
last_updated_by	NUMBER		X	
last_update_date	DATE		X	
last_update_logi n	NUMBER		X	
percent	NUMBER		X	
price_adjustmen t_id	NUMBER	C -for update and delete		X
program_applica tion_id	NUMBER			X
program_id	NUMBER			X

Parameter	Datatype	Required	Derived	Optional
program_update_date	DATE			X
request_id	NUMBER			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		
line_index	NUMBER			X
orig_sys_discount_ref	VARCHAR2(50)			X
change_request_code	VARCHAR2(30)		X	
status_flag	VARCHAR2(1)		X	
list_header_id	NUMBER	C - not required only for tax records		
list_line_id	NUMBER	C - not required only for tax records		
list_line_type_code	VARCHAR2(30)		X	
modifier_mechanism_type_code	VARCHAR2(30)		X	
modified_from	NUMBER		X	
modified_to	NUMBER		X	
updated_flag	VARCHAR2(1)		X	

Parameter	Datatype	Required	Derived	Optional
update_allowed	VARCHAR2(1)		X	
applied_flag	VARCHAR2(1)			
change_reason_code	VARCHAR2(30)			X
change_reason_text	VARCHAR2(2000)			X
operand	NUMBER			X
operand_per_qty	NUMBER			X
arithmetic_operator	VARCHAR2(30)		X	
cost_id	NUMBER		X	
tax_code	VARCHAR2(30)			
tax_exempt_flag	VARCHAR2(1)			
tax_exempt_number	VARCHAR2(80)			
tax_exempt_reason_code	VARCHAR2(30)			
parent_adjustment_id	NUMBER	X -for update and delete		
invoiced_flag	VARCHAR2(1)		X	
estimated_flag	VARCHAR2(1)		X	
inc_in_sales_performance	VARCHAR2(1)		X	

Parameter	Datatype	Required	Derived	Optional
split_action_code	VARCHAR2(30)			X
adjusted_amount	NUMBER		X	
adjusted_amount_per_qty	NUMBER		X	
pricing_phase_id	NUMBER		X	
charge_type_code	VARCHAR2(30)		X	
charge_subtype_code	VARCHAR2(30)		X	
list_line_no	VARCHAR2(240)		X	
source_system_code	VARCHAR2(30)		X	
benefit_qty	NUMBER		X	
benefit_uom_code	VARCHAR2(3)		X	
print_on_invoice_flag	VARCHAR2(1)		X	
expiration_date	DATE		X	
rebate_transaction_type_code	VARCHAR2(30)		X	
rebate_transaction_reference	VARCHAR2(80)		X	
rebate_payment_system_code	VARCHAR2(30)		X	

Parameter	Datatype	Required	Derived	Optional
redeemed_date	DATE		X	
redeemed_flag	VARCHAR2(1)		X	
accrual_flag	VARCHAR2(1)		X	
range_break_quantity	NUMBER		X	
accrual_conversion_rate	NUMBER		X	
pricing_group_sequence	NUMBER		X	
modifier_level_code	VARCHAR2(30)		X	
price_break_type_code	VARCHAR2(30)		X	
substitution_attribute	VARCHAR2(30)		X	
proration_type_code	VARCHAR2(30)		X	
credit_or_charge_flag	VARCHAR2(1)		X	
include_on_returns_flag	VARCHAR2(1)		X	
ac_attribute1...	VARCHAR2(240)			X
ac_attribute15)			
ac_context	VARCHAR2(150)			X
)			

TYPE Header_Adj_Tbl_Type IS TABLE OF Header_Adj_Rec_Type
INDEX BY BINARY_INTEGER;

For column descriptions, please refer to the Order Management e-TRM for the table:Header_Adj_Val_Rec_Type

Header_Adj_Val_Rec_Type

Header_Adj_Val_Rec_Type
Parameter
discount
list_name

TYPE Header_Adj_Val_Tbl_Type IS TABLE OF Header_Adj_Val_Rec_Type
INDEX BY BINARY_INTEGER;

Header_Price_Att_Rec_Type

For column descriptions, please refer to the Order Management table
OE_ORDER_PRICE_ATTRIBS

OE_ORDER_PRICE_ATTRIBS

Parameter	Datatype	Required	Derived	Optional
order_price_attri b_id	NUMBER	X -for update and delete		X
header_id	NUMBER	X		
creation_date	DATE		X	
created_by	NUMBER		X	
last_update_date	DATE		X	
last_updated_by	NUMBER		X	
last_update_logi n	NUMBER		X	

Parameter	Datatype	Required	Derived	Optional
program_application_id	NUMBER			X
program_id	NUMBER			X
program_update_date	DATE			X
request_id	NUMBER			X
flex_title	VARCHAR2(60)	X		
pricing_context	VARCHAR2(30)	X		
pricing_attribute1... pricing_attribute100	VARCHAR2(240))	at least one of these columns should have a value		X
context	VARCHAR2(30)			X
attribute1... attribute15	VARCHAR2(240))			X
Override_Flag	VARCHAR2(1)			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		

TYPE Header_Price_Att_Tbl_Type is TABLE of Header_Price_Att_rec_Type

INDEX by BINARY_INTEGER

Header_Adj_Att_Rec_Type

For column descriptions, please refer to the Order Management table
OE_PRICE_ADJ_ATTRIBS.

OE_PRICE_ADJ_ATTRIBS

Parameter	Datatype	Required	Derived	Optional
price_adj_attrib_id	NUMBER	C -for update and delete		
price_adjustment_id	NUMBER	C - if adj_index not passed		
cdj_index	NUMBER	C - if price_adjustment_id not passed		
flex_title	VARCHAR2(60)	X		
pricing_context	VARCHAR2(30)	X		
pricing_attribute	VARCHAR2(30)	X		
creation_date	DATE		X	
created_by	NUMBER		X	
last_update_date	DATE		X	
last_updated_by	NUMBER		X	
last_update_log_id	NUMBER		X	
program_application_id	NUMBER			X
program_id	NUMBER			X
program_update_date	DATE			X
request_id	NUMBER			X
pricing_attr_value_from	VARCHAR2(240)	X		

Parameter	Datatype	Required	Derived	Optional
pricing_attr_value_to	VARCHAR2(240)			X
comparison_operator	VARCHAR2(30)			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)			

TYPE Header_Adj_Att_Rec_Type is TABLE of Header_Adj_Att_rec_Type
INDEX by BINARY_INTEGER;

Type Header_Adj_Assoc_Rec_Type is RECORD

For column descriptions, please refer to the Order Management TRM for the table:
OE_PRICE_ADJ ASSOCS

OE_PRICE_ADJ ASSOCS

Parameter	Datatype	Required	Derived	Optional
price_adj_assoc_id	NUMBER	C - for update and delete		
line_id	NUMBER			
Line_Index	NUMBER			
price_adjustment_id	NUMBER	C - if adj_index not passed		
Adj_index	NUMBER	C - if price_adjustment_id not passed		
rltd_Price_Adj_Id	NUMBER			X

Parameter	Datatype	Required	Derived	Optional
Rltd_Adj_Index	NUMBER			X
creation_date	DATE		X	
created_by	NUMBER		X	
last_update_date	DATE		X	
last_updated_by	NUMBER		X	
last_update_logi n	NUMBER		X	
program_applica tion_id	NUMBER			X
program_id	NUMBER			X
program_update _date	DATE			X
request_id	NUMBER			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)			X

TYPE Header_Adj_Assoc_Tbl_Type is TABLE of Header_Adj_Assoc_rec_Type
INDEX by BINARY_INTEGER;

TYPE Header_Scredit_Rec_Type IS RECORD

For column descriptions, please refer to the Order Management TRM for the table:
OE_SALES_CREDITS

OE_SALES_CREDITS

Parameter	Datatype	Required	Derived	Optional
attribute1... attribute15	VARCHAR2(240)			X
context	VARCHAR2(30)			X
created_by	NUMBER		X	
creation_date	DATE		X	
dw_update_advice_flag	VARCHAR2(1)			X
header_id	NUMBER			X
last_updated_by	NUMBER		X	
last_update_date	DATE		X	
last_update_logi n	NUMBER		X	
percent	NUMBER	X		
salesrep_id	NUMBER	X		
sales_credit_type _id	NUMBER	X		
sales_credit_id	NUMBER			X
wh_update_date	DATE			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		

Parameter	Datatype	Required	Derived	Optional
orig_sys_credit_ref	VARCHAR2(50)			X
change_request_code	VARCHAR2(30)		X	
status_flag	VARCHAR2(1)		X	

TYPE Header_Scredit_Tbl_Type IS TABLE OF Header_Scredit_Rec_Type

INDEX BY BINARY_INTEGER;

TYPE Header_Scredit_Val_Rec_Type IS RECORD

(salesrep, sales_credit_type);

TYPE Header_Scredit_Val_Tbl_Type IS TABLE OF Header_Scredit_Val_Rec_Type

INDEX BY BINARY_INTEGER;

Line_Rec_Type

For column descriptions, please refer to the Order Management e-TRM for the table OE_ORDER_LINES_ALL

- Please note that columns attribute1 - attribute15 are currently defined as VARCHAR2(240), but only the first 150 characters are passed to Oracle Receivables during Invoice Interface. These columns contain Additional Order Information Descriptive Flexfield data

OE_ORDER_LINES_ALL

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
accounting_rule_id	NUMBER					X
actual_arrival_date	DATE				X	

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
actual_shipment_date	DATE				X	
agreement_id	NUMBER					X
arrival_set_id	NUMBER					X
ato_line_id	NUMBER				X	
attribute1..attribute20	VARCHAR2(240)					X
authorized_to_ship_flag	VARCHAR2(1)					X
auto_selected_quantity	NUMBER				X	
booked_flag	VARCHAR2(1)				X	
cancelled_flag	VARCHAR2(1)				X	
cancelled_quantity	NUMBER				X	
cancelled_quantity2	NUMBER				X	
commitment_id	NUMBER					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
component_code	VARCHAR2(100)					X
component_number	NUMBER				X	
component_sequence_id	NUMBER				X	
config_header_id	NUMBER				X	X
config_revision_nbr	NUMBER				X	X
config_display_sequence	NUMBER				X	
configuration_id	NUMBER				X	X
context	VARCHAR2(30)					X
created_by	NUMBER				X	
creation_date	DATE				X	
credit_invoice_line_id	NUMBER				X	
customer_dock_code	VARCHAR2(50)					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
customer_job	VARCHAR2(50)					X
customer_production_line	VARCHAR2(50)					X
customer_trx_line_id	NUMBER				X	
cust_model_serial_number	VARCHAR2(50)					X
cust_purchase_order_number	VARCHAR2(50)					X
cust_production_sequence_number	VARCHAR2(50)					X
delivery_lead_time	NUMBER					X
deliver_to_contact_id	NUMBER					X
deliver_to_org_id	NUMBER					X
demand_bucket_type_code	VARCHAR2(30)					X
demand_class_code	VARCHAR2(30)					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
dep_plan_required_lag	VARCHAR2(1)					X
earliest_acceptable_date	DATE				X	
end_item_unit_number	VARCHAR2(30)		C - depends on the item			X
explosion_date	DATE				X	
fob_point_code	VARCHAR2(30)					X
freight_carrier_code	VARCHAR2(30)				X	
freight_terms_code	VARCHAR2(30)					X
fulfilled_quantity	NUMBER				X	
fulfilled_quantity2	NUMBER				X	
global_attribute1...20	VARCHAR2(240)					X
global_attribute_category	VARCHAR2(30)					X
header_id	NUMBER					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
industry_attribute1...30	VARCHAR2(240)					X
industry_context	VARCHAR2(30)					X
tp_context	VARCHAR2(30)					X
tp_attribute1...20	VARCHAR2(240)					X
intermed_ship_to_org_id	NUMBER					X
intermed_ship_to_contact_id	NUMBER					X
inventory_item_id	NUMBER		X			
invoice_interface_status_code	VARCHAR2(30)				X	X
invoice_to_contact_id	NUMBER					X
invoice_to_org_id	NUMBER			X		X
invoicing_rule_id	NUMBER					X
ordered_item	VARCHAR2(200)		C: for <i>Generic Items</i>			

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
item_revision	VARCHAR2(3)					X
item_type_code	VARCHAR2(30)				X	
last_updated_by	NUMBER				X	
last_update_date	DATE				X	
last_update_login	NUMBER				X	
latest_acceptable_date	DATE					X
line_category_code	VARCHAR2(30)				X	X
line_id	NUMBER		C - for UPDATE & DELETE operations			
line_number	NUMBER					X
line_type_id	NUMBER		X			X
link_to_line_ref	VARCHAR2(50)					X
link_to_line_id	NUMBER				X	

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
link_to_line_index	NUMBER					X
model_group_number	NUMBER				X	
mfg_component_sequence_id	NUMBER				X	
open_flag	VARCHAR2(1)				X	
option_flag	VARCHAR2(1)				X	
option_number	NUMBER				X	
ordered_quantity	NUMBER			X		X
ordered_quantity2	NUMBER					X
order_quantity_uom	VARCHAR2(3)			X		X
ordered_quantity_uom2	VARCHAR2(3)				X	
org_id	NUMBER				X	
orig_sys_document_ref	VARCHAR2(50)					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
orig_sys_line_ref	VARCHAR2(50)					X
over_ship_reason_code	VARCHAR2(30)				X	
over_ship_resolved_flag	VARCHAR2(1)					X
payment_term_id	NUMBER			C- for order lines, not return lines		X
planning_priority	NUMBER				X	
preferred_grade	VARCHAR2(4)					X
price_list_id	NUMBER			X		
pricing_attribute10	VARCHAR2(240)				X	
pricing_attribute10						
pricing_context	VARCHAR2(240)				X	
pricing_date	DATE					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
pricing_quantity	NUMBER				X	
pricing_quantity uom	VARCHAR2(3)				X	
program_application_id	NUMBER					X
program_id	NUMBER					X
program_update_date	DATE					X
project_id	NUMBER					X
promise_date	DATE					X
re_source_flag	VARCHAR2(1)				X	
reference_customer_trx_line_id	NUMBER				X	
reference_header_id	NUMBER				X	
reference_line_id	NUMBER				X	
reference_type	VARCHAR2(30)				X	

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
request_date	DATE					X
request_id	NUMBER					X
reserved_quantity	NUMBER					X
return_attribute1...	VARCHAR2(240)					X
return_attribute20						
return_content	VARCHAR2(30)					X
return_reason_code	VARCHAR2(30)		C - for return lines only			
rla_schedule_type_code	VARCHAR2(30)					X
salesrep_id	NUMBER					X
schedule_arrival_date	DATE					X
schedule_shipping_date	DATE					X
schedule_action_code	VARCHAR2(30)					X
schedule_status_code	VARCHAR2(30)				X	

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
shipment_number	NUMBER				X	
shipment_priority_code	VARCHAR2(30)					X
shipped_quantity	NUMBER				X	
shipped_quantity2	NUMBER				X	
shipping_interface_flag	VARCHAR2(1)				X	
shipping_method_code	VARCHAR2(30)					X
shipping_quantity	NUMBER				X	
shipping_quantity2	NUMBER				X	
shipping_quantity_uom	VARCHAR2(3)				X	
shipping_quantity_uom2	VARCHAR2(3)				X	
ship_from_org_id	NUMBER			C - for return lines		X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
ship_model_complete_flag	VARCHAR2(30)				X	
ship_set_id	NUMBER					X
fulfillment_set_id	NUMBER					X
ship_tolerance_above	NUMBER					X
ship_tolerance_below	NUMBER					X
ship_to_contact_id	NUMBER					X
ship_to_order_id	NUMBER			C - for order lines, not return lines		X
sold_to_order_id	NUMBER			X		
sold_from_org_id	NUMBER			X		
sort_order	VARCHAR2(240)				X	
source_document_id	NUMBER					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
source_document_line_id	NUMBER					X
source_document_type_id	NUMBER					X
source_type_code	VARCHAR2(30)					X
split_from_line_id	NUMBER					X
task_id	NUMBER		C - depends on the project			
tax_code	VARCHAR2(50)			X		X
tax_date	DATE			X		X
tax_exempt_flag	VARCHAR2(30)			X		X
tax_exempt_number	VARCHAR2(50)					X
tax_exempt_reason_code	VARCHAR2(30)		C -if tax status is <i>Exempt</i>			X
tax_point_code	VARCHAR2(30)					for future use
tax_rate	NUMBER				X	
tax_value	NUMBER				X	

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
top_model_line_ref	VARCHAR2(50)					X
top_model_line_id	NUMBER				X	
top_model_line_index	NUMBER					X
unit_list_price	NUMBER			X		X
unit_list_price_per_qty	NUMBER					X
unit_selling_price	NUMBER			X		X
unit_selling_price_per_qty	NUMBER					X
veh_customer_cumulative_key_id	NUMBER					X
visible_demand_flag	VARCHAR2(1)				X	
return_status	VARCHAR2(1)				X	
db_flag	VARCHAR2(1)				X	
operation	VARCHAR2(30)		X			

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
first_ack_code	VARCHAR2(30)				X	
first_ack_date	DATE				X	
last_ack_code	VARCHAR2(30)				X	
last_ack_date	DATE				X	
change_reason	VARCHAR2(30)					X
change_comments	VARCHAR2(200)					X
arrival_set	VARCHAR2(30)					X
ship_set	VARCHAR2(30)					X
fulfillment_set	VARCHAR2(30)					X
order_source_id	NUMBER					X
orig_sys_shipment_ref	VARCHAR2(50)					X
change_sequence	VARCHAR2(50)					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
change_request_code	VARCHAR2(30)				X	
status_flag	VARCHAR2(1)				X	
drop_shipping_flag	VARCHAR2(1)				X	
customer_line_number	VARCHAR2(50)					X
customer_shipment_number	VARCHAR2(50)					X
customer_item_net_price	NUMBER					X
customer_payment_term_id	Obsolete		Obsolete	Obsolete	Obsolete	Obsolete
ordered_item_id	NUMBER		C: for Customer Items			
item_identifier_type	VARCHAR2(25)		C: for <i>customer</i> and <i>generic</i> items			X
shipping_instructions	VARCHAR2(200)					X
packing_instructions	VARCHAR2(200)					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
calculate_price_flag	VARCHAR2(1)					X
invoiced_quantity	NUMBER				X	
service_transaction_reason_code	VARCHAR2(30)					X
service_transaction_comments	VARCHAR2(2000)					X
service_duration	NUMBER			C - for service lines		X
service_period	VARCHAR2(3)		C - for service lines			X
service_start_date	DATE		C - for service lines			X
service_end_date	DATE		C - for service lines			X
service_terminate_flag	VARCHAR2(1)					X
unit_list_percent	NUMBER					X
unit_selling_percent	NUMBER					X
unit_percentage_base_price	NUMBER				X	

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
service_number	NUMBER				X	
service_reference_type_code	VARCHAR2(30)		C - for service lines			X
service_reference_line_id	NUMBER		C - for service lines			X
service_reference_system_id	NUMBER					X
service_reference_order_number	NUMBER		C - for service reference type of <i>Order</i>			X
service_reference_line_number	NUMBER		C - for service reference type of <i>Order</i>			X
service_reference_shipment_number	NUMBER					X
service_reference_option_number	NUMBER					X
service_line_index	NUMBER					X
Line_set_id	NUMBER				X	
split_by	VARCHAR2(240)					X

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
Split_Acti on_Code	VARCHAR2(30)					X
shippable_ flag	VARCHAR2(1)				X	
model_re mnant_ flag	VARCHAR2(1)				X	
flow_statu s_code	VARCHAR2(30)				X	
fulfilled_ flag	VARCHAR2(1)				X	
fulfillment _method_ code	VARCHAR2(30)				X	
revenue_ amount	NUMBER				X	
marketing _source_ code_id	NUMBER					X
fulfillment _date	DATE				X	
semi_proc essed_ flag	BOOLEAN				X	
override_ atp_date_ code	Varchar2(30)		No	No	No	Y
late_dema nd_penalt y_factor	Number		No	No	No	Y

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
xml_transaction_code	Varchar2(30)					Y
IB_OWNER	Varchar2(60)	Install Base Owner Identifier				
IB_OWNER_CODE	Varchar2(60)	Install Base Owner Identifier				
IB_INSTALLED_LOCATION	Varchar2(60)	Install Location Identifier				
IB_INSTALLED_LOCATION	Varchar2(60)					
IB_CURRENT_LOCATION	Varchar2(60)	Install Current Location Identifier				
IB_CURRENT_LOCATION_CODE	Varchar2(60)	Install Current Location Identifier				
END_CUSTOMER_ID	Number	Customer Id of End Customer				
END_CUSTOMER_CONTACT_ID	Number	Contact Id of End Customer Contact				

Parameter	Datatype	Description	Required	Required at Booking	Derived	Optional
END_CUSTOMER_SITE_USE_ID	Number	Site Use Id of End Customer Site				

TYPE Line_Tbl_Type IS TABLE OF Line_Rec_Type

Index BY BINARY_INTEGER;

Line_Rec_Type

Line_Rec_Type

Parameter	Value
IB_OWNER	VARCHAR2(60)
IB_INSTALLED_AT_LOCATION	VARCHAR2(60)
IB_CURRENT_LOCATION	VARCHAR2(60)
END_CUSTOMER_ID	NUMBER
END_CUSTOMER_CONTACT_ID	NUMBER
END_CUSTOMER_SITE_USE_ID	NUMBER
ship_to_party_id	NUMBER
ship_to_party_site_id	NUMBER
ship_to_party_site_use_id	NUMBER
deliver_to_party_id	NUMBER
deliver_to_party_site_id	NUMBER
deliver_to_party_site_use_id	NUMBER

Parameter	Value
invoice_to_party_id	NUMBER
invoice_to_party_site_id	NUMBER
invoice_to_party_site_use_id	NUMBER
ship_to_customer_party_id	NUMBER
deliver_to_customer_party_id	NUMBER
invoice_to_customer_party_id	NUMBER
ship_to_org_contact_id	NUMBER
deliver_to_org_contact_id	NUMBER
invoice_to_org_contact_id	NUMBER
ship_to_party_number	varchar2(30)
invoice_to_party_number	varchar2(30)
deliver_to_party_number	varchar2(30)
retrobill_request_id	NUMBER
CONTINGENCY_ID	NUMBER
REVREC_EVENT_CODE	varchar2(30)
REVREC_EXPIRATION_DAYS	NUMBER
ACCEPTED_QUANTITY	NUMBER
ACCEPTED_BY	NUMBER
REVREC_COMMENTS	VARCHAR2(2000)
REVREC_REFERENCE_DOCUMENT	VARCHAR2(240)

Parameter	Value
REVREC_SIGNATURE	VARCHAR2(240)
REVREC_SIGNATURE_DATE	DATE
REVREC_IMPLICIT_FLAG	VARCHAR2(1)

Line_Val_Rec_Type

Line_Val_Rec_Type

Parameter
accounting_rule
agreement
commitment
deliver_to_address1
deliver_to_address2
deliver_to_address3
deliver_to_address4
deliver_to_contact
deliver_to_location
deliver_to_org
demand_class
demand_bucket_type
fob_point
freight_terms

Parameter

inventory_item

invoice_to_address1

invoice_to_address2

invoice_to_address3

invoice_to_address4

invoice_to_contact

invoice_to_location

invoice_to_org

invoicing_rule

item_type

line_type

over_ship_reason

payment_term

price_list

project

return_reason

rla_schedule_type

salesrep

shipment_priority

ship_from_address1

Parameter

ship_from_address2

ship_from_address3

ship_from_address4

ship_from_location

ship_from_org

ship_to_address1

ship_to_address2

ship_to_address3

ship_to_address4

ship_to_contact

ship_to_location

ship_to_org

source_type

intermed_ship_to_address1

intermed_ship_to_address2

intermed_ship_to_address3

intermed_ship_to_address4

intermed_ship_to_contact

intermed_ship_to_location

intermed_ship_to_org

Parameter

sold_to_org

sold_from_org

task

tax_exempt

tax_exempt_reason

tax_point

veh_cus_item_cum_key

visible_demand

customer_payment_term - *obsolete*

ref_order_number

ref_line_number

ref_shipment_number

ref_option_number

ref_invoice_number

ref_invoice_line_number

credit_invoice_number

tax_group

status

freight_carrier

shipping_method

Parameter

ship_to_customer_name VARCHAR2(360)

invoice_to_customer_name VARCHAR2(360)

ship_to_customer_number VARCHAR2(50)

invoice_to_customer_number VARCHAR2(50)

deliver_to_customer_number VARCHAR2(50)

deliver_to_customer_name VARCHAR2(360)

end_customer_name VARCHAR2(360)

end_customer_number VARCHAR2(50)

end_customer_contact VARCHAR2(360)

end_cust_contact_last_name VARCHAR2(240)

end_cust_contact_first_name VARCHAR2(240)

end_customer_site_address1 VARCHAR2(240)

end_customer_site_address2 VARCHAR2(240)

end_customer_site_address3 VARCHAR2(240)

end_customer_site_address4 VARCHAR2(240)

end_customer_site_location VARCHAR2(240)

end_customer_site_state VARCHAR2(240)

end_customer_site_country VARCHAR2(240)

end_customer_site_zip VARCHAR2(240)

end_customer_site_county VARCHAR2(240)

Parameter

end_customer_site_province VARCHAR2(240)

end_customer_site_city VARCHAR2(240)

end_customer_site_postal_code VARCHAR2(240)

ib_owner_dsp VARCHAR2(60)

ib_current_location_dsp VARCHAR2(60)

ib_installed_at_location_dsp VARCHAR2(60)

CONTINGENCY_NAME VARCHAR2(240)

CONTINGENCY_DESCRIPTION VARCHAR2(240)

EXPIRATION_EVENT_ATTRIBUTE VARCHAR2(240)

REVREC_EVENT VARCHAR2(240)

ACCEPTED_BY_DSP VARCHAR2(100)

TYPE Line_Val_Tbl_Type IS TABLE OF Line_Val_Rec_Type

INDEX BY BINARY_INTEGER;

TYPE Line_Adj_Rec_Type IS RECORD

For column descriptions, please refer to the Order Management TRM for the table
OE_PRICE_ADJUSTMENTS

OE_PRICE_ADJUSTMENTS

Parameter	Datatype	Required	Derived	Optional
attribute1.... attribute15	VARCHAR2(240)			
automatic_flag	VARCHAR2(1)	X		X
context	VARCHAR2(30)			X

Parameter	Datatype	Required	Derived	Optional
created_by	NUMBER		X	
creation_date	DATE		X	
discount_id	NUMBER		X	
discount_line_id	NUMBER		X	
header_id	NUMBER	X		
last_updated_by	NUMBER		X	
last_update_date	DATE		X	
last_update_logi n	NUMBER		X	
line_id				
percent	NUMBER		X	
price_adjustmen t_id	NUMBER	C -for update and delete		X
program_applica tion_id	NUMBER			X
program_id	NUMBER			X
program_update _date	DATE			X
request_id	NUMBER			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		

Parameter	Datatype	Required	Derived	Optional
line_index	NUMBER			X
orig_sys_discount_ref	VARCHAR2(50)			X
change_request_code	VARCHAR2(30)			X
status_flag	VARCHAR2(1)			X
list_header_id	NUMBER	C - not required only for tax records		
list_line_id	NUMBER	C - not required only for tax records		
list_line_type_code	VARCHAR2(30)		X	
modifier_mechanism_type_code	VARCHAR2(30)		X	
modified_from	NUMBER		X	
modified_to	NUMBER		X	
updated_flag	VARCHAR2(1)			
update_allowed	VARCHAR2(1)		X	
applied_flag	VARCHAR2(1)		X	
change_reason_code	VARCHAR2(30)			X
change_reason_text	VARCHAR2(2000)			X
operand	NUMBER			X

Parameter	Datatype	Required	Derived	Optional
operand_per_qty	NUMBER			
arithmetic_operator	VARCHAR2(30)		X	
cost_id	NUMBER		X	
tax_code	VARCHAR2(30)			
tax_exempt_flag	VARCHAR2(1)			
tax_exempt_number	VARCHAR2(80)			
tax_exempt_reason_code	VARCHAR2(30)			
parent_adjustment_id	NUMBER	X -for update and delete		
invoiced_flag	VARCHAR2(1)		X	
estimated_flag	VARCHAR2(1)		X	
inc_in_sales_performance	VARCHAR2(1)		X	
split_action_code	VARCHAR2(30)			X
adjusted_amount	NUMBER		X	
adjusted_amount_per_qty	NUMBER			
pricing_phase_id	NUMBER		X	
charge_type_code	VARCHAR2(30)		X	

Parameter	Datatype	Required	Derived	Optional
charge_subtype_code	VARCHAR2(30)		X	
list_line_no	VARCHAR2(240)		X	
source_system_code	VARCHAR2(30)		X	
benefit_qty	NUMBER		X	
benefit_uom_code	VARCHAR2(3)		X	
print_on_invoice_flag	VARCHAR2(1)		X	
expiration_date	DATE		X	
rebate_transaction_type_code	VARCHAR2(30)		X	
rebate_transaction_reference	VARCHAR2(80)		X	
rebate_payment_system_code	VARCHAR2(30)		X	
redeemed_date	DATE		X	
redeemed_flag	VARCHAR2(1)		X	
accrual_flag	VARCHAR2(1)		X	
range_break_quantity	NUMBER		X	
accrual_conversion_rate	NUMBER		X	

Parameter	Datatype	Required	Derived	Optional
pricing_group_sequence	NUMBER		X	
modifier_level_code	VARCHAR2(30)		X	
price_break_type_code	VARCHAR2(30)		X	
substitution_attribute	VARCHAR2(30)		X	
proration_type_code	VARCHAR2(30)		X	
credit_or_charge_flag	VARCHAR2(1)		X	
include_on_returns_flag	VARCHAR2(1)		X	
ac_attribute1...15	VARCHAR2(240)			X
ac_context	VARCHAR2(150)			X
retrobill_request_id	NUMBER			
tax_rate_id	NUMBER			

TYPE Line_Adj_Tbl_Type IS TABLE OF Line_Adj_Rec_Type

INDEX BY BINARY_INTEGER;

TYPE Line_Adj_Val_Rec_Type IS RECORD

(discount List_name)

TYPE Line_Adj_Val_Tbl_Type IS TABLE OF Line_Adj_Val_Rec_Type

INDEX BY BINARY_INTEGER;

Line_Price_Att_Rec_Type

For column descriptions, please refer to the Order Management e-TRM for the table OE_ORDER_PRICE_ATTRIBS

OE_ORDER_PRICE_ATTRIBS

Parameter	Datatype	Required	Derived	Optional
order_price_attri b_id	NUMBER			X
header_id	NUMBER			
line_id	NUMBER			
line_index	NUMBER			
creation_date	DATE		X	
created_by	NUMBER		X	
last_update_date	DATE		X	
last_updated_by	NUMBER		X	
last_update_logi n	NUMBER		X	
program_applica tion_id	NUMBER			X
program_id	NUMBER			X
program_update _date	DATE			X
request_id	NUMBER			X
flex_title	VARCHAR2(60)			
pricing_context	VARCHAR2(30)			

Parameter	Datatype	Required	Derived	Optional
pricing_attribute 1...	VARCHAR2(240)			
pricing_attribute 100				
context	VARCHAR2(30)			
attribute1..15	VARCHAR2(240)			
Override_Flag	VARCHAR2(1)			
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	X			

TYPE Line_Price_Att_Tbl_Type is TABLE of Line_Price_Att_rec_Type
INDEX by BINARY_INTEGER

Line_Adj_Att_Rec_Type

For column descriptions, please refer to the Order Management e-TRM for the table
OE_PRICE_ADJ_ATTRIBS

OE_PRICE_ADJ_ATTRIBS

Parameter	Datatype	Required	Derived	Optional
price_adj_attrib_ id	NUMBER	C -for update and delete		
price_adjustmen t_id	NUMBER	C - if adj_index not passed		
Adj_index	NUMBER	C - if price_adjustmen t_id not passed		

Parameter	Datatype	Required	Derived	Optional
flex_title	VARCHAR2(60)	X		
pricing_context	VARCHAR2(30)	X		
pricing_attribute	VARCHAR2(30)	X		
creation_date	DATE		X	
created_by	NUMBER		X	
last_update_date	DATE		X	
last_updated_by	NUMBER		X	
last_update_logi n	NUMBER		X	
program_applica tion_id	NUMBER			X
program_id	NUMBER			X
program_update _date	DATE			X
request_id	NUMBER			X
pricing_attr_valu e_from	VARCHAR2(240)	X		
pricing_attr_valu e_to	VARCHAR2(240)			X
comparison_ope rator	VARCHAR2(30)			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		

TYPE Line_Adj_Att_Tbl_Type is TABLE of Line_Adj_Att_rec_Type
INDEX by BINARY_INTEGER

Line_Adj_Assoc_Rec_Type

For column descriptions, please refer to the Order Management e-TRM for the table
OE_PRICE_ADJ ASSOCS

OE_PRICE_ADJ ASSOCS

Parameter	Datatype	Required	Derived	Optional
price_adj_assoc_ id	NUMBER	C - for update and delete		
line_id	NUMBER	C - if line_index not passed		
Line_Index	NUMBER	C - if line_id not passed		
price_adjustmen t_id	NUMBER	C - if adj_index not passed		
Adj_index	NUMBER	C - if price_adjustmen t_id not passed		
rltd_Price_Adj_I d	NUMBER			X
Rltd_Adj_Index	NUMBER			X
creation_date	DATE		X	
created_by	NUMBER		X	
last_update_date	DATE		X	
last_updated_by	NUMBER		X	
last_update_logi n	NUMBER		X	

Parameter	Datatype	Required	Derived	Optional
program_application_id	NUMBER			X
program_id	NUMBER			X
program_update_date	DATE			X
request_id	NUMBER			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		

TYPE Line_Adj_Assoc_Tbl_Type is TABLE of Line_Adj_Assoc_rec_Type

INDEX by BINARY_INTEGER;

TYPE Line_Scredit_Rec_Type IS RECORD

OE_SALES_CREDITS

Parameter	Datatype	Required	Derived	Optional
attribute1.....attribute15	VARCHAR2(240)			X
context	VARCHAR2(30)			X
created_by	NUMBER		X	
creation_date	DATE		X	
dw_update_advice_flag	VARCHAR2(1)			X
header_id	NUMBER			X

Parameter	Datatype	Required	Derived	Optional
last_updated_by	NUMBER		X	
last_update_date	DATE		X	
last_update_logi n	NUMBER		X	
line_id	NUMBER			X
percent	NUMBER	X		
salesrep_id	NUMBER	X		
sales_credit_type _id	NUMBER	X		
sales_credit_id	NUMBER			X
wh_update_date	DATE			X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		
line_index	NUMBER			X
orig_sys_credit_r ef	VARCHAR2(50)			X
change_request_ code	VARCHAR2(30)		X	
status_flag	VARCHAR2(1)		X	

TYPE Line_Scredit_Tbl_Type IS TABLE OF Line_Scredit_Rec_Type

INDEX BY BINARY_INTEGER;

-- Line_Scredit value record type

```
TYPE Line_Scredit_Val_Rec_Type IS RECORD
```

```
(   salesrep

, sales_credit_type

);
```

```
TYPE Line_Scredit_Val_Tbl_Type IS TABLE OF Line_Scredit_Val_Rec_Type
```

```
INDEX BY BINARY_INTEGER;
```

Lot_Serial_Rec_Type

OE_LOT_SERIAL_NUMBERS

Parameter	Datatype	Required	Derived	Optional
attribute1....attribute15	VARCHAR2(240)			X
context	VARCHAR2(30)			X
created_by	NUMBER		X	
creation_date	DATE		X	
from_serial_number	VARCHAR2(30)	C - if lot number not passed		X
last_updated_by	NUMBER		X	
last_update_date	DATE		X	
last_update_logi n	NUMBER		X	
line_id	NUMBER	C - if line index not passed		X
lot_number	VARCHAR2(30)	C - if lot serial information not available		X

Parameter	Datatype	Required	Derived	Optional
lot_serial_id	NUMBER	C - for update and delete operation		X
quantity	NUMBER	X		X
to_serial_number	VARCHAR2(30)	C - if lot number not passed		X
return_status	VARCHAR2(1)		X	
db_flag	VARCHAR2(1)		X	
operation	VARCHAR2(30)	X		
line_index	NUMBER	C - if line id not passed		
orig_sys_lotserial_ref	VARCHAR2(50)			X
change_request_code	VARCHAR2(30)		X	
status_flag	VARCHAR2(1)		X	
line_set_id	NUMBER		X	

TYPE Lot_Serial_Tbl_Type IS TABLE OF Lot_Serial_Rec_Type

INDEX BY BINARY_INTEGER;

TYPE Lot_Serial_Val_Rec_Type IS RECORD

(line lot_serial);

TYPE Lot_Serial_Val_Tbl_Type IS TABLE OF Lot_Serial_Val_Rec_Type

INDEX BY BINARY_INTEGER;

Process Order Usage

Process Order is a complex API and a number of operations on a single order object and its entities can be performed via this API. This section is intended to help a user of the process order API in identifying the required parameters for some common operations

and to give an understanding of the business flow behind each of these operations.

CREATE operation

For creating new order entities, the entity records and entity tables passed to Process_Order should have the operation OE_GLOBALS.G_OPR_CREATE. Pass in the known attributes on these records and Process Order will default the other *missing* attributes.

Note: If you pass in NULL values for some attributes, process order will interpret it as insertion of NULL into those attributes column and will NOT get default values for those attributes.

Line adjustments, line sales credits and line lot serial numbers belong to an order line and the ID of this parent line is stored on these records. When inserting the parent line and the child entities of this line together, the ID of the parent line is unknown. In this case, the line_index field on the child entity records indicates the index for the parent entity record in the lines table. After the line is processed, the line ID is retrieved and based on the line_index, the correct line_ID is populated on the child entity record.

Similarly for config items, the model line's ID is stored on the option lines. Again, when inserting models and options in a configuration together, the line ID for the model line will not be available. The top_model_line_index field on the option lines can be used to indicate the index for this model line in the lines table. Since models and standard lines are processed before the options, this line ID can then be populated before the option lines are processed.

The following steps are executed for each entity within process order:

- Check security
- Attribute level validation
- Entity level validation
- Posted to the database
- Cross Record Logic

Create Operation Example 1

Creating a new order with 2 lines and 1 line adjustment and the adjustment belongs to the second line.

```
-- SETTING UP THE HEADER RECORD

-- Initialize record to missing
mo_global.init('ONT');
l_header_rec := OE_ORDER_PUB.G_MISS_HEADER_REC;
```

```

-- Required attributes (e.g. Order Type and Customer)

l_header_rec.order_type_id := 1000;

l_header_rec.sold_to_org_id := 100;

-- Other attributes

l_header_rec.price_list_id := 10;

.....

-- Null attribute: no defaulting for freight terms

l_header_rec.freight_term_code = NULL;

-- Indicates to process order that a new header is being created

l_header_rec.operation := OE_GLOBALS.G_OPR_CREATE;

-- FIRST LINE RECORD

-- Initialize record to missing

l_line_tbl(1) := OE_ORDER_PUB.G_MISS_LINE_REC;

-- Line attributes

l_line_tbl(1).inventory_item_id := 311;

l_line_tbl(1).ordered_quantity := 1;

l_line_tbl(1).operation := OE_GLOBALS.G_OPR_CREATE;

-- SECOND LINE RECORD

-- Initialize record to missing

l_line_tbl(2) := OE_ORDER_PUB.G_MISS_LINE_REC;

-- Line attributes

l_line_tbl(2).inventory_item_id := 312;

l_line_tbl(2).ordered_quantity := 2;

l_line_tbl(2).operation := OE_GLOBALS.G_OPR_CREATE;

```

```

-- LINE ADJUSTMENT RECORD

-- Initialize record to missing

l_line_adj_tbl(1) := OE_ORDER_PUB.G_MISS_LINE_ADJ_REC;

-- Attributes for the line adjustment

l_line_adj_tbl(1).discount_id := 1;

l_line_adj_tbl(1).percent := 5;

l_line_adj_tbl(1).operation := OE_GLOBALS.G_OPR_CREATE;

-- Indicator that this adjustment belongs to the second line

l_line_adj_tbl(1).line_index := 2;

-- CALL TO PROCESS ORDER

OE_Order_PUB.Process_Order(

.....

.....

-- Passing just the entity records that are a part of this order

p_header_rec => l_header_rec

p_line_tbl=> l_line_tbl

p_line_adj_tbl=> l_line_adj_tbl

-- OUT variables

x_header_rec=> l_header_rec

x_header_scredit_tbl=> l_header_scr_tbl

x_header_adj_tbl=> l_header_adj_tbl

x_line_tbl=> l_line_tbl

x_line_scredit_tbl=> l_line_scredit_tbl

x_line_adj_tbl=> l_line_adj_tbl

```

```

.....

x_return_status=> l_return_status

x_msg_count=> l_msg_count

x_msg_data=> l_msg_data);

-- Retrieve messages

if l_msg_count > 0 then

for l_index in 1..l_msg_count loop

l_msg_data := oe_msg_pub.get(p_msg_index => l_index, p_encoded => 'F');

end loop;

end if;

-- Check the return status

if x_return_status = FND_API.G_RET_STS_SUCCESS then

success;

else

failure;

end if;

```

Create Operation Example 2

Inserting a new line into an existing order.

Note: You cannot insert new lines (or, process any other entity) belonging to different orders in one process order call

```

mo_global.init('ONT');
-- NEW LINE RECORD

l_line_tbl(1) := OE_ORDER_PUB.G_MISS_LINE_REC;

-- Primary key of the parent entity (If not passed, this
will be retrieved from the parent record, p_header_rec, if
it was also passed to process order else there will be an
error in the processing of this line).

```

```

l_line_tbl(1).header_id := 1000;

-- Attributes for the new line

l_line_tbl(1).inventory_item_id := 311;

l_line_tbl(1).ordered_quantity := 1;

-- Indicator that a new line is being created

l_line_tbl(1).operation := OE_GLOBALS.G_OPR_CREATE;

-- CALL TO PROCESS ORDER

OE_ORDER_PUB.Process_Order(.....

-- Passing just the entity records that are being created

p_line_tbl=> l_line_tbl

-- OUT variables

.....);

```

UPDATE operation

Update of any attributes on the entities of the order object should always go through the process order API. For updates, the operations on the entity record being updated should be OE_GLOBALS.G_OPR_UPDATE. Only the attributes that are being updated, the primary key of this entity and the primary key of the parent entity, if any, need to be passed and others can be set to missing. Process Order will query these missing attributes from the database.

Note: To update an existing value to NULL, you can pass NULL for that attribute on the record parameter passed to process order API. There are some attributes, however, that are internally set by the system (e.g. status columns like booked_flag, flow_status_code) and should not be updated by the user.

Process Order can process entities belonging to one order object in one call. In a call where one entity record is being updated, another entity record can also be created as long as they belong to the same order object. For e.g. a line can be updated (G_OPR_UPDATE) and another line inserted (G_OPR_CREATE) in the same process order call if both lines have the same header_id.

The following processing steps are executed for each entity record that is updated:

- Check security for update of the changed attributes
- Validation of updated attributes
- Clearing of (or set to missing) attributes dependent on the updated attributes
- Defaulting of missing attributes
- Entity level validation
- Entity security check
- Posted to the database
- Cross Record Logic

Update Operation Example

Updating the bill to organization and order quantity on an order line.

```
-- LINE RECORD WITH THE CHANGES

-- Changed attributes

l_line_tbl(1) := OE_ORDER_PUB.G_MISS_LINE_REC.

l_line_tbl(1).invoice_to_org_id := 322;

l_line_tbl(1).ordered_quantity := 2;

-- Primary key of the entity i.e. the order line

l_line_tbl(1).line_id := 1000;

-- Indicates to process order that this is an update operation

l_line_tbl(1).operation := OE_GLOBALS.G_OPR_UPDATE;

-- CALL TO PROCESS ORDER

OE_ORDER_PUB.Process_Order(.....

-- Entity records

p_line_tbl=> l_line_tbl

-- OUT variables

.....);
```

DELETE operation

For deletes, the operations on the entity records being deleted should be OE_GLOBALS.G_OPR_DELETE. Only the primary key of the entity being deleted needs to be passed.

Deletes are cascaded down to the child entities for e.g. to delete an order, only the header record with the header ID needs to be passed to process order and all the child entities i.e. lines, header sales credits, header adjustments are deleted. Deleting the lines results in the deletion of line adjustments, line sales credits and line lot serial numbers as well. Also, it would delete delayed requests logged by or against this entity and its child entities.

Deletes would also result in the deletion of any holds and attachments associated with the deleted entity and its child entities. The workflow status information for this entity is also purged.

The following processing steps are executed for each entity record that is deleted:

- Entity security check.

Note: Child entities will always be deleted if parent entity passes security check. There will not be a separate security check for child entities

- Entity level validation for deletes.
- Delete posted to the database. This will also delete child entities, holds and attachments.
- Cross Record Logic

Delete Operation Example

-- Only the record for the parent entity needs to be passed. Since the whole order is being deleted, only the header record needs to be passed.

-- Primary key of this order i.e. the header

```
mo_global.init('ONT');
```

```
l_header_rec := OE_ORDER_PUB.G_MISS_LINE_REC;
```

-- Indicate to process order that the order is to be deleted

```
l_header_rec.operation := OE_GLOBALS.G_OPR_DELETE;
```

-- CALL TO PROCESS ORDER

```
OE_ORDER_PUB.Process_Order(.....
```

```
-- Entity records

p_header_rec=> l_header_rec

-- OUT variables

.....);
```

Process Order and Action Requests

Process order API can also process some other actions on the order object. For e.g. booking the order, applying/releasing holds.

For each action request, a request record should be passed in the action request table (p_action_request_tbl) to process order and the parameters of the request record can provide additional information needed to carry out that action. The validation_level parameter determines whether validation of the parameters of the request record needs to be performed.

If the ID of the order or line for which this request needs to get executed is not known as it is being created in the same call, the ENTITY_INDEX field on the request record indicates the index for the entity record in the entity table passed to

Following are the action requests currently supported and the parameters on the request record that are used in processing the request.

Book the order

This request is used to book the order.

Book Order

Parameter	Description
request_type	OE_GLOBALS.G_BOOK_ORDER
entity_code	This should be always set to OE_GLOBALS.G_ENTITY_HEADER as booking is an order level action.
entity_id	Header ID of the order to be booked. If the order is also being created in the same call to process order, then the user does not need to provide this value.
p_action_request_tbl	IN parameter
x_action_request_tbl	OUT parameter

Apply Automatic Attachments

The attachment rules will be evaluated for the entity and if applicable, documents will be attached to the entity.

Apply Automatic Attachments

Parameter	Description
request_type	OE_GLOBALS.G_APPLY_AUTOMATIC_AT CHMT
entity_code	OE_GLOBALS.G_ENTITY_HEADER if the rules are to be evaluated for the order header OE_GLOBALS.G_ENTITY_LINE if rules are to be evaluated for the order line.
entity_id	Header ID if the entity is G_ENTITY_HEADER, Line ID if the entity is G_ENTITY_LINE
entity_index	If entity is G_ENTITY_LINE and the line is being created in the same call to process order, then send in the index position for this line in the table, p_line_tbl.

Apply Hold

Holds will be applied based on the parameters sent in on the request record.

Apply Hold

Parameter	Description
request_type	OE_GLOBALS.G_APPLY_HOLD
entity_code	OE_GLOBALS.G_ENTITY_HEADER for order or OE_GLOBALS.G_ENTITY_LINE for line.
entity_id	ID of the order or line to be held

Parameter	Description
param1	Hold ID to identify the type of hold that should be applied. (HOLD_ID from OE_HOLD_DEFINITIONS)
param2	Hold entity code for the hold source to be created. C: Customer hold source S: Bill To or Ship To hold source I: Item hold source O: Order hold source W: Warehouse Hold Source
param3	Hold entity ID C, B, or S: for Org ID O: Header ID I: Inventory Item ID
param4	Hold comment
date_param1	Hold Until Date
parm6-param20	Attribute1-15 of the descriptive flexfield associated with the hold source record.

Release Hold

Releases the holds associated with an order or line.

Release Hold

Parameter	Description
request_type	OE_GLOBALS.G_RELEASE_HOLD
entity_code	OE_GLOBALS.G_ENTITY_HEADER for order or OE_GLOBALS.G_ENTITY_LINE for line.

Parameter	Description
entity_id	ID of the order or line to be released from hold
param1	Hold ID to specify the type of hold that is to be removed.
param2	Hold entity code on the hold source associated with the hold to be released C: Customer hold source S: Bill To or Ship To hold source I: Item hold source O: Order hold source W: Warehouse Hold Source
param3	Hold entity ID: C or S: for Org ID O: Header ID I: Inventory Item ID
param4	Release Reason Code
param5	Release Comment

Delink Config

This request deletes the configuration item.

Delink Config

Parameter	Description
request_type	OE_GLOBALS.G_DELINK_CONFIG
entity_code	OE_GLOBALS.G_ENTITY_LINE
entity_id	Line ID of the top model line of the ATO configuration

Match and Reserve

This request checks if an existing configuration item matches the configuration created by the user and if configuration item is available, it reserves it.

Match and Reserve

Parameter	Description
request_type	OE_GLOBALS.G_MATCH_AND_RESERVE
entity_code	OE_GLOBALS.G_ENTITY_LINE
entity_id	Line ID of the top model line of the ATO configuration

Get Ship Method

The actions can be performed only if the system parameter Enable Ship Method is enabled. Oracle Order Management will use the routing guides defined in product transportation to get the most appropriate carrier and ship method, dependent on defined business rules. The action always works on the entire order at a time. The action ignores following type of lines while deriving the ship method,

1. Dropship lines
2. Return lines
3. Service lines
4. CONFIG lines (The line created after progressing ATO model once it is booked and scheduled)
5. Lines that are already shipped
6. Lines that are already fulfilled
7. Lines that are already closed
8. Lines which do not have all required parameters. e.g. ship from, ship to.

Get Ship Method

Parameter	Description
-----------	-------------

request_type	OE_GLOBALS.G_GET_SHIP_METHOD
entity_code	OE_GLOBALS.G_ENTITY_HEADER or OE_GLOBALS.G_ENTITY_HEADER
entity_id	Header ID if entity is LINE Line ID if entity is LINE

Add/Remove from Fulfillment Set

Add or remove lines from fulfillment sets. The Parameter Description is as follows:

```
request_type OE_GLOBALS.G_ADD_FULFILLMENT_SET or
OE_GLOBALS.G_REMOVE_FULFILLMENT_SET
entity_code OE_GLOBALS.G_ENTITY_LINE for line.
entity_id Line ID of the line
param5 Fulfillment Set Name
```

Action Request Example

Process order can also process other action requests on the order or line as has been mentioned previously.

Apply a hold to an order line due to a defective item.

```
-- ACTION REQUEST RECORD

-- Indicates that it is a line level action

l_request_rec.entity := OE_GLOBALS.G_ENTITY_LINE;

-- primary key of the line on which the hold is to be applied

l_request_rec.entity_id := 100;

-- name of the action request

l_request_rec.request_name := OE_GLOBALS.G_APPLY_HOLD;

-- request record parameters

-- defective product hold (hold_id)

l_request_rec.param1 := 4;

-- indicator that it is an item hold (hold_entity_code)

l_request_rec.param2 = 'I';
```



```
-- Id of the item (hold_entity_id)

l_request_rec.param3 := 3214;

-- inserting request record into the action request table

l_action_request_tbl := l_request_rec;
```

See:

Oracle Applications Message Reference Manual

Oracle Order Management User's Guide

Oracle Order Management: Integration with Other Products

Integrating Oracle Order Management with Oracle Receivables and Invoicing

Oracle Order Management provides functionality to integrate with Oracle Receivables. Using AutoInvoice, you can create invoices, create credit memos and credits on account, recognize revenue, and manage sales credits.

Basic Needs

Oracle Order Management and Oracle Receivables provide features you need to satisfy the following integration needs:

- Create accurate and timely invoices, credit memos, and credits on account from Order Management transactions
- Control when order transactions are invoiced

Major Features

Invoicing Activity

Order Management provides a program that automatically collects order and return information and populates the Oracle Receivables AutoInvoice interface tables. Using order management transaction types, profile options and system parameters, you control certain accounting aspects of the information being transferred to Oracle Receivables. Invoicing module is modeled using workflow and can happen after shipping for shippable flows and any time after booking for non-shippable flows. Invoicing can be placed at both order line level workflows as well as order header level workflows. If placed at the order header level, all lines in an order are populated in

AutoInvoice interface tables at the same time.

Invoice Source

You must setup invoice sources in Oracle Receivables prior to using Oracle Order Management. Order Management uses the invoice source setup at the transaction type level or the system parameter Invoice Source when populating order line information into AutoInvoice interface tables. When defining invoice sources in Oracle Receivables, you must create at least one invoice source for Order Management's use if you want to interface orders and returns for processing by AutoInvoice.

The following table shows the necessary field values for the Transaction Source window in Oracle Receivables.

Transaction Sources Window: Required Settings

Field in Transaction Sources Window	Necessary Value
Batch Source region	
Type	Imported
Status	Active
Automatic Transaction Numbering	Yes if OM: Invoice Numbering Method is set to Automatic, No if set to Delivery Name
Customer Information region	Customer Information region
Sold-to Customer	Id
Bill-to Customer	Id
Bill-to Address	Id
Bill-to Contact	Id
Ship-to Customer	Id
Ship-to Address	Id
Ship-to Contact	Id
Payment Method Rule	(any)

Field in Transaction Sources Window	Necessary Value
Customer Bank Account	(any)
Accounting Information region	Accounting Information region
Invoicing Rule	Id
Accounting Rule	Id
Accounting Flexfield	(any)
Derive Date	(any)
Payment Terms	Id
Revenue Account allocation	(any)
Other Information region	Other Information region
Transaction Type	Id
Memo Reason	Id
Agreement	Id
Memo Line Rule	Id
Sales Territory	(any)
Inventory Item	Id
Unit of Measure	Id
FOB Point	Code
Freight Carrier	Code
Related Document	Id
Sales Credits Validation region	Sales Credits Validation region

Field in Transaction Sources Window	Necessary Value
Salesperson	Id
Sales Credit Type	Id
Sales Credit	Percent

Note: In the Batch Source region, you must define at least one transaction source with automatic invoice numbering, regardless of your setting for the *OM: Invoice Numbering Method* profile.

Automatic Tax Calculation

As orders from Order Management are processed, AutoInvoice automatically calculates sales tax based on the Sales Tax Location flexfield combination. If you have designated a customer as tax-exempt, AutoInvoice will not tax any items billed for the customer. Similarly, if you have designated an item as non-taxable, AutoInvoice will not tax the item.

Automatic Account Code Creation

Oracle Receivables uses AutoAccounting to determine the revenue account for all transactions from Order Management. AutoAccounting lets you define what information is used to define the various segments of your Accounting Flexfield.

Accounting and Invoicing Rules

Order Management uses accounting and invoicing rules. This information is transferred to Oracle Receivables and used to determine the invoice date (invoicing rule) and general ledger distribution records (accounting rule). Order Management passes an invoicing rule and accounting rule for each order transaction interfaced to Oracle Receivables, except for when the accounting rule is Immediate, in which case Order Management does not pass any value (inserts null).

Credit Method for Accounting Rule

Order Management transfers a Credit Method for Accounting Rule for each return line. This credit method is recognized only by invoices that use duration accounting rules. You can assign a Credit Method for Accounting Rule to the order type of the return. If the Credit Method for Accounting Rule field for the order type is null, then Order Management transfers LIFO (Last In First Out).

Credit Method for Installments

Order Management transfers a Credit Method for Installments for each return line. This credit method is used for crediting an invoice that uses split payment terms. You can assign a Credit Method for Installments to the order type of the return. If the Credit Method for Installments field for the order type is null, then Order Management transfers **LIFO** (Last In First Out).

Internal Sales Orders

The Invoicing Activity does not process internal sales order lines, even if it is placed in the internal sales order's workflow. However, freight charges associated with the internal order lines are interfaced to Receivables.

Internal sales orders are orders that originate in Oracle Purchasing as internal requisitions, and are imported to Order Management as internal sales orders using Order Import.

See:

Using AutoAccounting, *Oracle Receivables User's Guide*

Accounting Rules, *Oracle Receivables User's Guide*

Entering Commitments, *Oracle Receivables User's Guide*

Defining Items, *Oracle Inventory User's Guide*

Accounting for Credit Memos, *Oracle Receivables User's Guide*

Invoicing of ATO Configurations

Invoicing Attributes

For ATO configurations, Order Management considers the base model's item attribute of a configuration to see if it should consider passing invoice information to Oracle Receivables for each order line in the configuration. If you have the item attributes **Invoiceable Item** and **Invoice Enabled** set to Yes for the base model item, Order Management then considers these item attributes for each component in the bill of material for the model to see if they should be invoiced in Oracle Receivables. If the item attributes **Invoiceable Item** or **Invoice Enabled** are set to No for the base model item, Order Management does not pass invoicing information to Oracle Receivables for any order lines for the components within the configuration, regardless of the item attribute settings.

Required for Revenue Attribute

The bill of material attribute Required for Revenue allows you to define specific items in a bill that must be shipped before their parent can be invoiced. In all cases the control applies to only one level, the immediate parent. Except for classes, the control relationship is the child affecting the parent.

For example: Included Item A has the Required for Revenue attribute set to Yes. Option A is not eligible to interface to Oracle Receivables until Included Item A is shipped, even if Option A is also shippable and has shipped. All other components, including Model A and Model 1, are eligible to interface regardless of Included Item A's shipment status.

Example #2: Included Item B has the Required for Revenue attribute set to Yes. Model 1 is not eligible to interface to Oracle Receivables until Included Item B is shipped. And again, Option A is not eligible to interface until Included Item A is shipped.

Example #3: If any item below a class in a bill has the Required for Revenue attribute set

to Yes, then that item must be shipped before the parent item and the other items in the class are eligible to interface. For example, in the figure above, Included Item C has the Required for Revenue attribute set to Yes. Therefore, both Option C and Class C are not eligible to interface until Included Item C is shipped.

See:

Item Attributes Used by Order Management, *Oracle Order Management User's Guide*

Overview of Bills of Material, *Oracle Bills of Material User's Guide*

Understanding the Receivables Interface Tables

Oracle Order Management inserts information into two of the three AutoInvoice interface tables (RA_INTERFACE_LINES and RA_INTERFACE_SALES_CREDITS). RA_INTERFACE_DISTRIBUTIONS is not described in this essay because all account code creation is done by AutoInvoice based on the AutoAccounting rules you have defined. For customer acceptance, AR_INTERFACE_CONTS_ALL is populated for all lines that have contingencies attached to them. This information is interfaced to Receivables for lines and discount lines. This information is not passed for freight charges. If the line has pre-billing customer acceptance and it is rejected then Oracle Order Management does pass the invoicing information for this line to Receivables. Associated freight charges also will not be interfaced to Receivables in this case.

The following describes what information Order Management interfaces for each order and order line, each sales credit, and each freight charge.

Note: Order Management line numbers are populated in the following manner within RA_INTERFACE_LINES.:

INTERFACE_LINE_ATTRIBUTE6 (line_id)

INTERFACE_LINE_ATTRIBUTE12 (shipment_number)

INTERFACE_LINE_ATTRIBUTE13 (option_number)

INTERFACE_LINE_ATTRIBUTE14 (service_number)

RA_INTERFACE_LINES

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_LINE_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	None

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_LINE_CONTEXT	NA	Varchar2(30)	Order Management inserts your value for the OM: Source Code profile option	Profile option OM: Source Code
INTERFACE_LINE_ATTRIBUTES1	NA	Varchar2(30)	Order Management inserts OE_HEADERS.ORDER_NUMBER.	header.order_number
INTERFACE_LINE_ATTRIBUTES2	NA	Varchar2(30)	Order Management inserts OE_ORDER_TYPES_V.NAME in the base language.	type.name
INTERFACE_LINE_ATTRIBUTES3	NA	Varchar2(30)	<p><i>For a Shipped order line and for Freight Charges:</i> Order Management inserts substr (wsh_new_deliveries.name1..30</p> <p><i>For a Non-shipped order line or a Return Line:</i> Order Management inserts 0 (zero).</p>	<p>The available values are:</p> <ul style="list-style-type: none"> • Shipped order line: del.name • Non-Shipped order line.¹ • Return lines: 0 (zero) • Freight Charges: delivery name • Discount lines: 0 (zero)

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_LINE_AT TRIBUTE4	NA	Varchar2(30)	<p><i>For a Shipped order line and for Freight Charges:</i> Order Management inserts: substr(wsh_new_deliveries.waybill1..30)</p> <p><i>For a Non-shipped order line or a Return Line:</i> Order Management inserts 0 (zero).</p>	<p>Shipped order line: substr(del.waybill,1,30)</p> <p>Non-Shipped order line: 0 (zero)</p> <p>Return lines: 0 (zero)</p> <p>Freight Charges: substr(del.waybill,1,30)</p> <p>Discount lines: 0 (zero)</p>
INTERFACE_LINE_AT TRIBUTE5	NA	Varchar2(30)	<p><i>For a Sales order or return line:</i> Order Management inserts the number of times the order or return line has been interfaced for invoice or credit.</p> <p><i>For Freight charges:</i> Order Management inserts a value of 1 (one).</p>	<p>Sales Order lines and Return lines: the number of times the order or return line has been interfaced for invoice or credit.</p> <p>Freight Charges: 1 (one)</p>
INTERFACE_LINE_AT TRIBUTE6	NA	Varchar2(30)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.LINE_ID.</p> <p><i>For Freight charges:</i> OE_CHARGE_LINES_V.CHARGE_ID.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales Order lines and Return lines: line.line_id Discount lines: line.line_id Freight Charges: charge.charge_id

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_LINE_AT TRIBUTE7	NA	Varchar2(3 0)	<i>For a shipped order line, a return line, or for Freight Charges: Order Management inserts zero (0).</i>	<p>The available values are:</p> <ul style="list-style-type: none"> Shipped order line: 0 Return lines: 0 (zero) Freight Charges: 0 (zero)
INTERFACE_LINE_AT TRIBUTE8	NA	Varchar2(3 0)	<p><i>For a shipped order line: Bill of Lading</i></p> <p><i>For a non-shipped order line, a return line, or for Freight Charges: Order Management inserts zero (0).</i></p>	<p>The available values are:</p> <ul style="list-style-type: none"> Shipped order line: wsh_document_instances.sequence_number Non-Shipped order line: 0 (zero) Return lines: 0 (zero) Freight Charges: 0 Discount lines: 0 (zero)
INTERFACE_LINE_AT TRIBUTE9	NA	Varchar2(3 0)	Order Management inserts the customer item number, if one is defined. Otherwise, it inserts 0 (zero).	Insert customer item number if one is defined, otherwise insert 0 (zero).

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_LINE_AT TRIBUTE10	NA	Varchar2(30)	Order Management inserts OE_ORDER_LINES.S HIP_FROM_ORD_ID.	NA
INTERFACE_LINE_AT TRIBUTE11	NA	Varchar2(30)	Order Management inserts oe_price_adjustments. price_adjustment_id for discount lines: Note: System Parameter Show Discount Details on Invoice must be Yes. For all other lines the value 0 is passed.	Discount lines: pa.price_adjustment_id
INTERFACE_LINE_AT TRIBUTE12	NA	Varchar2(30)	Order Management inserts oe_order_lines_all.shi pment_number	The available values are: <ul style="list-style-type: none"> Sales Order lines, Return lines, and Discount lines: line.shipment_number Freight Charges: None
INTERFACE_LINE_AT TRIBUTE13	NA	Varchar2(30)	Order Management inserts oe_order_lines_all.opt ion_number	The available values are: <ul style="list-style-type: none"> Sales Order lines, Return lines, and Discount lines: line.option_number Freight Charges: None

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_LINE_AT TRIBUTE14	NA	Varchar2(30)	Order Management inserts oe_order_lines_all.service_number	<p>The available values are:</p> <ul style="list-style-type: none"> Sales Order lines, Return lines, and Discount line: line.service_number Freight Charges: None
INTERFACE_LINE_AT TRIBUTE15	NA	Varchar2(30)	Order Management does not insert a value into this column	Null

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
BATCH_SOURCE_NAME	Not Null	Varchar2(50)	<p>Order Management enters the invoice source name using the following rules:</p> <p>If the profile option <i>OM: Invoice Numbering Method</i> is set to Delivery Name, but line is non-shippable or a return line, Order Management passes the value of the system parameter <i>OM: Non-Delivery Invoice Source</i>.</p> <p>For all other lines, Order Management uses the following sequence to determine the invoice source:</p> <ol style="list-style-type: none"> 1. Order Transaction Type associated with the order line being interfaced to AutoInvoice. 2. Order Line Transaction Type associated with the order line being interfaced to AutoInvoice. 3. Value of the system parameter Invoice Source. 	<p>Invoice Source from the hierarchy:</p> <ul style="list-style-type: none"> • Line type • Order type • System Parameter Invoice Source

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
SET_OF_BOOKS_ID	Not Null	NUMBER (15)	Order Management inserts the ID from the OE: Set of Books system parameter for the operating unit of the order line.	OE_Sys_Parameters.v alue (SET_OF_BOOKS_ID)
LINE_TYPE	Not Null	Varchar2(20)	<p><i>For a Sales order or return line:</i> Order Management inserts LINE.</p> <p><i>For Freight charges:</i> Order Management inserts FREIGHT for shipment freight charges if the system parameter <i>Invoice Freight as Revenue</i> is set to No, otherwise the system inserts LINE.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Line: For sales order lines, return lines and discount lines: Freight Charges: Can be either <ul style="list-style-type: none"> Line: If the system parameter <i>Invoice Freight as Revenue</i> and profile option <i>EBTax: Allow Override of Tax Code</i> are set to YES. Otherwise, the value is Freight Charges.

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
DESCRIPTION	Not Null	Varchar2(240)	<p><i>For a Sales order or return line:</i></p> <p>Order Management inserts MTL_SYSTEM_ITEM.S.DESCRPTION for the item or if order line uses item identifiers (item/generic item):</p> <ul style="list-style-type: none"> if a description exists, customer item/generic item description is passed. if the description is null then mtl_system_items_vl.description is passed <p><i>For Freight charges:</i> Order Management inserts description of the change (OE_CHARGE_TYPES.description) for shipment freight charges.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales Order lines and Return lines: Customer and generic items get customer and generic item descriptions if it exists. Otherwise, the mtl_system_items_vl.description for service lines, concatenates the following: service item description, service quantity, serviced item description, service start date, and service end date. Freight Charges: If passed as a revenue line, then the freight item description appears. Otherwise, the standard description is transferred from the AR lookup.

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
CURRENCY_CODE	Not Null	Varchar2(30)	oe_order_headers.transactional_curr_code for all shipped/non shipped order/return lines.	<p>Shipped order lines and Discount lines: header.transactional_currency_code</p> <p>Non-shipped line or Return lines: header.transactional_currency_code</p> <p>Freight Charges: charge.transactional_currency_code</p>
AMOUNT	NA	NUMBER	<p><i>For a Sales order or return line:</i> Order Management inserts a calculated amount (OE_ORDER_LINES.UNIT_SELLING_PRICE multiplied by OE_ORDER_LINES.ORDERED_QUANTITY) based on the calculated quantity. Order Management rounds the amount based on the minimum accounting unit and precision associated with the currency of the order.</p> <p>The amount sign will match the sign on the quantity based on the value of RA_CUST_TRX_TYPES.CREATION_SIGN.</p> <p><i>For Freight charges:</i> Order Management inserts: oe_charge_lines_v.charge_amount</p>	<p>Sales Order lines and Return lines: if the system parameter Show discount details on Invoice is Yes, then quantity_to_invoice * line.list_price. Otherwise, the following will be calculated quantity * line.list_price. For returns, the amount will be * -1.</p> <p>Note: All prices are based on Ordered UOM.</p>

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
CUST_TRX_TYPE_NAME	NA	Varchar2(20)	Order Management does not insert a value into this column	NA
CUST_TRX_TYPE_ID	NA	NUMBER (15)	Please refer to the heading CUST_TRX_TYPE_ID at the end of this table for more information	Shipped order line and Discount lines. Return lines: See notes on cust_trx_type_id under defining OE Transaction Type form. Freight Charges: Same as order line.
TERM_NAME	NA	Varchar2(15)	Order Management does not insert a value into this column.	NA
TERM_ID		NUMBER (15)	<p><i>For a Sales order line:</i> Order Management inserts OE_ORDER_LINES.PAYMENT_TERM_ID.</p> <p><i>For a Return line:</i> Order Management does not insert a value into this column.</p> <p><i>For Freight charges:</i> Order Management inserts: oe_order_lines_all.payment_term_id.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales order line, discount lines: line.payment_term_id • Return lines: None • Freight Charges: line.payment_term_id
ORIG_SYSTEM_BILL_CUSTOMER_REF	NA	Varchar2(240)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
ORIG_SYSTEM_BILL_CUSTOMER_ID	NA	NUMBER(15)	Order Management inserts: SELECT customer_id FROM oe_invoice_to_orgs_v WHERE organization_id = oe_order_lines.invoice_to_org_id	customer_id associated with the line.invoice_to_site_id
ORIG_SYSTEM_BILL_ADDRESS_REF	NA	Varchar2(40)	Order Management does not insert a value into this column.	NA
ORIG_SYSTEM_BILL_ADDRESS_ID	NA	NUMBER(15)	Order Management inserts: SELECT address_id FROM oe_invoice_to_orgs_v WHERE organization_id = oe_order_lines.invoice_to_org_id	address_id associated with the line.invoice_to_site_id
ORIG_SYSTEM_BILL_CONTACT_REF	NA	Varchar2(40)	Order Management does not insert a value into this column.	NA
ORIG_SYSTEM_BILL_CONTACT_ID	NA	NUMBER(15)	Order Management inserts: oe_order_lines.invoice_to_contact_id	line.invoice_to_contact_id
ORIG_SYSTEM_SHIP_CUSTOMER_REF	NA	Varchar2(40)	Order Management does not insert a value into this column	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
ORIG_SYSTEM_SHIP_CUSTOMER_ID	NA	NUMBER(15)	Order Management inserts SELECT customer_id FROM oe_ship_to_orgs_v WHERE organization_id = oe_order_lines.ship_to_org_id	customer_id associated with line.ship_to_site_use_id
ORIG_SYSTEM_SHIP_ADDRESS_REF	NA	Varchar2(40)	Order Management does not insert a value into this column.	NA
ORIG_SYSTEM_SHIP_ADDRESS_ID	NA	NUMBER(15)	<i>For a Sales order or return line:</i> Order Management inserts RA_SITE_USES.ADDRESS_ID FROM RA_ADDRESSES WHERE RA_SITE_USES.SITE_USE_ID = NVL(OE_ORDER_LINES.SHIP_TO_SITE_USE_ID, OE_ORDER_HEADERS.SHIP_TO_SITE_USE_ID))). <i>For Freight charges:</i> Order Management inserts RA_SITE_USES.ADDRESS_ID FROM RA_ADDRESSES WHERE RA_SITE_USES.SITE_USE_ID = OE_ORDER_HEADERS.SHIP_TO_SITE_USE_ID).	The address_id associated with ship_to_site_use_id of line.

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
ORIG_SYSTEM_SHIP_CONTACT_REF	NA	Varchar2(24)	Order Management does not insert a value into this column.	NA
ORIG_SYSTEM_SHIP_CONTACT_ID	NA	NUMBER(15)	Order Management inserts: oe_order_lines.ship_to_contact_id	The available values are: <ul style="list-style-type: none"> Sales Order lines and Return lines: line.ship_to_contact_id. Freight Charges: line.ship_to_contact_id.
ORIG_SYSTEM_SOLD_CUSTOMER_REF	NA	Varchar2(240)	Order Management does not insert a value into this column.	NA
ORIG_SYSTEM_SOLD_CUSTOMER_ID	NA	NUMBER(15)	Order Management inserts OE_ORDER_HEADERS.SOLD_TO_ORG_ID.	address_id associated with sold_to_org_id of line
LINK_TO_LINE_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
LINK_TO_LINE_CON TEXT	NA	Vharchar2(30)	Order Management does not insert a value into this column.	NA
LINK_TO_LINE_ATTR IBUTE1 - 15	NA	Vharchar2(30)	Order Management does not insert <i>any</i> values into <i>any</i> of these columns.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
RECEIPT_METHOD_NAME	NA	Vharchar2(30)	Order Management does not insert a value into this column.	NA
RECEIPT_METHOD_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
CONVERSION_TYPE	Not NULL	Varchar2(30)	Order Management inserts NVL(OE_ORDER_HEADERS.CONVERSION_TYPE, <i>User</i>).	nvl (header.conversion_type,'User')
CONVERSION_DATE	NA	Date	Order Management inserts OE_ORDER_HEADERS.CONVERSION_DATE.	header.conversion_date
CONVERSION_RATE	NA	NUMBER	Order Management inserts OE_ORDER_HEADERS.CONVERSION_RATE.	header.conversion_rate
CUSTOMER_TRX_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
TRX_DATE	NA	Date	Order Management does not insert a value into this column.	NA
GL_DATE	NA	Date	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
DOCUMENT_NUMBER	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
TRX_NUMBER	NA	Varchar2(40)	If the OM: <i>Invoice Numbering Method</i> profile is set to Automatic, AutoInvoice determines a unique number for this transaction. If the profile is set to <i>Delivery Name</i> , Order Management inserts a delivery name. An index is appended if the delivery has more than one invoice. For example, Order Management might insert <i>delivery</i> for the first invoice, <i>delivery-1</i> for the second, <i>delivery-2</i> for the third, and so on.	If the OM: Invoice Numbering Method is set to Delivery Name, then insert the delivery name.
LINE_NUMBER	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
QUANTITY	NA	NUMBER	<p>Order Management inserts a calculated quantity based on the type of line being interfaced. This calculation is based on rules for fulfillment, over and under shipments, required for revenue and what was previously invoiced.</p> <p><i>For a Sales order line:</i> The quantity will be either negative or positive, depending on the value of RA_CUST_TRX_Types.CREATION_SIGN associated with the invoice type for the sales order. If RA_CUST_TRX_Types.CREATION_SIGN is N, then quantity passed is -1 multiplied by the quantity calculated.</p>	<p>Use the following hierarchy for sales order lines, return lines, and discount lines:</p> <ul style="list-style-type: none"> • Fulfilled quantity • Shipped quantity • Ordered quantity • Freight Charges: None
QUANTITY_ORDERED	NA	NUMBER	<p><i>For a Sales order line or return line:</i> Order Management inserts OE_ORDER_LINES.ordered_quantity.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column</p>	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales Order lines and Return lines: line.ordered_quantity • Freight Charges: None • Discount Lines: line.ordered_quantity

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
UNIT_SELLING_PRICE	NA	NUMBER	<p><i>For a Sales order line or return lines:</i> Order Management inserts OE_ORDER_LINES.UNIT_SELLING_PRICE.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	<p>Sales Order lines and Return lines: Depends on whether you want to show discount details:</p> <ul style="list-style-type: none"> • Show discounts Y (Yes): line.unit_list_price. • Show discounts N (No): line.selling_price. • Freight Charges: None. • Discount lines: Use pricing API. Show discounts Y (Yes).
UNIT_SELLING_PRICE_PER_PQTY	NA	NUMBER	Unit Selling Price Per Pricing Quantity.	NA
UNIT_STANDARD_PRICE	NA	NUMBER	<p><i>For a Sales order line or return line:</i> Order Management inserts OE_ORDER_LINES.UNIT_LIST_PRICE.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales Order lines and Return lines: line.list_price. • Freight Charges: None. • Discount lines: Use pricing API. Show discounts Y (Yes).

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
PRINTING_OPTION	NA	Varchar2(20)	Order Management does not insert a value into this column.	NA
INTERFACE_STATUS	NA	Varchar2(1)	Order Management does not insert a value into this column.	NA
REQUEST_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
RELATED_BATCH_SOURCE_NAME	NA	Varchar2(50)	Order Management does not insert a value into this column.	NA
RELATED_TRX_NUMBER	NA	Varchar2(20)	Order Management does not insert a value into this column.	NA
RELATED_CUSTOMER_TRX_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
PREVIOUS_CUSTOMER_TRX_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
CREDIT_METHOD_F OR_ACCT_RULE	NA	Varchar2(30)	<p><i>For a Sales order line:</i> Order Management does not insert a value into this column.</p> <p><i>For a Return line:</i> Order Management inserts from based upon the following hierarchy:</p> <ol style="list-style-type: none"> 1. Line transaction type 2. Order transaction type 3. else insert constant value <i>LIFO</i> 	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales Order lines: None • Return lines
CREDIT_METHOD_F OR_INSTALLMENTS	NA	Varchar2(30)	<p><i>For a Sales order line:</i> Order Management does not insert a value into this column.</p> <p><i>For a Return line:</i> Order Management inserts from based upon the following hierarchy:</p> <ol style="list-style-type: none"> 1. Line transaction type 2. Order transaction type 	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales order lines: None • Return lines

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
REASON_CODE	NA	Varchar2(30)	<p><i>For a Sales order line:</i> Order Management does not insert a value into this column.</p> <p><i>For a Return line:</i> Order Management inserts OE_ORDER_LINES. RETURN_REASON_CODE</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales order lines: None Return lines: line.return_reason_code
TAX_RATE	NA	NUMBER	Order Management does not insert a value into this column.	NA
TAX_CODE	NA	Varchar2(50)	If <i>Tax: Allow Override of Tax Code = Yes</i> , then pass oe_order_lines.tax_code	<p>The available values are:</p> <ul style="list-style-type: none"> Sales order lines, return lines, and discount lines: if profile option EBTAX: Allow Override of Tax Code is set to YES, then interface line.tax_code, otherwise the interface returns a NULL value. Freight Charges: None
TAX_PRECEDENCE	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
EXCEPTION_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
EXEMPTION_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
SHIP_DATE_ACTUAL	NA	Date	<p><i>For a Shipped order lines and for Freight charge lines:</i> Order Management inserts: wsh_new_deliveries.initial_pickup_date.</p> <p><i>For a Non-shipped order line:</i> If profile OM: Set Receivables Transaction Date as Current Date for Non-Shippable Lines is Yes then sysdate otherwise NULL</p> <p><i>For Freight charges:</i> Order Management inserts OE_ORDER_LINE_ACTUAL_SHIPMENT_DATE.</p> <p>For Return line: rcv_transactions.transaction_date</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Shipped order line and discount lines: line.actual_shipment_date Non-shipped order line: If profile option OM: Set Receivables Transaction Date as Current Date for Non-Shippable Lines is Yes then sysdate otherwise NULL. Freight charges: line.actual_shipment_date Return line: rcv_transactions.transaction_date

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
FOB_POINT	NA	Varchar2(20)	<p>Order Management only populates this column if the order line being invoiced has been shipped.</p> <p><i>For a Shipped order line:</i> Order Management inserts OE_ORDER_LINES.FOB_POINT_CODE.</p> <p><i>For a Non-shipped order line:</i> Order Management does not insert a value into this column.</p> <p><i>For Freight charges:</i> Order Management inserts OE_ORDER_LINES.FOB_POINT_CODE.</p>	NA
SHIP_VIA	NA	Varchar2(20)	<p>Either nvl(wsh_carrier_shipping_methods_v.freight_code or oe_order_lines.freight_carrier_code)</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Shipped order line, discount line: nvl (freight carrier associated with del.shipping_method_code, line..freight_carrier_code) Non-shipped order line: None Freight Charges: same as line.

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
WAYBILL_NUMBER	NA	Varchar2(50)	<p><i>For a Shipped order line and For Freight charges:</i> Order Management inserts: substr(wsh_new_delivery.waybill,1,30)</p> <p><i>For a Non-shipped order line:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Shipped order line: delivery waybill Non-shipped order line: None Freight charges: delivery waybill Discount lines: None
INVOICING_RULE_NAME	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA
INVOICING_RULE_ID	NA	NUMBER(15)	<p><i>For a Sales order line and for Freight charge lines:</i> Order Management inserts DECODE(ACCOUNTING_RULE_ID,1,NULL,INVOICING_RULE_ID).</p> <p><i>For a Non-shipped order line:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales order lines and discount lines: decode (accounting_rule_id,1,NULL,invoicing_rule_id) Return lines: None Freight charges: decode(accounting_rule_id,1,NULL,invoicing_rule_id)

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
ACCOUNTING_RULE_NAME	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA
ACCOUNTING_RULE_ID	NA	NUMBER(15)	<p><i>For a Sales order line and for Freight charge lines:</i> Order Management inserts DECODE(ACCOUNTING_RULE_ID,1,NULL,ACCOUNTING_RULE_ID).</p> <p><i>For a Return line:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales Order lines and Discount lines: decode(accounting_rule_id,1,NULL,accounting_rule_id) Return lines: None Freight charges: decode(accounting_rule_id,1,NULL,accounting_rule_id)
ACCOUNTING_RULE_DURATION	NA	NUMBER(15)	If the accounting rule is of type <i>Variable Duration</i> then insert <i>Service Duration</i> for Service lines.	<p>The available values are:</p> <ul style="list-style-type: none"> Service line: If accounting_rule is of type variable duration (ACC_DUR) then insert service duration. Other line: None.
RULE_START_DATE	NA	Date	Order Management inserts oe_order_lines.service_start_date into this column.	oe_order_lines.service_start_date

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
PRIMARY_SALESREP_NUMBER	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA
PRIMARY_SALESREP_ID	NA	NUMBER(15)	Order Management inserts NVL (OE_ORDER_LINES.SALESREP_ID, OE_ORDER_HEADERS.SALESREP_ID).	nvl(line.salesrep_id, header.salesrep_id)
SALES_ORDER	NA	Varchar2(50)	Order Management inserts OE_ORDER_HEADERS.ORDER_NUMBER	header.order_number
SALES_ORDER_LINE	NA	Varchar2(30)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.LINE_NUMBER.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales order lines and return lines: line.line_number • Freight Charges: None • Discount lines: line.line_number
SALES_ORDER_DATE	NA	Date	Order Management inserts OE_ORDER_HEADERS.ORDERED_DATE.	header.ordered_date
SALES_ORDER_SOURCE	NA	Varchar2(50)	Order Management inserts your value for the OM: <i>Source Code</i> profile option.	profile option OM: Source code

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
SALES_ORDER_REVISION	NA	NUMBER	Order Management does not insert a value into this column.	NA
PURCHASE_ORDER	NA	Varchar2(50)	Order Management inserts OE_ORDER_HEADERS.LINES.CUST_PO_NUMBER.	line.cust_po_number
PURCHASE_ORDER_REVISION	NA	Varchar2(50)	Order Management does not insert a value into this column.	NA
PURCHASE_ORDER_DATE	NA	Date	Order Management does not insert a value into this column.	NA
AGREEMENT_NAME	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
AGREEMENT_ID	NA	NUMBER(15)	<p><i>For a Sales order line:</i> Order Management inserts OE_ORDER_LINES.AGREEMENT_ID.</p> <p><i>For a Return line with a purchase order or sales order reference:</i> Order Management inserts the AGREEMENT_ID from the referenced order line.</p> <p><i>For return lines with an invoice reference:</i> Order Management does not insert a value into this column.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales Order lines and Discount lines: line.agreement_id Return lines: agreement_id from the referenced order header. Freight charges: None
MEMO_LINE_NAME	NA	Varchar2(50)	Order Management does not insert a value into this column.	NA
MEMO_LINE_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INVENTORY_ITEM_ID	NA	NUMBER(15)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.INVENTORY_ITEM_ID.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column. If the system parameter Invoice Freight As Revenue is set to Yes, the value of the system parameter Inventory Item For Freight is used.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales Order lines and Return lines: line.inventory_item_id Freight Charges: can be either: <ul style="list-style-type: none"> NULL if the parameter Invoice Freight as Revenue is set to No, or set by system parameter Inventory Item for Freight Discount lines: None
MTL_SYSTEM_ITEMS_SEG1....20	NA	Varchar2(30)	Order Management does not insert values into any of these columns.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
REFERENCE_LINE_ID	NA	NUMBER(15)	<p><i>For a Sales order line:</i> Order Management inserts OE_ORDER_LINES.COMMITMENT_ID.</p> <p><i>For a Return line:</i> Order Management inserts OE_ORDER_LINES.CREDIT_INVOICE_LINE_ID.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> • Sales Order lines: cust_trx_id from the line.commitment_id • Return lines: line.reference_line_id • Freight charges: None • Discount lines: pa.line_id
REFERENCE_LINE_CONTEXT	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA
REFERENCE_LINE_ATTRIBUTES1....15	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
TERRITORY_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	<p>The value is derived from the following:</p> <ul style="list-style-type: none"> • Get default territory from AR system parameter. • Get territory ID from one of the following tables. If default territory is <ul style="list-style-type: none"> • SALES: join ra_sales_rep_territory using the sales rep id. • BILL: join ra_site_uses using line.invoice_to_org_id • SHIP: join ra_site_uses using line.ship_to_org_id.
TERRITORY_SEGMENTS1.....20	NA	Varchar2(25)	Order Management does not insert a value into this column.	Join the territory ID above with ra_territories to get these segments.

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
ATTRIBUTE_CATEGORY	NA	Varchar2(30)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.CONTEXT.</p> <p><i>For Freight charges:</i> Order Management inserts OE_CHARGE_LINES_V.CONTEXT.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales order lines and Return lines: line.context Freight charges: charge.context Discount lines: pa.context
ATTRIBUTE1...15	NA	Varchar2(150)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.ATTRIBUTE1-15.</p> <p><i>For Freight charges:</i> Order Management inserts OE_CHARGE_LINES_V.ATTRIBUTE1-15.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales order lines and return lines: line.attribute1-15 Freight charges: charge.attribute1-15 Discount lines: pa.attribute1-15
HEADER_ATTRIBUTE_CATEGORY	NA	Varchar2(30)	<p><i>For Sales order or return line:</i> Order Management inserts OE_ORDER_HEADERS.CONTEXT.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	header.context

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
HEADER_ATTRIBUTE 1...15	NA	Varchar2(150)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.ATTRIBUTE1-15.</p> <p><i>For Freight charges:</i> Order Management inserts oe_order_headers.attribute1-15.</p>	header.attribute1-15
COMMENTS	NA	Varchar2(240)	Order Management does not insert a value into this column.	NA
INTERNAL_NOTES	NA	Varchar2(240)	Order Management does not insert a value into this column.	NA
INITIAL_CUSTOMER_TRX_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
USSGL_TRANSACTION_CODE_CONTEXT	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA
USSGL_TRANSACTION_CODE	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA
ACCTD_AMOUNT	NA	NUMBER	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
CUSTOMER_BANK_ACCOUNT_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
CUSTOMER_BANK_ACCOUNT_NAME	NA	Varchar2(25)	Order Management does not insert a value into this column.	NA
UOM_CODE	NA	Varchar2(3)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.SHIPPING_QUANTITY_UOM.</p> <p><i>For Freight charges:</i> Order Management does not insert a value into this column.</p>	<p>The available values are:</p> <ul style="list-style-type: none"> Sales order lines, return lines, and discount lines: line.order_quantity_UOM. Freight charges: if line type = Freight, pass NULL value; otherwise the freight item UOM.
UOM_NAME	NA	Varchar2(25)	Order Management does not insert a value into this column.	NA
DOCUMENT_NUMBER_SEQUENCE_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
REASON_CODE_NAME	NA	Varchar2(30)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
VAT_TAX_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
LOCATION_RATE_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
REASON_CODE_MEANING	NA	Varchar2(80)	Order Management does not insert a value into this column.	NA
LAST_PERIOD_TO_CREDIT	NA	NUMBER	Order Management does not insert a value into this column.	NA
PAYING_CUSTOMER_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
PAYING_SITE_USE_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
TAX_EXEMPT_FLAG	NA	Varchar2(10)	Order Management inserts OE_HEADERS.TAX_EXEMPT_FLAG for order lines.	The available values are: <ul style="list-style-type: none"> Freight: None Other: nvl(line.tax_exempt_flag, S)
SALES_TAX_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
CREATED_BY	NA	NUMBER(15)	Order Management enters an identification number to identify the user who created the record. No Validation occurs	who id
CREATION_DATE	NA	Date	Order Management enters the current system date when a record is created.	sysdate of creation time
LAST_UPDATED_BY	NA	NUMBER(15)	Order Management enters an identification number to identify the user who created or who most recently modified the record. No Validation occurs	who id
LAST_UPDATE_DATE	NA	Date	Order Management enters the current date when a record is updated. Standard Validation occurs	sysdate of update time
LOCATION_SEGMENT_ID	NA	NUMBER(15)	Order Management does not insert a value into this column.	NA
TAX_EXEMPT_REASON_CODE	NA	Varchar2(30)	Order Management inserts OE_ORDER_LINE.TAX_EXEMPT_REASON_CODE for order lines.	Freight: None Other: line.tax_exempt_reason_code

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
TAX_EXEMPT_NUMB ER	NA	Varchar2(8 0)	Order Management inserts OE_ORDER_LINE.T AX_EXEMPT_NUMB ER for order lines.	Freight: None Other: line.tax_exempt_num
TAX_EXEMPT_REASO N_CODE_MEANING	NA	Varchar2(8 0)	Order Management does not insert a value into this column.	NA
Line_Gdf_Attr_Categor y	NA	NA	NA	Populated by Globalization API called by OM.
Line_Gdf_Attr1-20,	NA	NA	NA	JG_ZZ_OM_Common _PKG.Copy_Gdff
PARENT_LINE_ID	NA	NUMBER(15)	<i>For child line in a configuration:</i> Order Management inserts oe_order_lines.top_m odel_line_id <i>For Service Lines:</i> Order Management inserts service parent line id	<ul style="list-style-type: none"> For child line in a configuration: oe_order_lines.to p_model_line_id For Service Lines: service parent line id
DEFERRAL_EXCLUSI ON_FLAG	NA	VARCHA R2(1)	For freight lines: Y	For freight lines: Y otherwise NULL
PAYMENT_TRXN_EX TENSION_ID	NA	NUMBER(15)	Order Management inserts oe_payments. txn_extension_id	Oe_payments.txn_ex tension_id
RULE_END_DATE	NA	DATE	<i>For service lines:</i> If order lines uses daily rate accounting rules then Order Management inserts oe_order_lines.service _end_date	If accounting_rule is of type PP_DR_ALL or PP_DR_PP variable duration oe_order_lines.service _end_date

Note: ¹Non-Shipped order lines

The following information refers to the Interface_Line_Attribute3 attribute described in the preceding table:

- No-Ship Cycle

All order lines should be interfaced with INTERFACE_LINE_ATTRIBUTE3 = 0.

- Ship Cycle

- All non-shippable STANDARD lines should be invoiced separately with INTERFACE_LINE_ATTRIBUTE3 = 0.

- All non-shippable KIT lines that are not part of a PTO model should be invoiced separately with INTERFACE_LINE_ATTRIBUTE3 = NVL(First Delivery associated with included items. A 0 value appears if there are no shippable included items.

Note: If there are shippable included items, the KIT line will be Pick Release Released and Ship Confirm Confirmed.

- All non-shippable model components: Line Type = MODEL, CLASS, KIT, or STANDARD without shippable included items should be invoiced with the following:

INTERFACE_LINE_ATTRIBUTE3 = NVL(First Delivery of shippable model invoiced with non-shippable model components. A 0 value appears if no delivery is available.

- All non-shippable model components: Line Type = MODEL, CLASS, or KIT with shippable included items should be invoiced with:

INTERFACE_LINE_ATTRIBUTE3 = NVL(Delivery associated with included items).

- All shippable model components: Line Type = MODEL, CLASS, KIT, or STANDARD should be invoiced with:

INTERFACE_LINE_ATTRIBUTE3 = Delivery.

Derived Values

CUST_TRX_TYPE_ID

Value is determined based upon the following hierarchy:

For a Sales order line:

Line transaction type

Order Transaction type

System Parameter *Invoice Transaction type*

For referenced return line:

Retrieve receivables transaction type from:

1. 1. Derive credit memo type based on return context and reference information (that is, reference invoice or line type and order type of reference line)
2. Line transaction type
3. Order transaction type
4. System parameter Credit memo Transaction Type

For non-referenced return line:

1. Line transaction type
2. Order transaction type
3. Derive credit memo type based on return context and reference information.
4. System Parameter *OM: Credit memo Transaction Type*

For freight charges:

- The freight charges value is the same as the order/return line that is interfacing the freight charge.

RA_INTERFACE_SALESCREDITS

Order Management inserts one row for each sales credit row according to the following sequence:

1. Insert sales credits associated with the line; if none exists but the line is part of a configuration (ITEM_TYPE_CODE is CLASS, KIT or STANDARD and OPTION_FLAG is Y), then;
2. Insert sales credits associated with the model *parent* line; if none exists, then;
 - Insert sales credits associated with the order header.

RA_INTERFACE_SALESCREDITS

Column	Type	Description	Inserted by Invoicing Module
INTERFACE_SALE SCREDIT_ID	NUMBER(15)	Order Management does not insert a value into this column.	None
INTERFACE_LINE _ID	NUMBER(15)	Order Management does not insert a value into this column.	None
INTERFACE_LINE _CONTEXT	Varchar2(30)	Order Management inserts your value for the OM: <i>Source Code profile option.</i>	Profile option OM: Source Code
INTERFACE_LINE _ATTRIBUTE1	Varchar2(30)	Order Management inserts OE_ORDER_LINE. ORDER_NUMBER.	Same as line inserted into ra_interface_lines
INTERFACE_LINE _ATTRIBUTE2	Varchar2(30)	Order Management inserts OE_TRANSACTION. N_TYPES.NAME.	Same as line inserted into ra_interface_lines
INTERFACE_LINE _ATTRIBUTE3	Varchar2(30)	<i>For a Shipped order line and for Freight Charges:</i> Order Management inserts substr(wsh_new_d eliveries.name1..30 <i>For a Non-shipped order line or a Return Line:</i> Order Management inserts 0 (zero).	Same as line inserted into ra_interface_lines

Column	Type	Description	Inserted by Invoicing Module
INTERFACE_LINE_ATTRIBUTE4	Varchar2(30)	<p><i>For a Shipped order line and for Freight Charges:</i> Order Management inserts: substr(wsh_new_deliveries.waybill1..30)</p> <p><i>For a Non-shipped order line or a Return Line:</i> Order Management inserts 0 (zero).</p> <p>Order Management inserts SUBSTR (WSH_DELIVERY_LINE_STATUS_V.WAYBILL, 1, 30).</p>	Same as line inserted into ra_interface_lines
INTERFACE_LINE_ATTRIBUTE5	Varchar2(30)	Order Management inserts the number of times the order, freight charge, or return line has been interfaced for invoice or credit.	Same as line inserted into ra_interface_lines
INTERFACE_LINE_ATTRIBUTE6	Varchar2(30)	<p><i>For a Sales order or return line:</i> Order Management inserts OE_ORDER_LINES.LINE_ID.</p> <p><i>For Freight charges:</i> OE_CHARGE_LINES.CHARGE_ID.</p>	Same as line inserted into ra_interface_lines

Column	Type	Description	Inserted by Invoicing Module
INTERFACE_LINE_ATTRIBUTE7	Varchar2(30)	For a Shipped order line, a Return line, or Freight charges: Order Management inserts 0 (zero).	Same as line inserted into ra_interface_lines
INTERFACE_LINE_ATTRIBUTE8	Varchar2(30)	<i>For a Shipped order line:</i> Order Management inserts WSH_DEPARTURES. BILL_OF_LADING . <i>For a Return line, or Freight charges:</i> Order Management inserts 0 (zero)	Same as line inserted into ra_interface_lines
INTERFACE_LINE_ATTRIBUTE9	Varchar2(30)	Order Management inserts the customer item number if one is defined. Otherwise, it inserts 0 (zero).	Same as line inserted into ra_interface_lines
INTERFACE_LINE_ATTRIBUTE10	Varchar2(30)	Order Management inserts OE_ORDER_LINES . SHIP_FROM_ORG_ID.	Same as line inserted into ra_interface_lines

Column	Type	Description	Inserted by Invoicing Module
INTERFACE_LINE_ATTRIBUTE11	Varchar2(30)	oe_price_adjustments.price_adjustment_id for discount lines: system paramter Show Discount Details on Invoice must be Yes. For all other lines '0' is passed.	Same as line inserted into ra_interface_lines
INTERFACE_LINE_ATTRIBUTE12....15	Varchar2(30)	Order Management does not insert a value into this column.	Same as line inserted into ra_interface_lines
SALESREP_NUMBER	Varchar2(30)	Order Management does not insert a value into this column.	None
SALESREP_ID	NUMBER(15)	Order Management inserts OE_SALES_CREDIT.SALESREP_ID.	oe_sales_credits.salesrep_id
SALES_CREDIT_TYPE_NAME	NA	Order Management does not insert a value into this column.	None
SALES_CREDIT_TYPE_ID	NUMBER(15)	Order Management inserts OE_SALES_CREDIT.SALES_CREDIT_TYPE_ID.	oe_sales_credits.sales_credit_type_id
SALES_CREDIT_AMOUNT_SPLIT	NUMBER	Order Management does not insert a value into this column.	None

Column	Type	Description	Inserted by Invoicing Module
SALES_CREDIT_PERCENT_SPLIT	NUMBER	Order Management inserts OE_SALES_CREDIT.PERCENT.	oe_sales_credits.percent
INTERFACE_STATUS	Varchar(1)	Order Management does not insert a value into this column.	None
REQUEST_ID	NUMBER(15)	Order Management does not insert a value into this column.	None
ATTRIBUTE_CATEGORY	Varchar(30)	Order Management inserts OE_SALES_CREDIT.CONTEXT.	oe_sales_credits.context
ATTRIBUTE1...15	Varchar(150)	Order Management inserts OE_SALES_CREDIT.ATTRIBUTE1....15.	oe_sales_credits.Attribute1-15
Org_Id	NUMBER (15)	Operating Unit	line.org_id
Created_By	NUMBER (15)	Username	who id
Creation_Date	Date	Date of creation	creation date
Last_Updated_By	NUMBER (15)	User name	who id
Last_Update_Date	Date	Last Update Date	current date

Accounts Receivables Contingency Columns

AR_INTERFACE_CONTS

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
INTERFACE_CONTINGENCY_ID	NA	NUMBER	Order Management does not insert a value into this column.	None
INTERFACE_LINE_CONTEXT	NA	Varchar2(30)	Same as RA_INTERFACE_LINES	Same as RA_INTERFACE_LINES
INTERFACE_LINE_ATTRIBUTE1-15	NA	Varchar2(30)	Same as RA_INTERFACE_LINES	Same as RA_INTERFACE_LINES
CONTINGENCY_ID	Not null	NUMBER(15)	Order Management inserts Oe_order_lines.continency_id into this column	Oe_order_lines.continency_id
EXPIRATION_DAYS	NA	NUMBER	Order Management inserts oe_order_lines.REVREC_EXPIRATION_DAYS into this column	oe_order_lines.REVREC_EXPIRATION_DAYS
EXPIRATION_DATE	NA	DATE	Order Management does not insert a value into this column.	None

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
COMPLETED_FLAG	NA	VARCHAR2(1)	Order Management inserts value of 'Y' if the contingency is already removed on the order line.	'Y' if contingency is already removed
COMPLETED_BY	NA	NUMBER	Order Management does not insert a value into this column.	None
EXPIRATION_EVENT_DATE	NA	DATE	Order Management does not insert a value into this column.	None
CONTINGENCY_CODE	NA	VARCHAR2(30)	Order Management does not insert a value into this column.	None
INTERFACE_LINE_ID	NA	NUMBER	Order Management does not insert a value into this column.	None
INTERFACE_STATUS	NA	VARCHAR2(1)	Order Management does not insert a value into this column.	None

Column Name	Null?	Type	Description	What is inserted by Invoicing Module
ATTRIBUTE_CATEGORY	NA	VARCHAR2(30)	Order Management does not insert a value into this column.	None
ATTRIBUTE1-15	NA	VARCHAR2(30)	Order Management does not insert a value into this column.	None

Telecommunications Services Orders (MACD) Interfaces

Process_MACD_Order Application Program Interface

The Process_MACD_Order API can be used to:

- Create MACD orders from instances
- Process MACD actions
- Create and process MACD actions

The API accepts instances and header information (sold_to_org_id) to create MACD orders. It also accepts MACD actions to process the order. Instances can be added to the existing order by passing the header_id. To process actions on an existing order, you can call the API with the lines table and an action. You can initiate agent actions that need processing on order lines with Configurator attributes and Configurator extended attributes.

Process_MACD_Order API Features

You can create MACD orders by passing instance information and Sold to org. You need to pass customer related information to create the order before adding the MACD lines to the order.

Process MACD Order API is also used to add MACD orders lines by passing the instance information and order header information. The API will look for the root item of the passed instances, and if the root instance is present in the order, then the new instances are added to the existing root. Alternatively the API creates the complete top structure to fit the passed in instances.

You can use the API to create and process MACD orders by passing the instance information, Sold to org (Customer) information, and MACD action. The action can be passed at the API level or at the instance level. First the API creates the order lines for the instances and then the MACD action is applied.

You can use the Process MACD Order API to process actions on existing MACD orders by passing in the line table with actions. The action can be passed at the API level or at the line level.

Functional Overview

The steps to create and process MACD orders are described below.

Create a MACD order or Add instances to an existing order

To create MACD orders from instances or add instances to existing orders, the Process MACD Order API accepts item instances and order header information (header_id or sold_to_org_id), and extended attributes as input.

1. Instance table needs to be passed to the API.
2. Bill_to, ship_to passed by the calling entity will be validated against the sold_to column on the new order.
3. P_instance_table and p_x_line_table are mutually exclusive. P_instance_table should be passed to the API for create or create and process operation. P_x_line_table should be passed if user needs to process the existing order.
4. Sold_to_org_id and header_id are mutually exclusive, only one of them should be used as input. If a new order needs to be created, Sold_to_org_id is used as the input or header_id needs to be passed to append the instances converted lines to existing order.
5. All selected instances should belong to the same sold_to_org_id.
6. Order Management will validate and raise an error if the Install Base owner is not same on the all instances passed to this API.
7. MACD Action should be passed through the parameter p_macd_action, if the same action applies to all the instances passed in the instance table or lines passed in the lines tables. If caller needs to perform multiple actions then pass the action on each instance record or on the line record. API processing fails if the action is passed on both instance/line table and p_macd_action.
8. This API accepts the Operating Unit as optional input. The instance or order header information or extended attributes need to call the MOAC APIs if the orders need to be processed or created in an operating unit apart from the default operating unit. If an operating unit is not specified, the order lines will be created in the

default operating unit. Verify the operating unit value that is based on the Ship-to and Bill-to on the item instances. If the operating unit of the Ship-to and Bill-to are different then an error is raised. All the instances should belong to the same operating unit.

9. If the header_id is passed, the system will append the instances to the existing order and create container and child order lines in that order. The API searches for all order lines in that sequence to verify if one referring to the parent container item of the passed in item instances already exists. If such a container is found, instances would be added to the existing container, otherwise a new container would be created.
10. p_MACD_Action should not be passed to existing MACD Orders. We need to use actions only if we need to process them.
11. If the API creates order lines for more than one network container, the first container line_id is returned.
12. The API also returns the line table with order line records created as a result of the API call.

Create and Process MACD orders

To create and process MACD order the Process MACD Order API accepts item instances with action and the order header information (header_id and sold_to), extended attributes, and p_macd_action as input.

You need to follow the steps listed in the section Create a MACD order or Add instances to an existing order, page 8-53. After that, follow the steps given below to process an order:

1. If the MACD action is passed (either in p_macd_action or in the instance record) it calls the CZ public API CZ_CF_API.Validate and passes the first class CZ attributes and extended CZ attributes along with the operations to the CZ API.
 - Order Management will implicitly treat a UPDATE on ordered quantity = 0 as DELETE and will take the following two rules into consideration.
 - A DELETE operation on order lines sends DELETE operation to CZ if the component does not have a base line rev number, that is if the component is a new component that has been added to a completely new configuration or a re-configuration.
 - In case of DELETE operation on order line with a baseline rev number > 0 , Order Management will error out. This is because CZ will not support REVERT action in this phase.
 - DISCONTINUE operation on an order line with a component that has a base

line rev number > 0 (real MACD re-configuration) will send DELETE to CZ. DISCONTINUE operation on a MACD order line without base line rev will raise a Order Management error as you cannot discontinue a component which is not yet in Install Base.

- DISCONTINUE operation on a non-MACD order line should raise error.
2. Synchronize the order lines to configuration changes resulted due to the CZ_CF_API.Validate API call.

Process MACD Actions on existing orders

Process MACD Orders API accepts the following as input: line table (which includes the CZ first class attributes), extended attributes tables, and MACD action.

1. Validate the input parameters.
 - Validate the Header_id, line_id and operation passed on the line table e.g. CREATE operation on order lines is not supported.
 - Validate if the index pointing back to the line table exists on the CZ extended attributes table.
2. Process the action that is passed to attribute OPERATION in the input line table, which will determine the work done by the API.
 - UPDATE and DELETE operations are supported on order lines. CREATE operation of order lines is not supported. The DISCONTINUE operation on order line will be sent to CZ (as below) without performing any operation on that order line.
 - UPDATE, DELETE, DISCONTINUE operations on CZ first class attributes is supported. REVERT operation on CZ is out of scope.
 - The only valid operation on the CZ extended attributes table is always UPDATE.
3. Updates order lines for Order Management attributes using the Process Order API. For example, Request Date may need to be updated in case of Disconnect operation.
4. API will call CZ API CZ_CF_API.Validate and pass the first class CZ attributes and extended CZ attributes along with the operations to that API so that CZ can take appropriate action.
 - Update on ordered quantity of an order line would result in implicit call to UPDATE the CZ first class attribute quantity.

- Similarly OM will implicitly treat a UPDATE on ordered quantity = 0 as DELETE and will follow the rules below.
- DELETE operation on order line will send DELETE operation to CZ if the component does not have a base line rev number if the component is new and is added to a completely new configuration or a re-configuration.
- In case of DELETE operation on order line with a baseline rev number > 0 , Order Management will error out. This is because CZ will not support REVERT action in this phase.

DISCONTINUE operation on an order line with a component with a base line rev number > 0 (real MACD re-configuration) will send DELETE to CZ.
DISCONTINUE operation on a MACD order line without base line rev will raise an Order Management error as you cannot discontinue a component that is not in Install Base.

- DISCONTINUE operation on a non-MACD order line should raise an error.

Setting Up the Process_MACD_Order API

The following chart describes all parameters used by the public Process_MACD_Order API. All of the inbound and outbound parameters are listed.

Parameter	Usage	Type	Required	Derived	Optional
P_API_VERSION_NUMBER	IN	Number	Yes		
P_org_id	IN	Number	Either org_id or operating unit is required		
P_operating_unit	IN	Varchar2	Either org_id or operating unit is required		

Parameter	Usage	Type	Required	Derived	Optional
P_sold_to_org_id	IN	Number	Either header_id or XXsold_to_org is required		
P_header_id	IN	Number	Either header_id or sold_to_org is required		
P_MACD_Action	IN	VARCHAR2	Null		
P_Instance_Tbl	IN	CSI_DATASTRUC TURES_PUB .IB_Instance_Tbl_Type	Null		
P_X_Line_Tbl	IN/OUT	OE_Order_Pub.Line_Tbl_Type	Null		
P_Extended_Attrib_Tbl	IN	OE_Order_Pub.Extended_Attribs_Tbl_Type	Null		
x_container_line_id	OUT	NUMBER			
X_RETURN_STATUS	OUT	Varchar2			
X_MSG_COUNT	OUT	Number			

Parameter	Usage	Type	Required	Derived	Optional
X_MSG_DATA	OUT	Varchar2			

Raising Business Events

The following custom events will be raised when you choose an action by clicking the Actions button on the Contact Center window:

- OM_PRE_ACTION_EVENT
- OM_POST_ACTION_EVENT

The following custom events will be raised when you click Book Orders on the Contact Center window:

- OM_PRE_BOOK_EVENT
- OM_PRE_BOOK_EVENT

Oracle Release Management XML Transactions

Overview of Oracle Release Management XML

XML in Oracle Release Management uses a combination of the following Oracle products:

- Oracle XML Gateway
- Oracle Advance Queuing
- Oracle Workflow / Business Event System (BES)
- Oracle Transportation Agent (OTA)

XML enables you to receive planning transactions from any trading partner that has the capability of producing and transmitting OAG/XML based transactions. The following sequence of actions are performed:

1. Receive XML planning transaction from various trading partners and origins.
2. Transactions are queued and processed by the XML Gateway Execution engine, which performs the following:
 - Validate the transaction: Transaction must be well formed and valid.
 - Derive trading partner and determine the map to be applied.
 - Apply code conversions, defaulting, and derivation logic on the data from the XML transaction to the Release Management interface tables.

All successfully processed transactions appear in the Release Management Workbench as Available to Process schedules. Similar to EDI or manual transactions, from this point onwards, all Release Management processes are executed on these transactions

The scope of the implementation is limited to Shipping Schedules with Actual quantities. The following elements are considered to be out of scope:

- SSLNEXCPTN segment
- DATETIME.RSPDOCGEN segment
- All elements pertaining to response schedules, such as DOCUMENTRV, RSPSCHEDID, RSPREVNUM, PSSTATUS, and RSPLINNUM

Receive message

The XML Gateway Execution Engine does the following during inbound processing:

- Dequeue Message from Inbound Queue
- Validate Message via XML Parser

Uses the XML Parser to validate the inbound message to determine if it is well-formed and valid (based on DTD stored in DTD directory) before proceeding further.

- Validate Trading Partner or Hub

If the inbound message is both well-formed and valid, the Execution Engine proceeds to validate that the Trading Partner and document are defined. If the Trading Partner is not defined or the document is not defined for the Trading Partner, the XML message will not be processed further.

- Get Message Map from Repository

If the message map associated with the Trading Partner is not available in the XML Gateway repository, the XML message will not be processed further.

- Maps Data

If the Trading Partner is valid and the message map exists in the repository, the Execution Engine maps the data in the XML message to its target data fields in Oracle e-Business Suite tables and columns identified in the message map. These are often the Application Open Interface tables. Mapping process also includes:

- Apply Code Conversion
 - Apply Actions
- Detect and Report Processing Errors

Errors may be detected by the Oracle XML Gateway Execution Engine, Oracle Advanced Queuing, Oracle Workflow, or Oracle Transport Agent. Information regarding the error is enqueued onto the Error Queue. An e-mail notification is sent via Oracle Workflow to notify the trading partner regarding data errors, or the XML

Gateway system administrator regarding system or process errors. In addition, for system or process errors, a copy of the XML message is placed in the XML message directory for use in troubleshooting the reported error. For trading partner-related data errors, the trading partner can refer to the copy of the XML message.

Planning Schedule Transaction

The SYNCH_PLANSCHD_001 transaction is similar to the EDIFACT/DELFOR or ANSI X12/830 transactions and is used to carry forecast or mixed firm/forecast requirements from the customer to the supplier. The purpose of the SYNC PLANSCHD Business Object Definition (BOD) is to communicate requirement information, for example part number and quantity, between a customer and their supplier on a regular basis.

SYNC PLANSCHD enables the addition of new requirements and the modifications of previously established requirements through various SYNC Indicator values. Existing SYNC Indicator values from OAG release 6.2 include "A" for add, "C" or change, and "D" for delete. In addition, a new SYNC indicator value "R" for replacement is defined for this PLANSCHD. Some of the elements contained on the XML include the following.

- Schedule Generation Date
- Horizon Dates
- Schedule Reference Number
- Schedule Purpose
- ECE TP Location Code
- Supplier Name
- Document Description
- Customer PO Number
- Customer Name
- Customer PO Date
- Requirement Date
- Item Quantity and UOM
- Customer Dock Code
- Item Description
- Customer Item Revision

- Item Detail Type
- PO Line Number
- PO Release Number
- Ship To/bill To Partner Information
- Ship From Partner Information

Sample Document Type Definition (DTD) and XML for Planning Schedule

The scope of the implementation is limited to Planning Schedules with Actual quantities. The following elements are considered to be out of scope (please refer to the sample DTD and XML):

- PSLNEXCPTN and FEATVALDAT segments
- DATETIME.RSPDOCGEN, DATETIME.DOCUMENT segments
- All elements pertaining to response schedules like DOCUMENTRV, RSPSCHEDID, RSPREVENUM, PSSTATUS, and RSPLINNUM

The following section includes a sample DTD and sample XML used for Planning Schedule in Oracle Release Management.

Planning Schedule Sample DTD

```
<!-- Structure Overview
      SYNC_PLANSCHD (PSHEADER, PSLINE+, PSFLEXBKT*)
      PSHEADER (PARTNER+)
      PSFLEXBKT()
      PSLINE (PARTNER+, FEATVALDAT*, PLNSCHED*, PSLNEXCPTN*)
      FEATVALDAT()
      PLNSCHED()
      PSLNEXCPTN()
      PSLNEXCPTN()
-->
<!-- ===== - >
<!ENTITY % RESOURCES SYSTEM "oagis_resources.dtd">%RESOURCES;
<!-- ===== - >

<!ELEMENT SYNC_PLANSCHD_001 (CNTROLAREA, DATAAREA+)>
<!ATTLIST VERB value CDATA #FIXED "SYNC">
<!ATTLIST NOUN value CDATA #FIXED "PLANSCHD">
<!ATTLIST REVISION value CDATA #FIXED "001">

<!ELEMENT DATAAREA (SYNC_PLANSCHD)>

<!ELEMENT SYNC_PLANSCHD (PSHEADER, PSLINE+, PSFLEXBKT*)>

<!ELEMENT PSHEADER (
  (%DATETIME.GENERATION;),
  (%DATETIME.SCHSTART;),
  (%DATETIME.SCHEND;),
  (%DATETIME.DOCUMENT;)?, (%DATETIME.RSPDOCGEN;)?,
  SCHEDULEID, SYNCIND, DATETYPE?,
  DESCRIPTN?, DOCUMENTRV?, NOTES?,
  POID?, PSSTATUS?, RSPSCHEDID?,
  RSPREVNUM?, SCHEDULETYP?,
  USERAREA?, PARTNER+, PSFLEXBKT*
)>

<!ELEMENT PSFLEXBKT (
  (%DATETIME.BKTEND;),
  (%DATETIME.BKTSTART;),
  FLEXBKTID, USERAREA?
)>

<!ELEMENT PSLINE (
  (%DATETIME.DELIVACT;)?, (%DATETIME.ENGCHG;)?,
  (%DATETIME.FROM;)?, (%DATETIME.PO;)?,
  (%DATETIME.RECEIVED;)?, (%DATETIME.SHIP;)?,
  (%DATETIME.TO;)?, (%QUANTITY.DELIVERED;)?,
  (%QUANTITY.ITEM;)?, (%QUANTITY.MINIMUM;)?,
  (%QUANTITY.MAXIMUM;)?, (%QUANTITY.PRIOR;)?,
  (%QUANTITY.RECEIVED;)?, (%QUANTITY.SHIPPED;)?,
  LINENUM, AUTHCODE?, DESCRIPTN?, DOCKID?,
  DRAWING?, HAZRDMATL?, ITEMRV?, ITEMRVX?,
  ITEMX?, LINETYPE?, NOTES?, PACKING?,
  PACKNGDESC?, POID?, POLINENUM?, PORELEASE?,
  PRIORITY?, PROJECT?, PSSTATUS?, RSPLINNUM?,
  SALESORDID?, SHIPPERNUM?, TASK?, UPC?, USERAREA?,
  ((COMMODITY, FEATUREID?, ITEM?) |
   (FEATUREID, COMMODITY?, ITEM?) |
   (ITEM, COMMODITY?, FEATUREID?)),
  )>
```

```

PARTNER+, FEATVALDAT*, PLNSCHED*, PSLNEXCPTN*
)>

<!ELEMENT FEATVALDAT (
  FEATUREID, ITMFTVAL, DESCRIPTN?,
  ITEMRV?, NAME*, PARTNRID?, UOM?, USERAREA?
)>

<!ELEMENT PLNSCHED (
  (%QUANTITY.ITEM;), (%DATETIME.FROM;),
  (%DATETIME.TO;)?, (%QUANTITY.MINIMUM;)?,
  (%QUANTITY.MAXIMUM;)?, (%QUANTITY.PRIOR;)?,
  PSDLINENUM, BKTYPE, AUTHCODE?,
  PRIORITY?, RSPSDLNUM?, USERAREA?, PSLNEXCPTN*
)>

<!ELEMENT PSLNEXCPTN (
  (%DATETIME.FROM;)?, (%DATETIME.TO;)?,
  ACTIONCODE?, DESCRIPTN?, EXCPLINNUM?,
  EXCPTNCODE?, REASONCODE?, USERAREA?
)>

```

Sample XML for Planning Schedule

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE SYNC_PLANSCHD_001 SYSTEM "162_sync_planschd_001.dtd">
<SYNC_PLANSCHD_001>
  <CNTROLAREA>
    <BSR>
      <VERB value="SYNC">SYNC</VERB>
      <NOUN value="PLANSCHD">PLANSCHD</NOUN>
      <REVISION value="001">001</REVISION>
    </BSR>
  <SENDER>
    <LOGICALID>XX141HG09</LOGICALID>
    <COMPONENT>PLANSCHD</COMPONENT>
    <TASK>SYNC</TASK>
    <REFERENCEID>1233</REFERENCEID>
    <CONFIRMATION>1</CONFIRMATION>
    <LANGUAGE>ENG</LANGUAGE>
    <CODEPAGE>0000</CODEPAGE>
    <AUTHID>2817</AUTHID>
  </SENDER>
  <DATETIME qualifier="CREATION">
    <YEAR>2000</YEAR>
    <MONTH>10</MONTH>
    <DAY>15</DAY>
    <HOUR>12</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>59</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>-0500</TIMEZONE>
  </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SYNC_PLANSCHD>
      <PSHEADER>
        <DATETIME qualifier="GENERATION">
          <YEAR>2000</YEAR>
          <MONTH>11</MONTH>
          <DAY>16</DAY>
          <HOUR>12</HOUR>
          <MINUTE>59</MINUTE>
          <SECOND>59</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>-0500</TIMEZONE>
        </DATETIME>
        <DATETIME qualifier="SCHSTART">
          <YEAR>2000</YEAR>
          <MONTH>12</MONTH>
          <DAY>17</DAY>
          <HOUR>12</HOUR>
          <MINUTE>59</MINUTE>
          <SECOND>59</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>-0500</TIMEZONE>
        </DATETIME>
        <DATETIME qualifier="SCHEND">
          <YEAR>2000</YEAR>
          <MONTH>01</MONTH>
          <DAY>18</DAY>
          <HOUR>12</HOUR>
```

```

<MINUTE>59</MINUTE>
<SECOND>59</SECOND>
<SUBSECOND>0000</SUBSECOND>
<TIMEZONE>-0500</TIMEZONE>
</DATETIME>
<SCHEDULEID>XML-PLN-01</SCHEDULEID>
<SYNCIND>Replace</SYNCIND>
  <DATETYPE>Date-type</DATETYPE>
  <DESCRIPTN>document-desk</DESCRIPTN>
  <NOTES>header-notes</NOTES>
  <PSSTATUS>5</PSSTATUS>
<SCHDULETYP>DEMAND</SCHDULETYP>
<PARTNER>
<NAME index="1">Modern Truck</NAME>
<ONETIME>0</ONETIME>
<PARTNRID>MCL</PARTNRID>
<PARTNRTYPE>SoldTo</PARTNRTYPE>
<CURRENCY>USD</CURRENCY>
</PARTNER>
<PARTNER>
<NAME index="1">Federal Mogul</NAME>
<ONETIME>0</ONETIME>
<PARTNRID>FM</PARTNRID>
<PARTNRTYPE>Supplier</PARTNRTYPE>
<CURRENCY>USD</CURRENCY>
</PARTNER>
</PSHEADER>
<PSLINE>
  <LINENUM>1</LINENUM>
  <AUTHCODE>Auth-code</AUTHCODE>
  <ITEMRV>itemrv</ITEMRV>
  <ITEMRVX>rvx</ITEMRVX>
  <ITEMX>itemx</ITEMX>
  <LINETYPE>Firm</LINETYPE>
  <NOTES>notes</NOTES>
  <PACKING>packcd</PACKING>
  <PACKNGDESC>packingdesc</PACKNGDESC>
  <POID>po-id</POID>
  <POLINENUM>po-lnum</POLINENUM>
  <PORELEASE>po-rel</PORELEASE>
  <PROJECT>proj</PROJECT>
  <PSSTATUS>2</PSSTATUS>
  <RSPLINNUM>rsplinenum</RSPLINNUM>
  <SALESORDID>salesordid</SALESORDID>
  <SHIPPERNUM>10023</SHIPPERNUM>
  <TASK>task</TASK>
  <UPC>upc</UPC>
  <COMMODITY index="1">CMDTY1</COMMODITY>
  <FEATUREID>feat</FEATUREID>
  <ITEM>item12</ITEM>
<PARTNER>
<NAME index="1">Federal Parts</NAME>
<ONETIME>0</ONETIME>
  <PARTNRID>FP</PARTNRID>
  <PARTNRTYPE>ShipTo</PARTNRTYPE>
<CURRENCY>USD</CURRENCY>
</PARTNER>
<PARTNER>
<NAME index="1">Federal Parts II</NAME>
<ONETIME>0</ONETIME>
<PARTNRID>M2</PARTNRID>

```

```

<PARTNRTYPE>ShipFrom</PARTNRTYPE>
<CURRENCY>USD</CURRENCY>
</PARTNER>
<PLNSCHED>
  <QUANTITY qualifier="ITEM">
    <VALUE>3</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>EACH</UOM>
  </QUANTITY>
  <DATETIME qualifier="FROM">
    <YEAR>2401</YEAR>
    <MONTH>02</MONTH>
    <DAY>23</DAY>
    <HOUR>12</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>59</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>-0500</TIMEZONE>
  </DATETIME>
  <PSDLINENUM>linenum2</PSDLINENUM>
  <BKTYPE>Day</BKTYPE>
  </PLNSCHED>
</PSLINE>
<PSFLEXBKT>
  <DATETIME qualifier="BKTEND">
    <YEAR>2002</YEAR>
    <MONTH>03</MONTH>
    <DAY>17</DAY>
    <HOUR>12</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>59</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>-0500</TIMEZONE>
  </DATETIME>
  <DATETIME qualifier="BKTSTART">
    <YEAR>2002</YEAR>
    <MONTH>03</MONTH>
    <DAY>15</DAY>
    <HOUR>12</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>59</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>-0500</TIMEZONE>
  </DATETIME>
  <FLEXBKTID>Bi-monthly</FLEXBKTID>
</PSFLEXBKT>
</SYNC_PLANSCHD>
</DATAAREA>
</SYNC_PLANSCHD_001>

```

Planning Schedule XML to Database Mapping

The Planning Schedule contains information that is mapped to columns in the RLM_INTERFACE_HEADERS_ALL and RLM_INTERFACE_LINES_ALL tables. A one-to-one relation between the XML elements and the underlying database columns is outlined below. Also included is a list of defaulting rules and additional validations that are performed.

XML Element/Segment	RLM_INTERFACE_HEADERS_ALL Column
DATETIME.GENERATION	SCHED_GENERATION_DATE
DATETIME.SCHSTART	SCHED_HORIZON_START_DATE
DATETIME.SCHEND	SCHED_HORIZON_END_DATE
DATETIME.DOCUMENT	Not Mapped
DATETIME.RSPDOCGEN	Not Mapped
SCHEDULEID	SCHEDULE_REFERENCE_NUM
SYNCIND	SCHEDULE_PURPOSE_EXT
DATETYPE	Line Level (DATE_TYPE_CODE_EXT)
DESCRIPTN	DOCUMENT_DESCRIPTN
DOCUMENTRV	Not Mapped
NOTES	HEADER_NOTES_EXT
POID	Line Level - CUST_PO_NUMBER
PSSTATUS	Not Mapped
RSPSCHEDID	Not Mapped
RSPREVNUM	Not Mapped
SCHDULETYP	Not Mapped
USERAREA	Not Mapped
PARTNER	
NAME	CUST_NAME_EXT
PARTNRID	ONE_TIME_CUST_FLAG_EXT

XML Element/Segment	RLM_INTERFACE_HEADERS_ALL Column
PARTNRTYPE	Expected Value SoldTo
PARTNER	
NAME	SUPPLIER_NAME_EXT
ONETIME	ONE_TIME_SUPPLIER_FLAG_EXT
PARTNRID	
PARTNRTYPE	Expected Value Supplier

Defaults applied to RLM_INTERFACE_HEADERS_ALL

- PROCESS_STATUS is set to 2, indicating that the schedule is Available to Process.
- Standard WHO columns are populated in the message map.
- HEADER_ID is generated by a sequence
- EDI_TEST_INDICATOR is set to 'P'.
- SCHEDULE_SOURCE is set to 'XML'.
- SCHEDULE_TYPE is set to 'PLANNING_RELEASE'.
- SCHEDULE_TYPE_EXT is defaulted to 'PLANSCHD'.

A value of DEMAND is expected for the element SCHDULETYP. If this element contains any other value, then the XML message is rejected. At least two PARTNER segments are expected at the header level, with PARTNRTYPE values of SoldTo and Supplier. This information is used to identify the trading partners.

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
DATETIME.DELIVACT	Not Mapped
DATETIME.ENGCHG	ITEM_ENG_CNG_LVL_DATE_EXT
DATETIME.FROM	START_DATE_TIME (Depends on LINETYPE)

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
DATETIME.PO	CUST_PO_DATE
DATETIME.RECEIVED	START_DATE_TIME (Depends on LINETYPE)
DATETIME.SHIP	START_DATE_TIME (Depends on LINETYPE)
DATETIME.TO	START_DATE_TIME (Depends on LINETYPE)
QUANTITY.DELIVERED	Not Mapped
QUANTITY.ITEM	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)
QUANTITY.MINIMUM	ITEM_DETAIL_QUANTITY_MIN_EXT
QUANTITY.MAXIMUM	ITEM_DETAIL_QUANTITY_MAX_EXT
QUANTITY.PRIOR	ITEM_DETAIL_QUANTITY_PRIOR_EXT
QUANTITY.RECEIVED	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)
QUANTITY.SHIPPED	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)
LINENUM	Not Mapped
AUTHCODE	AUTHORIZATION_CODE_EXT
DESCRIPTN	ITEM_DESCRIPTION_EXT
DOCKID	CUSTOMER_DOCK_CODE
DRAWING	Not Mapped
HAZRDMATL	HAZARD_CODE_EXT
ITEMRV	Not Mapped

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
ITEMRVX	CUSTOMER_ITEM_REVISION
ITEMX	CUSTOMER_ITEM_EXT
LINETYPE	ITEM_DETAIL_TYPE_EXT
NOTES	ITEM_NOTE_EXT
PACKING	PACKAGING_CODE_EXT
PACKNGDESC	PACKAGING_DESC_EXT
POID	CUST_PO_NUMBER
POLINENUM	CUST_PO_LINE_NUM
PORELEASE	CUST_PO_RELEASE_NUM
PRIORITY	REQUIREMENT_PRIORITY_EXT
PROJECT	PROJECT_NUMBER_EXT
PSSTATUS	Not Mapped
RSPLINNUM	Not Mapped
SALESORDID	Not Mapped
SHIPPERNUM	ITEM_DETAIL_REF_VALUE_1
TASK	TASK_NUMBER_EXT
UPC	Not Mapped
USERAREA	Not Mapped
COMMODITY	COMMODITY_EXT
FEATUREID	Not Mapped

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
ITEM	SUPPLIER_ITEM_EXT
PARTNER	
NAME	SHIP_TO_NAME_EXT
ONETIME	ONE_TIME_SHIP_TO_FLAG_EXT
PARTNRID	CUST_SHIP_TO_EXT
PARTNRTYPE	Expected Value ShipTo
PARTNER	
NAME	BILL_TO_NAME_EXT
ONETIME	ONE_TIME_BILL_TO_FLAG_EXT
PARTNRID	CUST_BILL_TO_EXT
PARTNRTYPE	Expected Value BillTo
PARTNER	
NAME	SHIP_FROM_NAME_EXT
ONETIME	ONE_TIME_SHIP_FROM_FLAG_EXT
PARTNRID	CUST_SHIP_FROM_ORG_EXT
PARTNRTYPE	Expected Value ShipFrom
PLNSCHED Segment	
QUANTITY.ITEM	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)
DATETIME.FROM	START_DATE_TIME (Depends on LINETYPE)

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
DATETIME.TO	END_DATE_TIME
QUANTITY.MINIMUM	ITEM_DETAIL_QUANTITY_MIN_EXT
QUANTITY.MAXIMUM	ITEM_DETAIL_QUANTITY_MAX_EXT
QUANTITY.PRIOR	ITEM_DETAIL_QUANTITY_PRIOR_EXT
PSDLINENUM	Not Mapped
BKTYPE	Determines the value of ITEM_DETAIL_SUBTYPE_EXT
AUTHCODE	AUTHORIZATION_CODE_EXT
PRIORITY	REQUIREMENT_PRIORITY_EXT
RSPSDLNUM	Not Mapped
USERAREA	Not Mapped
PSFLEXBKT	
DATETIME.BKTEND	END_DATE_TIME (Based on BKTYPE/FLEXBKTID)
DATETIME.BKTSTART	START_DATE_TIME (Based on BKTYPE/FLEXBKTID)
FLEXBKTID	FLEX_BKT_CODE

Defaults applied to RLM_INTERFACE_LINES_ALL

- PROCESS_STATUS is defaulted to 2.
- LINE_SOURCE is set to XML.
- HEADER_ID is copied from RLM_INTERFACE_HEADERS_ALL
- Standard WHO columns are assigned values in the message map.
- QTY_TYPE_CODE_EXT and QTY_TYPE_CODE are set to ACTUAL.

- LINE_ID is generated by a sequence.

Conditional Mapping

1. Depending on the element LINETYPE, determine the line level information as follows:

IF LINETYPE = 'PAST DUE'/'FIRM'/'FORECAST' or

LINETYPE is NULL /* not transmitted *//THEN

- ITEM_DETAIL_TYPE_EXT: = LINETYPE/'Forecast'
- ITEM_DETAIL _SUBTYPE_EXT: = NVL (BKTYPE, 'Day')
- DATE_TYPE_CODE_EXT: = NVL (DATETYPE, 'Deliver')
- START_DATE_TIME: = NVL (DATETIME.FROM, SYSDATE);
- END_DATE_TIME: = DATETIME.TO
- ITEM_DETAIL_QUANTITY: = QUANTITY.ITEM
- CUST_UOM_EXT: = QUANTITY.ITEM.UOM

ELSIF LINETYPE = 'SHIPPED/RECEIVED' THEN

- ITEM_DETAIL_TYPE_EXT: = LINETYPE
- IF DATETIME.RECEIVED IS NOT NULL THEN
 - ITEM_DETAIL _SUBTYPE_EXT: = 'RECEIPT'
 - DATE_TYPE_CODE_EXT: = 'RECEIVED'
 - START_DATE_TIME: = DATETIME.RECEIVED
 - ITEM_DETAIL_QUANTITY: = QUANTITY.RECEIVED
 - CUST_UOM_EXT: = QUANTITY.RECEIVED.UOM

ELSIF DATETIME.SHIPPED IS NOT NULL THEN

- ITEM_DETAIL _SUBTYPE_EXT: = 'SHIPMENT'
- DATE_TYPE_CODE_EXT: = 'SHIPPED'
- START_DATE_TIME: = DATETIME.SHIPPED
- ITEM_DETAIL_QUANTITY: = QUANTITY.SHIPPED

- CUST_UOM_EXT: = QUANTITY.SHIPPED.UOM

END IF;

ELSIF LINETYPE = 'AUTHORIZATION' THEN

- ITEM_DETAIL_TYPE_EXT:= LINETYPE
- ITEM_DETAIL_SUBTYPE_EXT: = NVL(BKTYPE, 'FINISHED')
- DATE_TYPE_CODE_EXT: = 'FROM_TO'
- START_DATE_TIME: = DATETIME.REQUIRED
- ITEM_DETAIL_QUANTITY: = QUANTITY.ITEM
- CUST_UOM_EXT: = QUANTITY.ITEM.UOM

ELSIF LINETYPE = 'OTHER' THEN

- ITEM_DETAIL_TYPE_EXT:= 'OTHER'
- ITEM_DETAIL_SUBTYPE_EXT: = NVL(BKTYPE, 'AHEAD_BEHIND')
- DATE_TYPE_CODE_EXT: = 'AS_OF' · START_DATE_TIME: = DATETIME.REQUIRED
- ITEM_DETAIL_QUANTITY: = QUANTITY.ITEM
- CUST_UOM_EXT: = QUANTITY.ITEM.UOM

END IF

2. The start and end date times on the requirement line are also dependent on the BKTYPE that is in on the PLNSCHED level. If the BKTYPE matches any of the FLEXBKTID values on the PSFLEXBKT segments, then the DATETIME.BKTEND and DATETIME.BKTSTART constitute the end and start date times respectively for the requirement line.
3. If the POID on the PSLINE level is null, then the value of POID from PSHEADER is used.
4. If ShipFrom Partner information is not present at the line level, then the Supplier details from the header are copied to the line level.

Derivation of schedule_item_num

SCHEDULE_ITEM_NUM is needed to identify a group for the purpose of validating the line level data. This value is derived based on the combination of the following attributes from the XML message:

- Ship from
- Ship to
- Bill to
- Customer item
- Item description
- Dock code
- Hazard code
- Item revision
- Item notes
- Customer PO number
- Customer PO line number
- Customer PO release number
- Customer PO date
- Supplier item

After inserting the data in the database, a post process API groups the lines using the above attributes and for each group, unique values are assigned to schedule_item_num.

Other issues/considerations

Data that comes in at the PLNSCHED level overrides the data on PSLINE level.

Shipping Schedule Transaction

The SYNC_SHIPSCHD_001 transaction is similar to the EDIFACT/DELJIT or ANSI X12/862 transactions and is used to carry firm requirements with eventual related JIT information (KANBAN) from the customer to the supplier.

The purpose of the SYNC SHIPSCHD BOD is to enable the exchange of shipment schedule information, authorizing a shipment quantity and date for specific trading partners and addresses.

The shipping schedule is commonly generated by a material planning application and transmitted to an order or material planning application.

Some of the elements contained on the XML include the following.

- Schedule Generation Date
- Horizon Dates
- Schedule Reference Number
- Schedule Purpose
- ECE TP Location Code
- Supplier Name
- Document Description
- Customer PO Number
- Customer Name
- Customer PO Date
- Requirement Date
- Item Quantity and UOM
- Customer Dock Code
- Item Description
- Customer Item Revision
- Item Detail Type
- PO Line Number
- PO Release Number
- Ship To/bill To Partner Information
- Ship From Partner Information

Sample Document Type Definition (DTD) and XML for Shipment Schedule

```
<!-- Structure Overview
      SYNC_SHIPSCHD (SHIPSCHDHD, SHIPSCHDLN+)
      SHIPSCHDHD (PARTNER+)
      SHIPSCHDLN (PARTNER+, SSLNEXCPTN*)  - >
<!-- ===== - >
<!ENTITY % RESOURCES SYSTEM "oagis_resources.dtd">%RESOURCES;
<!-- ===== - >

<!ELEMENT SYNC_SHIPSCHD_001 (CNTROLAREA, DATAAREA+)>
<!ATTLIST VERB value CDATA #FIXED "SYNC">
<!ATTLIST NOUN value CDATA #FIXED "SHIPSCHD">
<!ATTLIST REVISION value CDATA #FIXED "001">

<!ELEMENT DATAAREA (SYNC_SHIPSCHD)>

<!ELEMENT SYNC_SHIPSCHD (SHIPSCHDHD, SHIPSCHDLN+)>

<!ELEMENT SHIPSCHDHD (
  (%DATETIME.GENERATION;),
  (%DATETIME.SCHEND;),
  (%DATETIME.SCHSTART;),
  (%DATETIME.DOCUMENT;)?,
  (%DATETIME.RSPDOCGEN;)?,
  SCHEDULEID, SYNCIND,
  DATETYPE?, DESCRIPTN?,
  DOCUMENTRV?, NOTES?, PSSTATUS?,
  RSPSCHEDID?, RSPREVNUM?,
  SCHEDULETYP?, USERAREA?, PARTNER+
)>

<!ELEMENT SHIPSCHDLN (
  (%DATETIME.REQUIRED;), (%QUANTITY.ITEM;),
  (%DATETIME.CUMULATIVE;)?, (%DATETIME.DELIVACT;)?,
  (%DATETIME.ENGCHG;)?, (%DATETIME.PO;)?,
  (%DATETIME.RECEIVED;)?, (%DATETIME.SHIP;)?,
  (%QUANTITY.CUMULATIVE;)?, (%QUANTITY.DELIVERED;)?,
  (%QUANTITY.PRIOR;)?, (%QUANTITY.RECEIVED;)?,
  (%QUANTITY.SHIPPED;)?,
  ITEM, LINENUM, AUTHCODE?,
  DESCRIPTN?, DOCKID?, DRAWING?,
  HAZRDMATL?, ITEMRV?, ITEMRVX?,
  ITEMX?, KANBAN?, LINETYPE?, NOTES?,
  PACKING?, PACKNGDESC?, POID?, POLINENUM?,
  PORELEASE?, PRIORITY?, PROJECT?, RSPLINNUM?,
  SALESORDID?, SHIPPERNUM?, TASK?, UPC?,
  USERAREA?, PARTNER+, SSLNEXCPTN*
)>

<!ELEMENT SSLNEXCPTN (
  (%DATETIME.FROM;)?, (%DATETIME.TO;)?,
  ACTIONCODE?, DESCRIPTN?, EXCPLINNUM?,
  EXCPTNCODE?, REASONCODE?, USERAREA?
)>
```


Sample XML for Shipment Schedule

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE SYNC_SHIPSCHD_001 SYSTEM "168_sync_shipschd_001.dtd">
<SYNC_SHIPSCHD_001>
  <CNTROLAREA>
    <BSR>
      <VERB value="SYNC">SYNC</VERB>
      <NOUN value="SHIPSCHD">SHIPSCHD</NOUN>
      <REVISION value="001">001</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID>XX141HG09</LOGICALID>
      <COMPONENT>SHIPSCHD</COMPONENT>
      <TASK>SYNC</TASK>
      <REFERENCEID>1233</REFERENCEID>
      <CONFIRMATION>1</CONFIRMATION>
      <LANGUAGE>ENG</LANGUAGE>
      <CODEPAGE>0000</CODEPAGE>
      <AUTHID>8821</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2000</YEAR>
      <MONTH>10</MONTH>
      <DAY>15</DAY>
      <HOUR>12</HOUR>
      <MINUTE>59</MINUTE>
      <SECOND>59</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>-0500</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SYNC_SHIPSCHD>
      <SHIPSCHDHD>
        <DATETIME qualifier="GENERATION">
          <YEAR>2002</YEAR>
          <MONTH>01</MONTH>
          <DAY>01</DAY>
          <HOUR>12</HOUR>
          <MINUTE>59</MINUTE>
          <SECOND>59</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>-0500</TIMEZONE>
        </DATETIME>
        <DATETIME qualifier="SCHEND">
          <YEAR>2002</YEAR>
          <MONTH>01</MONTH>
          <DAY>31</DAY>
          <HOUR>12</HOUR>
          <MINUTE>59</MINUTE>
          <SECOND>59</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>-0500</TIMEZONE>
        </DATETIME>
        <DATETIME qualifier="SCHSTART">
          <YEAR>2002</YEAR>
          <MONTH>01</MONTH>
          <DAY>01</DAY>
          <HOUR>12</HOUR>
```

```

<MINUTE>59</MINUTE>
  <SECOND>59</SECOND>
  <SUBSECOND>0000</SUBSECOND>
  <TIMEZONE>-0500</TIMEZONE>
</DATETIME>
      <SCHEDULEID>XML-SHP-01</SCHEDULEID>
<SYNCIND>REPLACE</SYNCIND>
<DESCRIPTN>xmldesc</DESCRIPTN>
<NOTES>header notes</NOTES>
<SCHDULETYP>DEMAND</SCHDULETYP>      <PARTNER>
  <NAME index="1">Modern Truck</NAME>
  <ONETIME>0</ONETIME>
  <PARTNRID>MCL</PARTNRID>
    <PARTNRRTYPE>SoldTo</PARTNRRTYPE>      </PARTNER>
<PARTNER>
  <NAME index="1">Federal Parts</NAME>
  <ONETIME>0</ONETIME>
  <PARTNRID>A987123</PARTNRID>
    <PARTNRRTYPE>Supplier</PARTNRRTYPE>      </PARTNER>
</SHIPSCHDHD>
<SHIPSCHDLN>
  <DATETIME qualifier="REQUIRED">
    <YEAR>2001</YEAR>
    <MONTH>11</MONTH>
    <DAY>15</DAY>
    <HOUR>12</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>59</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>-0500</TIMEZONE>
  </DATETIME>
  <QUANTITY qualifier="ITEM">
    <VALUE>200</VALUE>
    <NUMOFDEC>0</NUMOFDEC>
    <SIGN>+</SIGN>
    <UOM>EA</UOM>
  </QUANTITY>
  <ITEM>MT100</ITEM>
  <LINENUM>1</LINENUM>
  <AUTHCODE>Auth-code</AUTHCODE>
  <DESCRIPTN>Standard Mount</DESCRIPTN>
  <DOCKID>Dock-code</DOCKID>
  <DRAWING>Drawing</DRAWING>
  <HAZRDMATL>hazard</HAZRDMATL>
  <ITEMRV>item-rev</ITEMRV>
  <ITEMRVX>EXT</ITEMRVX>
  <ITEMX>CM99223</ITEMX>
  <KANBAN>kanban</KANBAN>
  <LINETYPE>Firm</LINETYPE>
  <NOTES>line-notes</NOTES>
  <PACKING>packingcd</PACKING>
  <PACKNGDESC>packing-desc</PACKNGDESC>
  <POID>po-id</POID>
  <POLINENUM>po-linenum</POLINENUM>
  <PORELEASE>PO-release</PORELEASE>
  <PRIORITY>2</PRIORITY>
  <PROJECT>proj</PROJECT>
  <SALESORDID>T</SALESORDID>
  <SHIPPERNUM>10023</SHIPPERNUM>
  <TASK>task</TASK>
  <UPC>UPC</UPC>

```

```

<PARTNER>
  <NAME index="1">Boston Manu</NAME>
  <ONETIME>0</ONETIME>
  <PARTNRID>M2</PARTNRID>
  <PARTNRTYPE>ShipFrom</PARTNRTYPE>
  <CURRENCY>USD</CURRENCY>
</PARTNER>
<PARTNER>
  <NAME index="1">ACME2</NAME>
  <ONETIME>0</ONETIME>
  <PARTNRID>1500</PARTNRID>
  <PARTNRTYPE>ShipTo</PARTNRTYPE>
  <CURRENCY>USD</CURRENCY>
</PARTNER>
</SHIPSCHDLN>
</SYNC_SHIPSCHD>
</DATAAREA>
</SYNC_SHIPSCHD_001>

```

Shipping Schedule XML to Database Mapping

A one-to-one relation between the XML elements and the underlying database columns is outlined below:

XML Element/Segment	RLM_INTERFACE_HEADERS_ALL Column
DATETIME.GENERATION	SCHED_GENERATION_DATE
DATETIME.SCHEND	SCHED_HORIZON_END_DATE
DATETIME.SCHSTART	SCHED_HORIZON_START_DATE
DATETIME.DOCUMENT	Not Mapped
DATETIME.RSPDOCGEN	Not Mapped
SCHEDULEID	SCHEDULE_REFERENCE_NUM
SYNCIND	SCHEDULE_PURPOSE_EXT
DATETYPE	Line Level (DATE_TYPE_CODE_EXT)
DESCRIPTN	DOCUMENT_DESCRIPTION
DOCUMENTRV	Not Mapped
NOTES	HEADER_NOTES_TEXT

XML Element/Segment	RLM_INTERFACE_HEADERS_ALL Column
PSSTATUS	Not Mapped
PSSTATUS	Not Mapped
RSPSCHEDID	Not Mapped
RSPREVNUM	Not Mapped
SCHDULETYP	Not Mapped
USERAREA	Not Mapped
PARTNER	
NAME	CUST_NAME_EXT
ONETIME	ONE_TIME_CUST_FLAG_EXT
PARTNRID	ECE_TP_LOCATION_CODE_EXT andCUSTOMER_EXT
PARTNRTYPE	Expected Value SoldTo
PARTNER	
NAME	SUPPLIER_NAME_EXT
ONETIME	ONE_TIME_SUPPLIER_FLAG_EXT
PARTNRID	
PARTNRTYPE	Expected Value Supplier

Defaults applied to RLM_INTERFACE_HEADERS_ALL

- PROCESS_STATUS is set to 2, indicating that the schedule is Available to Process.
- Standard WHO columns are populated in the message map.
- HEADER_ID is generated as a sequence

- EDI_TEST_INDICATOR is set to 'P', to indicate production schedules.
- SCHEDULE_SOURCE is set to 'XML'.
- SCHEDULE_TYPE is set to 'SHIPPING'.
- SCHEDULE_TYPE_EXT is defaulted to 'SHIPSCHD'.

A value of DEMAND is expected for the element SCHDULETYP. If this element contains any other value, the XML message is rejected. At least two PARTNER segments are expected at the header level, with PARTNRTYPE values of SoldTo and Supplier. This information is used to identify the trading partners.

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
DATETIME.REQUIRED	START_DATE_TIME (Depends on LINETYPE)
QUANTITY.ITEM	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)
DATETIME.CUMULATIVE	Not Mapped
DATETIME.DELIVACT	Not Mapped
DATETIME.ENGCHG	ITEM_ENG_CNG_LVL_DATE_EXT
DATETIME.PO	CUST_PO_DATE
DATETIME.RECEIVED	START_DATE_TIME (Depends on LINETYPE)
DATETIME.SHIP	START_DATE_TIME (Depends on LINETYPE)
QUANTITY.CUMULATIVE	Not Mapped
QUANTITY.DELIVERED	Not Mapped
QUANTITY.PRIOR	ITEM_DETAIL_QUANTITY_PRIOR_EXT
QUANTITY.RECEIVED	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
QUANTITY.SHIPPED	ITEM_DETAIL_QUANTITY (Depends on LINETYPE)
ITEM	SUPPLIER_ITEM_EXT
LINENUM	LINE_NUMBER
AUTHCODE	AUTHORIZATION_CODE_EXT
DESCRIPTN	ITEM_DESCRIPTION_EXT
DOCKID	CUSTOMER_DOCK_CODE
DRAWING	Not Mapped
HAZRDMATL	HAZARD_CODE_EXT
ITEMRV	Not Mapped
ITEMRVX	CUSTOMER_ITEM_REVISION
ITEMX	CUSTOMER_ITEM_EXT
KANBAN	INDUSTRY_ATTRIBUTE4
LINETYPE	ITEM_DETAIL_TYPE_EXT
NOTES	ITEM_NOTE_TEXT
PACKING	PACKAGING_CODE_EXT
PACKNGDESC	PACKAGING_DESC_EXT
POID	CUST_PO_NUMBER
POLINENUM	CUST_PO_LINE_NUM
PORELEASE	CUST_PO_RELEASE_NUM
PRIORITY	REQUIREMENT_PRIORITY_EXT

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
PROJECT	PROJECT_NUMBER_EXT
RSPLINNUM	Not Mapped
SALESORDID	Not Mapped
SHIPPERNUM	ITEM_DETAIL_REF_VALUE_1
TASK	TASK_NUMBER_EXT
UPC	Not Mapped
USERAREA	Not Mapped
PARTNER	
NAME	BILL_TO_NAME_EXT
ONETIME	ONE_TIME_BILL_TO_FLAG_EXT
PARTNRID	CUST_BILL_TO_EXT
PARTNRTYPE	Expected Value BillTo
PARTNER	
NAME	SHIP_FROM_NAME_EXT
ONETIME	ONE_TIME_SHIP_FROM_FLAG_EXT
PARTNRID	CUST_SHIP_FROM_ORG_EXT
PARTNRTYPE	Expected Value ShipFrom
PARTNER	
NAME	SHIP_TO_NAME_EXT
ONETIME	ONE_TIME_SHIP_TO_FLAG_EXT

XML Element/Segment	RLM_INTERFACE_LINES_ALL Column
PARTNRID	CUST_SHIP_TO_EXT
PARTNRTYPE	Expected Value ShipTo

Defaults applied to RLM_INTERFACE_LINES_ALL

- PROCESS_STATUS is defaulted to 2.
- LINE_SOURCE is set to XML.
- HEADER_ID is copied from RLM_INTERFACE_HEADERS_ALL
- Standard WHO columns are assigned values in the message map.
- QTY_TYPE_CODE_EXT and QTY_TYPE_CODE are set to 'ACTUAL'.
- LINE_ID is generated by a sequence.

Conditional Mapping

1. Depending on the element LINETYPE, determine the line level information as follows:

IF LINETYPE = 'PAST DUE' / 'FIRM'/'FORECAST' OR
LINETYPE is NULL /* not transmitted */ THEN

- ITEM_DETAIL_TYPE_EXT: = LINETYPE/'Firm'
- ITEM_DETAIL_SUBTYPE_EXT: = 'Day'
- START_DATE_TIME: = DATETIME.REQUIRED
- ITEM_DETAIL_QUANTITY: = QUANTITY.ITEM
- DATE_TYPE_CODE_EXT: = NVL (DATETYPE, 'Deliver')
- CUST_UOM_EXT: = QUANTITY.ITEM.UOM

ELSIF LINETYPE='SHIPPED/RECEIVED' THEN

- ITEM_DETAIL_TYPE_EXT: = LINETYPE
- IF DATETIME.RECEIVED IS NOT NULL THEN
 - ITEM_DETAIL_SUBTYPE_EXT: = 'RECEIPT'

- DATE_TYPE_CODE_EXT: = 'RECEIVED'
- START_DATE_TIME: = DATETIME.RECEIVED
- ITEM_DETAIL_QUANTITY: = QUANTITY.RECEIVED
- CUST_UOM_EXT: = QUANTITY.RECEIVED.UOM

ELSIF DATETIME.SHIPPED IS NOT NULL THEN

- ITEM_DETAIL_SUBTYPE_EXT: = 'SHIPMENT'
- DATE_TYPE_CODE_EXT: = 'SHIPPED'
- START_DATE_TIME: = DATETIME.SHIPPED
- ITEM_DETAIL_QUANTITY: = QUANTITY.SHIPPED
- CUST_UOM_EXT: = QUANTITY.SHIPPED.UOM

END IF;

ELSIF LINETYPE = 'AUTHORIZATION' THEN

- ITEM_DETAIL_TYPE_EXT:= LINETYPE
- ITEM_DETAIL_SUBTYPE_EXT: = 'FINISHED'
- DATE_TYPE_CODE_EXT: = 'FROM_TO'
- START_DATE_TIME: = DATETIME.REQUIRED
- ITEM_DETAIL_QUANTITY: = QUANTITY.ITEM
- CUST_UOM_EXT: = QUANTITY.ITEM.UOM

ELSIF LINETYPE = 'OTHER' THEN

- ITEM_DETAIL_TYPE_EXT:= 'OTHER'
- ITEM_DETAIL_SUBTYPE_EXT: = 'AHEAD_BEHIND'
- DATE_TYPE_CODE_EXT: = 'AS_OF'
- START_DATE_TIME: = DATETIME.REQUIRED
- ITEM_DETAIL_QUANTITY: = QUANTITY.ITEM
- CUST_UOM_EXT: = QUANTITY.ITEM.UOM

END IF

2. If ShipFrom Partner information doesn't come in at the line level, copy the Supplier Partner data from the header to the lines level.

Derivation of schedule_item_num

SCHEDULE_ITEM_NUM is required to identify a group for the purpose of validating the line level data. The Shipping Schedule has only two levels and there is no corresponding element on the XML message to uniquely identify a group. Hence, this value is derived based on the combination of the following attributes from the XML message:

- Ship from
- Ship to
- Bill to
- Customer item
- Item description
- Dock code
- Hazard code
- Item revision
- Item notes
- Customer PO number
- Customer PO line number
- Customer PO release number
- Customer PO date
- Supplier item (from the XML message)

After inserting data in the database, a post process API groups the lines using the above attributes and for each group, a unique values is assigned to schedule_item_num.

Oracle Release Management EDI Transactions

This chapter covers the following topics:

- Production Sequence EDI
- Shipping Schedule EDI
- Planning Schedule EDI
- Purpose Codes
- Transaction Descriptions

Production Sequence EDI

Production Sequence is used to import customer production sequence schedules into Oracle Release Management, to provide a means to request the order in which the shipment of goods arrives, and to specify the order in which the goods are unloaded, enabling the sequence in which the goods enter the production process and are consumed. The Demand Processor imports customer production sequence schedules to Oracle Order Management and Oracle Planning.

Production Sequence is initiated from the Release Management or e-Commerce Gateway responsibility.

Shipping Schedule EDI

The shipping schedule transaction set can be used by a customer to convey precise shipping schedule requirements to a supplier, and is intended to supplement the planning schedule transaction set (830). The shipping schedule transaction set will supersede certain shipping and delivery information transmitted in a previous planning schedule transaction, but it does not replace the 830 transaction set. The shipping schedule transaction set shall not be used to authorize labor, materials, or other resources.

The use of this transaction set will facilitate the practice of Just-In-Time (JIT) manufacturing by providing the customer with a mechanism to issue precise shipping schedule requirements on a more frequent basis than with the issuance of a planning schedule transaction, e.g. daily shipping schedules versus weekly planning schedules. The shipping schedule transaction also provides the ability for a customer location to issue shipping requirements independent of other customer locations when planning schedules are issued by a consolidated scheduling organization.

Planning Schedule EDI

Planning Schedule With Release Capability (830)

A Planning schedule with release capability transaction set provides for customary and established business practice relative to the transfer of forecasting/material release information between organizations.

The planning schedule transaction may be used in various ways or in a combination of ways, such as:

- A simple forecast
- A forecast with the buyer's authorization for the seller to commit resources, such as labor or material
- A forecast that is also used as an order release mechanism, containing such elements as resource authorizations, period-to-date cumulative quantities, and specific ship/delivery patterns for requirements that have been represented in buckets, such as weekly, monthly, or quarterly

The order release forecast may also contain all data related to purchase orders, as required, because the order release capability eliminates the need for discreet generation of purchase orders.

Purpose Codes

EDI Demand Transaction Purpose Codes vs. Oracle Purpose Codes

For a customer demand transaction to be processed correctly in the Demand Processor, the appropriate Oracle purpose codes must be associated with it.

EDI Gateway Code Conversion cross-references for RLM_TRX_PURP are pre-defined for the standard purpose codes used in EDI transactions for ASC X12 and EDIFACT standards.

You must define additional customer-specific code conversion cross-references in the EDI Gateway that will uniquely map to the correct Oracle purpose code if:

- Your customer uses a different EDI standard whose purpose codes are not

pre-defined.

- You have EDI trading partner customers who send demand transactions with a purpose code that differs in meaning from the pre-defined cross-reference to the Oracle purpose code (external values 2-4 can contain Trading Partner Code, Schedule Type, Schedule Type Qualifier, or other element as needed).

For example, your customer may use an **Original** purpose code for the first schedule of a new model year, which is intended to replace all previously issued demand from the prior model year. If customer-specific mapping is defined to link this schedule to an Oracle **Replace** purpose code, the demand will be handled properly. However, if the customer-specific mapping is not defined, the Oracle **Original** purpose code will be assigned. This will result in overstated demand, because demand from the last schedule of the prior model year will remain on file, and demand from the first schedule of the new model year will be added.

Assuming that existing and new demand came from the same schedule type (for example, 830, 862, or 866), the following chart describes Oracle purpose codes and their functionality:

Oracle Purpose Code

Code	Demand Processor Interpretation
Original	The requirements on this schedule have not been previously issued. They will all be added, even if existing demand is on file.
Cancellation	The requirements on this schedule will be canceled.
Change	This is a partial revision of a previous schedule for selected items; the requirements on this schedule will replace what was sent on a previous schedule for these items within the schedule horizon.
Replace	<p>The requirements on this schedule will replace existing requirements from a previous schedule, subject to the following exceptions:</p> <p>for items not on the schedule, all existing demand remains intact</p> <p>for schedule items, existing demand dated after the schedule horizon end date remains intact.</p>

Code	Demand Processor Interpretation
Add	The requirements on this schedule will be added to existing requirements from a previous schedule.
Delete	The requirements on this schedule will be deleted from existing requirements from a previous schedule.
Confirmation	The requirements on this schedule will be archived but not processed, assuming that changes were made manually before the schedule was received. They are changes to a previous schedule which confirm verbal emergency requirements.

Transaction Descriptions

Delivery Schedule (DELFOR)

A message from buyer to supplier giving product requirements regarding details for short term delivery instructions and/or medium to long term product/service forecast for planning purposes according to conditions set out in a contract or order. The message can be used to authorized the commitment of labor and materials resources; however, authorizations are not actually used in Europe.

Delivery Just In Time (DELJIT)

A message provides the ability for a customer to convey precise delivery sequence and Just In Time schedule requirements to a supplier, and is intended to supplement the Delivery Schedule message (DELFOR). It provides a mechanism to issue precise shipping schedule requirements in terms of a 24-hour clock and on a more frequent basis than with the instance of a delivery schedule message, e.g. daily shipping schedules versus weekly planning schedules.

Delivery Instruction (DELINS)

The Delivery Instruction Message is sent by a buyer and is intended to provide information regarding details for both short term delivery instructions and medium to long term requirements for planning purposes according to conditions set out in a contract or order.

Note: The analysis is based upon the Odette message DELINS, version 4. The Odette Delivery Instruction message corresponds in purpose and content to the ASC X12 830 - Planning Schedule with Release Capability and the X12 862 - Shipping Schedule.

KANBAN Signal (KANBAN)

The KANBAN Signal Message is issued by a consignee giving authorization to the consignor to ship material based upon receiving a kanban signal and following the principles of the Just-In-Time philosophy.

Note: The analysis is based upon the Odette KANBAN message, version 2. The Odette message corresponds in purpose and content to the ASC X12 862 - Shipping Schedule when used as a kanban Signal.

For more information, see the *Oracle XML Gateway User's Guide*.

Oracle Release Management Open Interfaces and APIs

This chapter covers the following topics:

- Understanding the Demand Processor Interface Tables

Understanding the Demand Processor Interface Tables

Oracle Release Management uses 2 tables in which the Demand Processor receives data that you import from other systems.

- Release Management Schedule Headers: RLM_INTERFACE_HEADERS_ALL
- Release Management Schedule Lines: RLM_INTERFACE_LINES_ALL

When the Demand Processor receives data, it validates and converts your imported data into customer demand schedules within Oracle Release Management and order lines within Oracle Order Management.

Release Management Schedule Headers RLM_INTERFACE_HEADERS_ALL

This table stores the header level details of the Release Management schedules. It has details pertaining to the specific schedule and customer level information. The Release Management Demand Processor uses this table to maintain a current picture of demand schedule headers.

RLM_INTERFACE_HEADERS_ALL

Column Name	Type	Required	Derived	Optional
HEADER_ID	NUMBER	X	-	-

Column Name	Type	Required	Derived	Optional
CUST_ADDRES S_1_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_2_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_3_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_4_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_5_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_6_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_7_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_8_EXT	VARCHAR2(35)	-	-	X
CUST_ADDRES S_9_EXT	VARCHAR2(35)	-	-	X
CUST_CITY_EX T	VARCHAR2(30)	-	-	X
CUST_COUNTR Y_EXT	VARCHAR2(3)	-	-	X
CUST_COUNTY _EXT	VARCHAR2(25)	-	-	X
CUSTOMER_EX T	VARCHAR2(30)	-	-	X
CUST_NAME_E XT	VARCHAR2(20)	-	-	X

Column Name	Type	Required	Derived	Optional
ONE_TIME_CU ST_FLAG_EXT	VARCHAR2(35)	-	-	X
CURRENCY_EX T	VARCHAR2(35)	-	-	X
TAX_EXEMPT_F LAG_EXT	VARCHAR2(35)	-	-	X
TAX_ID_EXT	VARCHAR2(35)	-	-	X
CUST_POSTAL_ CD_EXT	VARCHAR2(15)	-	-	X
CUST_PROVIN CE_EXT	VARCHAR2(10)	-	-	X
CUST_STATE_E XT	VARCHAR2(10)	-	-	X
CUSTOMER_ID	NUMBER	-	-	X
ECE_PRIMARY_ ADDRESS_ID	NUMBER	-	-	X
ECE_TP_LOCAT ION_CODE_EX T	VARCHAR2(35)	-	-	X
ECE_TP_TRANS LATOR_CODE	VARCHAR2(35)	-	-	X
EDI_CONTROL _NUM_1	VARCHAR2(15)	-	-	X
EDI_CONTROL _NUM_2	VARCHAR2(15)	-	-	X
EDI_CONTROL _NUM_3	VARCHAR2(15)	-	-	X

Column Name	Type	Required	Derived	Optional
EDI_TEST_INDICATOR	VARCHAR2(1)	-	-	X
HEADER_CON TACT_CODE_1	VARCHAR2(3)	-	-	X
HEADER_CON TACT_CODE_2	VARCHAR2(3)	-	-	X
HEADER_CON TACT_VALUE_1	VARCHAR2(80)	-	-	X
HEADER_CON TACT_VALUE_2	VARCHAR2(80)	-	-	X
HEADER_CON TACT_EMAIL_1 _EXT	VARCHAR2(50)	-	-	X
HEADER_CON TACT_EMAIL_2 _EXT	VARCHAR2(50)	-	-	X
HEADER_CON TACT_FAX_1_E XT	VARCHAR2(35)	-	-	X
HEADER_CON TACT_FAX_2_E XT	VARCHAR2(35)	-	-	X
HEADER_CON TACT_TEL_1_E XT	VARCHAR2(50)	-	-	X
HEADER_CON TACT_TEL_2_E XT	VARCHAR2(80)	-	-	X
HEADER_NOTE _TEXT	VARCHAR2(400 0)	-	-	X

Column Name	Type	Required	Derived	Optional
HEADER_REF_CODE_1	VARCHAR2(3)	-	-	X
HEADER_REF_CODE_2	VARCHAR2(3)	-	-	X
HEADER_REF_CODE_3	VARCHAR2(3)	-	-	X
HEADER_REF_VALUE_1	VARCHAR2(35)	-	-	X
HEADER_REF_VALUE_2	VARCHAR2(35)	-	-	X
HEADER_REF_VALUE_3	VARCHAR2(35)	-	-	X
ORG_ID	NUMBER	-	-	X
PROCESS_STAT US	NUMBER	-	-	X
SCHEDULE_HEADER_ID	NUMBER	-	-	X
SCHEDULE_TYPE	VARCHAR2(30)	-	X	-
SCHEDULE_TYPE_EXT	VARCHAR2(30)	-	-	X
SCHED_GENERATION_DATE	DATE	-	-	X
SCHED_HORIZON_END_DATE	DATE	-	-	X
SCHED_HORIZON_START_DATE	DATE	-	-	X

Column Name	Type	Required	Derived	Optional
SCHEDULE_PU RPOSE	VARCHAR2(30)	-	X	-
SCHEDULE_PU RPOSE_EXT	VARCHAR2(30)	-	-	X
SCHEDULE_RE FERENCE_NUM	VARCHAR2(35)	-	-	X
SCHEDULE_SO URCE	VARCHAR2(30)	-	-	X
LAST_UPDATE _DATE	DATE	-	X	-
LAST_UPDATE D_BY	NUMBER	-	X	-
CREATION_DA TE	DATE	-	X	-
CREATED_BY	NUMBER	-	X	-
ATTRIBUTE_CA TEGORY	VARCHAR2(30)	-	-	X
ATTRIBUTE1	VARCHAR2(150)	-	-	X
ATTRIBUTE2	VARCHAR2(150)	-	-	X
ATTRIBUTE3	VARCHAR2(150)	-	-	X
ATTRIBUTE4	VARCHAR2(150)	-	-	X
ATTRIBUTE5	VARCHAR2(150)	-	-	X

Column Name	Type	Required	Derived	Optional
ATTRIBUTE6	VARCHAR2(150)	-	-	X
ATTRIBUTE7	VARCHAR2(150)	-	-	X
ATTRIBUTE8	VARCHAR2(150)	-	-	X
ATTRIBUTE9	VARCHAR2(150)	-	-	X
ATTRIBUTE10	VARCHAR2(150)	-	-	X
ATTRIBUTE11	VARCHAR2(150)	-	-	X
ATTRIBUTE12	VARCHAR2(150)	-	-	X
ATTRIBUTE13	VARCHAR2(150)	-	-	X
ATTRIBUTE14	VARCHAR2(150)	-	-	X
ATTRIBUTE15	VARCHAR2(150)	-	-	X
LAST_UPDATE _LOGIN	NUMBER	-	X	-
REQUEST_ID	NUMBER	-	X	-
PROGRAM_AP PLICATION_ID	NUMBER	-	X	-
PROGRAM_ID	NUMBER	-	X	-
PROGRAM_UP DATE_DATE	DATE	-	X	-

Column Name	Type	Required	Derived	Optional
TP_ATTRIBUTE 1	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 2	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 3	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 4	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 5	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 6	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 7	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 8	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 9	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 10	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 11	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 12	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 13	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 14	VARCHAR2(150)	-	-	X

Column Name	Type	Required	Derived	Optional
TP_ATTRIBUTE 15	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE _CATEGORY	VARCHAR2(30)	-	-	X
CUST_FAX_EXT	VARCHAR2(35)	-	-	X
CUST_REGION_ EXT	VARCHAR2(35)	-	-	X
CUST_TAX_JUR ISDICTION_EXT	VARCHAR2(35)	-	-	X
CUST_TEL_EXT	VARCHAR2(35)	-	-	X
CUST_URL_EXT	VARCHAR2(100)	-	-	X
ORIGINAL_DO CUMENT_DATE	DATE	-	-	X
RESP_DOCUME NT_DATE	DATE	-	-	X
DOCUMENT_D ESCRIPTION	VARCHAR2(50)	-	-	X
DOCUMENT_R EV_NUM	VARCHAR2(35)	-	-	X
RESP_SCHEDU LE_ID	VARCHAR2(35)	-	-	X
RESP_DOCUME NT_REV_NUM	VARCHAR2(35)	-	-	X
BUCKET_END_ DATE	DATE	-	-	X

Column Name	Type	Required	Derived	Optional
BUCKET_START_DATE	DATE	-	-	X
FLEX_BUCKET_CODE	VARCHAR2(30)	-	-	X
SUPPLIER_NAME_EXT	VARCHAR2(60)	-	-	X
ONE_TIME_SUPPLIER_FLAG_EXTENSION	VARCHAR2(35)	-	-	X
SUPPLIER_ID	NUMBER	-	-	X
SUPPLIER_CURRENCY_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_DESCRIPTION_EXT	VARCHAR2(35)	-	-	X
CUST_DESCRIPTION_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_TAX_EXEMPT_FLAG_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_TAX_ID_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADDRESS_1_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADDRESS_2_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADDRESS_3_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADDRESS_4_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
SUPPLIER_ADD RESS_5_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADD RESS_6_EXT	VARCHAR2(240)	-	-	X
SUPPLIER_ADD RESS_7_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADD RESS_8_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_ADD RESS_9_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_CITY _EXT	VARCHAR2(30)	-	-	X
SUPPLIER_COU NTY_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_FAX _EXT	VARCHAR2(35)	-	-	X
SUPPLIER_POS TAL_CD_EXT	VARCHAR2(15)	-	-	X
SUPPLIER_REGI ON_EXT	VARCHAR2(35)	-	-	X
SUPPLIER_STA TE_EXT	VARCHAR2(10)	-	-	X
SUPPLIER_PRO VINCE_EXT	VARCHAR2(10)	-	-	X
SUPPLIER_TAX _JURISDICTION _EXT	VARCHAR2(35)	-	-	X
SUPPLIER_TEL_ EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
SUPPLIER_URL _EXT	VARCHAR2(100)	-	-	X
SUP_HEADER_ CONTACT_CO DE_1	VARCHAR2(3)	-	-	X
SUP_HEADER_ CONTACT_CO DE_2	VARCHAR2(3)	-	-	X
SUP_HEADER_ CONTACT_VAL UE_1	VARCHAR2(80)	-	-	X
SUP_HEADER_ CONTACT_VAL UE_2	VARCHAR2(80)	-	-	X
SUP_HEADER_ CONTACT_EM AIL_1_EXT	VARCHAR2(50)	-	-	X
SUP_HEADER_ CONTACT_EM AIL_2_EXT	VARCHAR2(50)	-	-	X
SUP_HEADER_ CONTACT_FAX _1_EXT	VARCHAR2(35)	-	-	X
SUP_HEADER_ CONTACT_FAX _2_EXT	VARCHAR2(35)	-	-	X
SUP_HEADER_ CONTACT_TEL _1_EXT	VARCHAR2(50)	-	-	X
SUP_HEADER_ CONTACT_TEL _2_EXT	VARCHAR2(50)	-	-	X

Column Name	Type	Required	Derived	Optional
SUPPLIER_COU NTRY_EXT	VARCHAR2(3)	-	-	X

Column Descriptions

HEADER_ID Number

Sequence generated unique identifier.

CUST_ADDRESS_1_EXT VARCHAR2(35)

Customer address line 1, as sent by the customer on the N3 segment.

CUST_ADDRESS_2_EXT VARCHAR2(35)

Customer address line 2, as sent by the customer on the N3 segment.

CUST_ADDRESS_3_EXT VARCHAR2(35)

Customer address line 3, as sent by the customer on the N3 segment.

CUST_ADDRESS_4_EXT VARCHAR2(35)

Customer address line 4, as sent by the customer on the N3 segment.

CUST_ADDRESS_5_EXT VARCHAR2(35)

Customer address line 5, as sent by the customer on the N3 segment.

CUST_ADDRESS_6_EXT VARCHAR2(35)

Customer address line 6, as sent by the customer on the N3 segment.

CUST_ADDRESS_7_EXT VARCHAR2(35)

Customer address line 7, as sent by the customer on the N3 segment.

CUST_ADDRESS_8_EXT VARCHAR2(35)

Customer address line 8, As sent by the customer on the N3 segment.

CUST_ADDRESS_9_EXT VARCHAR2(35)

Customer address line 9, as sent by the customer on the N3 segment.

CUST_CITY_EXT VARCHAR2(30)

Customer address city, as sent by the customer on the N4 segment.

CUST_COUNTRY_EXT VARCHAR2(3)

Customer country, as sent by the customer on the N4 segment.

CUST_COUNTY_EXT VARCHAR2(25)

Customer county, as sent by the customer on the N4 segment.

CUSTOMER_EXT VARCHAR2(30)

Trading Partner designator cross reference.

CUST_NAME_EXT VARCHAR2(60)

Customer name, as sent by the customer in the N1 or N2 segment.

ONE_TIME_CUST_FLAG_EXT VARCHAR2(35)

For future use.

CURRENCY_EXT VARCHAR2(35)

For future use.

TAX_EXEMPT_FLAG_EXT VARCHAR2(35)

For future use.

TAX_ID_EXT VARCHAR2(35)

For future use.

CUST_POSTAL_CD_EXT VARCHAR2(15)

Customer address postal code, as sent by the customer on the N4 segment.

CUST_PROVINCE_EXT VARCHAR2(10)

Customer province, as sent by the customer on the N4 segment.

CUST_STATE_EXT VARCHAR2(10)

Customer state, as sent by the customer on the N4 segment.

CUSTOMER_ID NUMBER

Customer identifier. FK to RA_CUSTOMERS.

ECE_PRIMARY_ADDRESS_ID NUMBER

Customer primary address ID; populated by e-Commerce Gateway code conversion of external trading partner location code. FK to RA_ADDRESSES.

ECE_TP_LOCATION_CODE_EXT VARCHAR2(35)

For EDI transactions only, external trading partner location code assigned to the location in the e-Commerce Gateway common control record.

ECE_TP_TRANSLATOR_CODE VARCHAR2(35)

For EDI transactions only, e-Commerce Gateway trading partner code assigned to the transaction in the e-Commerce Gateway common control record.

EDI_CONTROL_NUM_1 VARCHAR2(15)

EDI control number 1 (ISA) assigned by the customer's EDI translator for audit.

EDI_CONTROL_NUM_2 VARCHAR2(15)

EDI control number 2 (GS) assigned by the customer's EDI translator for audit.

EDI_CONTROL_NUM_3 VARCHAR2(15)

EDI control number 3 (ST) assigned by the customer's EDI translator for audit.

EDI_TEST_INDICATOR VARCHAR2(1)

EDI Test/Production transaction indicator (T or P), assigned by the customer's EDI translator. E-Commerce Gateway rules determine the handling of test transactions within the Demand Processor.

HEADER_CONTACT_CODE_1 VARCHAR2(3)

Contact code 1 included in customer's EDI transaction (not validated).

HEADER_CONTACT_CODE_2 VARCHAR2(3)

Contact code 2 included in customer's EDI transaction (not validated).

HEADER_CONTACT_VALUE_1 VARCHAR2(80)

Values associated with contact code 1: concatenation of name, communication code, and communication number.

HEADER_CONTACT_VALUE_2 VARCHAR2(80)

Values associated with contact code 2: concatenation of name, communication code, and communication number.

HEADER_CONTACT_EMAIL_1_EXT VARCHAR2(50)

For future use.

HEADER_CONTACT_EMAIL_2_EXT VARCHAR2(50)

For future use.

HEADER_CONTACT_FAX_1_EXT VARCHAR2(35)

For future use.

HEADER_CONTACT_FAX_2_EXT VARCHAR2(35)

For future use.

HEADER_CONTACT_TEL_1_EXT VARCHAR2(50)

For future use.

HEADER_CONTACT_TEL_2_EXT VARCHAR2(50)

For future use.

HEADER_NOTE_TEXT VARCHAR2(240)

Free form note text included in the schedule at the header level.

HEADER_REF_CODE_1 VARCHAR2(3)

Reference code 1 included in customer's EDI transaction.

HEADER_REF_CODE_2 VARCHAR2(3)

Reference code 2 included in customer's EDI transaction.

HEADER_REF_CODE_3 VARCHAR2(3)

Reference code 3 included in customer's EDI transaction.

HEADER_REF_VALUE_1 VARCHAR2(35)

Value associated with reference code 1.

HEADER_REF_VALUE_2 VARCHAR2(35)

Value associated with reference code 2.

HEADER_REF_VALUE_3 VARCHAR2(35)

Value associated with reference code 3.

ORG_ID NUMBER

Operating unit unique ID. FK to HR_ORGANIZATION_UNITS.

PROCESS_STATUS NUMBER

Indicates the current processing status of a record for headers with lines in error:

1= Available to be processed

2= Processed with error

3= Processed successfully

4= Line is frozen firm

5= Partially processed with error(s)

SCHEDULE_HEADER_ID NUMBER

Schedule header unique identifier. FK to RLM_SCHEDULE_HEADERS.

SCHEDULE_TYPE VARCHAR2(30)

Schedule Type: planning, shipping or sequenced. Validated against
FND_LOOKUPS.LOOKUP_TYPE = RLM_SCHEDULE_TYPE.

SCHEDULE_TYPE_EXT VARCHAR2(30)

External value from which RLM_SCHEDULE_TYPE was derived in EDI Gateway code
conversion.

SCHED_GENERATION_DATE DATE

Date customer generated either the schedule or the planning information on the
schedule.

SCHED_HORIZON_END_DATE DATE

Schedule horizon end date, to be derived based on demand contained within the
schedule if not specified.

SCHED_HORIZON_START_DATE DATE

Schedule horizon start date, to be derived based on demand contained within the schedule if not specified.

SCHEDULE_PURPOSE VARCHAR2(30)

Schedule purpose code, i.e. original, replace, cancellation, etc. Validated against FND_LOOKUPS.LOOKUP_TYPE =RLM_SCHEDULE_PURPOSE

SCHEDULE_PURPOSE_EXT VARCHAR2(30)

External value from which SCHEDULE_PURPOSE was derived in e-Commerce Gateway code conversion.

SCHEDULE_REFERENCE_NUM VARCHAR2(35)

Customer assigned schedule reference or release number.

SCHEDULE_SOURCE VARCHAR2(30)

EDI or manual transaction source for this schedule, i.e. 830, 862, 866, DELFOR, MANUAL, etc.

LAST_UPDATE_DATE DATE

Standard Who column.

LAST_UPDATED_BY NUMBER

Standard Who column.

CREATION_DATE DATE

Standard Who column.

CREATED_BY NUMBER

Standard Who column.

ATTRIBUTE_CATEGORY VARCHAR2(30)

Descriptive flexfield context column.

ATTRIBUTE1 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE2 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE3 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE4 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE5 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE6 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE7 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE8 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE9 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE10 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE11 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE12 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE13 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE14 VARCHAR2(150)
Descriptive flexfield segment column.

ATTRIBUTE15 VARCHAR2(150)
Descriptive flexfield segment column.

LAST_UPDATE_LOGIN NUMBER
Standard Who column.

REQUEST_ID NUMBER
Standard Who column.

PROGRAM_APPLICATION_ID NUMBER
Standard Who column.

PROGRAM_ID NUMBER
Standard Who column.

PROGRAM_UPDATE_DATE DATE
Standard Who column.

TP_ATTRIBUTE1 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE2 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE3 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE4 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE5 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE6 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE7 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE8 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE9 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE10 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE11 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE12 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE13 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE14 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE15 VARCHAR2(150)
Trading Partner flexfield column.

TP_ATTRIBUTE_CATEGORY VARCHAR2(30)
Trading Partner flexfield context column.

CUST_FAX_EXT VARCHAR2(35)
For future use.

CUST_REGION_EXT VARCHAR2(35)

For future use.

CUST_TAX_JURISDICTION_EXT VARCHAR2(35)

For future use.

CUST_TEL_EXT VARCHAR2(35)

For future use.

CUST_URL_EXT VARCHAR2(100)

For future use.

ORIGINAL_DOCUMENT_DATE DATE

For future use.

RESP_DOCUMENT_DATE DATE

For future use.

DOCUMENT_DESCRIPTION VARCHAR2(50)

For future use.

DOCUMENT_REV_NUM VARCHAR2(35)

For future use.

RESP_SCHEDULE_ID VARCHAR2(35)

For future use.

RESP_DOCUMENT_REV_NUM VARCHAR2(35)

For future use.

BUCKET_END_DATE DATE

For future use.

BUCKET_START_DATE DATE

For future use.

FLEX_BUCKET_CODE VARCHAR2(30)

For future use.

SUPPLIER_NAME_EXT VARCHAR2(60)

For future use.

ONE_TIME_SUPPLIER_FLAG_EXT VARCHAR2(35)

For future use.

SUPPLIER_ID NUMBER

For future use.

SUPPLIER_CURRENCY_EXT VARCHAR2(35)

For future use.

SUPPLIER_DESCRIPTION_EXT VARCHAR2(35)

For future use.

CUST_DESCRIPTION_EXT VARCHAR2(35)

For future use.

SUPPLIER_TAX_EXEMPT_FLAG_EXT VARCHAR2(35)

For future use.

SUPPLIER_TAX_ID_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_1_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_2_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_3_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_4_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_5_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_6_EXT VARCHAR2(240)

For future use.

SUPPLIER_ADDRESS_7_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_8_EXT VARCHAR2(35)

For future use.

SUPPLIER_ADDRESS_9_EXT VARCHAR2(35)

For future use.

SUPPLIER_CITY_EXT VARCHAR2(30)

For future use.

SUPPLIER_COUNTY_EXT VARCHAR2(35)

For future use.

SUPPLIER_FAX_EXT VARCHAR2(35)
For future use.

SUPPLIER_POSTAL_CD_EXT VARCHAR2(15)
For future use.

SUPPLIER_REGION_EXT VARCHAR2(35)
For future use.

SUPPLIER_STATE_EXT VARCHAR2(10)
For future use.

SUPPLIER_PROVINCE_EXT VARCHAR2(10)
For future use.

SUPPLIER_TAX_JURISDICTION_EXT VARCHAR2(35)
For future use.

SUPPLIER_TEL_EXT VARCHAR2(35)
For future use.

SUPPLIER_URL_EXT VARCHAR2(100)
For future use.

SUP_HEADER_CONTACT_CODE_1 VARCHAR2(3)
For future use.

SUP_HEADER_CONTACT_CODE_2 VARCHAR2(3)
For future use.

SUP_HEADER_CONTACT_VALUE_1 VARCHAR2(80)
For future use.

SUP_HEADER_CONTACT_VALUE_2 VARCHAR2(80)
For future use.

SUP_HEADER_CONTACT_EMAIL_1_EXT VARCHAR2(50)
For future use.

SUP_HEADER_CONTACT_EMAIL_2_EXT VARCHAR2(50)
For future use.

SUP_HEADER_CONTACT_FAX_1_EXT VARCHAR2(35)
For future use.

SUP_HEADER_CONTACT_FAX_2_EXT VARCHAR2(35)
For future use.

SUP_HEADER_CONTACT_TEL_1_EXT VARCHAR2(50)

For future use.

SUP_HEADER_CONTACT_TEL_2_EXT VARCHAR2(50)

For future use.

SUPPLIER_COUNTRY_EXT VARCHAR2(3)

For future use.

Release Management Schedule Lines RLM_INTERFACE_LINES_ALL

This table stores the item and item detail level information associated with customer planning, shipping, or production sequenced schedules. For production sequence schedules, the item represents the feature, not the configuration.

This table stores all details for scheduled items: dated demand, authorizations, and customer shipment/receipt information. It has a child relationship to RLM_INTERFACE_HEADERS_ALL.

RLM_INTERFACE_LINES_ALL

Column Name	Type	Required	Derived	Optional
LINE_ID	NUMBER	X	-	-
HEADER_ID	NUMBER	X	-	-
AGREEMENT_ID	NUMBER	-	-	X
ATO_DATA_TYPE	VARCHAR2(30)	-	-	X
BILL_TO_ADDRESS_1_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_2_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_3_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_4_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
BILL_TO_ADDR ESS_ID	NUMBER	-	-	X
INVOICE_TO_O RG_ID	NUMBER	-	-	X
BILL_TO_CITY_ EXT	VARCHAR2(30)	-	-	X
BILL_TO_COU NTRY_EXT	VARCHAR2(3)	-	-	X
BILL_TO_COU NTY_EXT	VARCHAR2(25)	-	-	X
BILL_TO_NAM E_EXT	VARCHAR2(60)	-	-	X
BILL_TO_POST AL_CD_EXT	VARCHAR2(15)	-	-	X
BILL_TO_PROV INCE_EXT	VARCHAR2(10)	-	-	X
BILL_TO_SITE_ USE_ID	NUMBER	-	-	X
BILL_TO_STAT E_EXT	VARCHAR2(10)	-	-	X
CARRIER_ID_C ODE_EXT	VARCHAR2(35)	-	-	X
CARRIER_QUA LIFIER_EXT	VARCHAR2(3)	-	-	X
COMMODITY_ EXT	VARCHAR2(35)	-	-	X
COUNTRY_OF_ ORIGIN_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
CUST_ASSEMBLY_EXT	VARCHAR2(30)	-	-	X
CUST_ASSIGNED_ID_EXT	VARCHAR2(20)	-	-	X
CUST_BILL_TO_EXT	VARCHAR2(35)	-	-	X
CUST_CONTRACT_NUM_EXT	VARCHAR2(35)	-	-	X
CUSTOMER_DOCK_CODE	VARCHAR2(50)	-	-	X
CUST_INTRMD_SHIP_TO_EXT	VARCHAR2(35)	-	-	X
CUST_ITEM_PRICE_EXT	NUMBER	-	-	X
CUST_ITEM_PRICE_UOM_EXT	VARCHAR2(3)	-	-	X
CUSTOMER_ITEM_REVISION	VARCHAR2(3)	-	-	X
CUSTOMER_JOB	VARCHAR2(50)	-	-	X
CUST_MANUFACTURER_EXT	VARCHAR2(35)	-	-	X
CUST_MODEL_NUMBER_EXT	VARCHAR2(35)	-	-	X
CUST_MODEL_SERIAL_NUMBER	VARCHAR2(35)	-	-	X
CUST_ORDER_NUM_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
CUST_PROCESS_NUM_EXT	VARCHAR2(35)	-	-	X
CUST_PRODUCTION_LINE	VARCHAR2(50)	-	-	X
CUST_PRODUCTION_SEQUENCE_NUM	VARCHAR2(35)	-	-	X
CUST_SET_NUM_EXT	VARCHAR2(35)	-	-	X
CUST_SHIP_FROM_ORG_EXT	VARCHAR2(80)	-	-	X
CUST_SHIP_TO_EXT	VARCHAR2(35)	-	-	X
CUST_UOM_EXT	VARCHAR2(10)	-	-	X
CUSTOMER_ITEM_EXT	VARCHAR2(50)	-	-	X
CUSTOMER_ITEM_ID	NUMBER	-	-	X
REQUEST_DATE	DATE	-	-	X
SCHEDULE_DATE	DATE	-	-	X
DATE_TYPE_CODE	VARCHAR2(30)	-	X	-
DATE_TYPE_CODE_EXT	VARCHAR2(30)	-	-	X
DELIVERY_LEAD_TIME	NUMBER	-	-	X

Column Name	Type	Required	Derived	Optional
END_DATE_TIME	DATE	-	-	X
EQUIPMENT_CODE_EXT	VARCHAR2(3)	-	-	X
EQUIPMENT_NUMBER_EXT	VARCHAR2(35)	-	-	X
HANDLING_CODE_EXT	VARCHAR2(3)	-	-	X
HAZARD_CODE_EXT	VARCHAR2(10)	-	-	X
HAZARD_CODE_QUAL_EXT	VARCHAR2(3)	-	-	X
HAZARD_DESCRIPTION_EXT	VARCHAR2(80)	-	-	X
IMPORT_LICENSE_DATE_EXT	DATE	-	-	X
IMPORT_LICENSE_EXT	VARCHAR2(35)	-	-	X
INDUSTRY_ATRIBUTE1	VARCHAR2(150)	-	-	X
INDUSTRY_ATRIBUTE10	VARCHAR2(150)	-	-	X
INDUSTRY_ATRIBUTE11	VARCHAR2(150)	-	-	X
INDUSTRY_ATRIBUTE12	VARCHAR2(150)	-	-	X
INDUSTRY_ATRIBUTE13	VARCHAR2(150)	-	-	X

Column Name	Type	Required	Derived	Optional
INDUSTRY_AT TRIBUTE14	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE15	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE2	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE3	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE4	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE5	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE6	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE7	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE8	VARCHAR2(150)	-	-	X
INDUSTRY_AT TRIBUTE9	VARCHAR2(150)	-	-	X
INDUSTRY_CO NTEXT	VARCHAR2(30)	-	-	X
INTRMD_SHIP_ TO_ID	NUMBER	-	-	X
SHIP_TO_ORG_ ID	NUMBER	-	-	X
INTRMD_ST_A DDRESS_1_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
INTRMD_ST_A DDRESS_2_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_3_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_4_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_CI TY_EXT	VARCHAR2(30)	-	-	X
INTRMD_ST_C OUNTRY_EXT	VARCHAR2(3)	-	-	X
INTRMD_ST_C OUNTY_EXT	VARCHAR2(25)	-	-	X
INTRMD_ST_N AME_EXT	VARCHAR2(60)	-	-	X
INTRMD_ST_P OSTAL_CD_EX T	VARCHAR2(15)	-	-	X
INTRMD_ST_PR OVINCE_EXT	VARCHAR2(10)	-	-	X
INTRMD_ST_ST ATE_EXT	VARCHAR2(10)	-	-	X
INTRMD_ST_SI TE_USE_ID	NUMBER	-	-	X
INVENTORY_IT EM_ID	NUMBER	-	-	X
INVENTORY_IT EM_SEGMENT1	VARCHAR2(40)	-	-	X

Column Name	Type	Required	Derived	Optional
INVENTORY_IT EM_SEGMENT1 0	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 1	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 2	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 3	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 4	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 5	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 6	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 7	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 8	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT1 9	VARCHAR2(40)	-	-	X
INVENTORY_IT EM_SEGMENT2	VARCHAR2(40)	-	-	X

Column Name	Type	Required	Derived	Optional
INVENTORY_ITEM_SEGMENT20	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT3	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT4	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT5	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT6	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT7	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT8	VARCHAR2(40)	-	-	X
INVENTORY_ITEM_SEGMENT9	VARCHAR2(40)	-	-	X
ITEM_CONTACT_CODE_1	VARCHAR2(3)	-	-	X
ITEM_CONTACT_CODE_2	VARCHAR2(3)	-	-	X
ITEM_CONTACT_VALUE_1	VARCHAR2(80)	-	-	X
ITEM_CONTACT_VALUE_2	VARCHAR2(80)	-	-	X
ITEM_DESCRIPTION_EXT	VARCHAR2(80)	-	-	X
ITEM_DETAIL_QUANTITY	NUMBER	-	-	X

Column Name	Type	Required	Derived	Optional
ITEM_DETAIL_ REF_CODE_1	VARCHAR2(3)	-	-	X
ITEM_DETAIL_ REF_CODE_2	VARCHAR2(3)	-	-	X
ITEM_DETAIL_ REF_CODE_3	VARCHAR2(3)	-	-	X
ITEM_DETAIL_ REF_VALUE_1	VARCHAR2(35)	-	-	X
ITEM_DETAIL_ REF_VALUE_2	VARCHAR2(35)	-	-	X
ITEM_DETAIL_ REF_VALUE_3	VARCHAR2(35)	-	-	X
ITEM_DETAIL_ SUBTYPE	VARCHAR2(30)	-	X	-
ITEM_DETAIL_ SUBTYPE_EXT	VARCHAR2(30)	-	-	X
ITEM_DETAIL_ TYPE	VARCHAR2(30)	-	X	-
ITEM_DETAIL_ TYPE_EXT	VARCHAR2(30)	-	-	X
ITEM_ENG_CN G_LVL_EXT	VARCHAR2(35)	-	-	X
ITEM_MEASUR EMENTS_EXT	VARCHAR2(240)	-	-	X
ITEM_NOTE_TE XT	VARCHAR2(240)	-	-	X
ITEM_REF_COD E_1	VARCHAR2(3)	-	-	X

Column Name	Type	Required	Derived	Optional
ITEM_REF_CODE_2	VARCHAR2(3)	-	-	X
ITEM_REF_CODE_3	VARCHAR2(3)	-	-	X
ITEM_REF_VALUE_1	VARCHAR2(35)	-	-	X
ITEM_REF_VALUE_2	VARCHAR2(35)	-	-	X
ITEM_REF_VALUE_3	VARCHAR2(35)	-	-	X
ITEM_RELEASE_STATUS_EXT	VARCHAR2(3)	-	-	X
LADING_QUANTITY_EXT	NUMBER	-	-	X
LETTER_CREDIT_EXPDT_EXT	DATE	-	-	X
LETTER_CREDIT_EXT	VARCHAR2(35)	-	-	X
LINE_REFERENCE	VARCHAR2(50)	-	-	X
LINK_TO_LINE_REF	VARCHAR2(50)	-	-	X
ORDER_HEADER_ID	NUMBER	-	-	X
ORG_ID	NUMBER	-	-	X
OTHER_NAME_CODE_1	VARCHAR2(3)	-	-	X

Column Name	Type	Required	Derived	Optional
OTHER_NAME _CODE_2	VARCHAR2(3)	-	-	X
OTHER_NAME _VALUE_1	VARCHAR2(80)	-	-	X
OTHER_NAME _VALUE_2	VARCHAR2(80)	-	-	X
PACK_SIZE_EX T	NUMBER	-	-	X
PACK_UNITS_P ER_PACK_EXT	NUMBER	-	-	X
PACK_UOM_C ODE_EXT	VARCHAR2(3)	-	-	X
PACKAGING_C ODE_EXT	VARCHAR2(10)	-	-	X
PARENT_LINK_ LINE_REF	VARCHAR2(50)	-	-	X
PRICE_LIST_ID	NUMBER	-	-	X
PRIMARY_QUA NTITY	NUMBER	-	-	X
PRIMARY_UO M_CODE	VARCHAR2(3)	-	-	X
PRIME_CONTR CTR_PART_EXT	VARCHAR2(35)	-	-	X
PROCESS_STAT US	NUMBER	-	-	X
CUST_PO_RELE ASE_NUM	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
CUST_PO_DATE	DATE	-	-	X
CUST_PO_LINE_NUM	VARCHAR2(35)	-	-	X
CUST_PO_NUMBER	VARCHAR2(50)	-	-	X
QTY_TYPE_CODE	VARCHAR2(30)	-	X	-
QTY_TYPE_CODE_EXT	VARCHAR2(30)	-	-	X
RETURN_CONTAINER_EXT	VARCHAR2(35)	-	-	X
SCHEDULE_LINE_ID	NUMBER	-	-	X
ROUTING_DESC_EXT	VARCHAR2(35)	-	-	X
ROUTING_SEQUENCE_CODE_EXT	VARCHAR2(3)	-	-	X
SCHEDULE_ITEM_NUM	NUMBER	-	-	X
SHIP_DEL_PATERN_EXT	VARCHAR2(3)	-	-	X
SHIP_DEL_TIME_CODE_EXT	VARCHAR2(3)	-	-	X
SHIP_DEL_RULE_NAME	VARCHAR2(30)	-	-	X
SHIP_FROM_ADDRESS_1_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
SHIP_FROM_ADDRESS_2_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_ADDRESS_3_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_ADDRESS_4_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_CITY_EXT	VARCHAR2(30)	-	-	X
SHIP_FROM_COUNTRY_EXT	VARCHAR2(3)	-	-	X
SHIP_FROM_COUNTY_EXT	VARCHAR2(25)	-	-	X
SHIP_FROM_NAME_EXT	VARCHAR2(60)	-	-	X
SHIP_FROM_ORG_ID	NUMBER	-	-	X
SHIP_FROM_POSTAL_CD_EXT	VARCHAR2(15)	-	-	X
SHIP_FROM_PROVINCE_EXT	VARCHAR2(10)	-	-	X
SHIP_FROM_STATE_EXT	VARCHAR2(10)	-	-	X
SHIP_LABEL_INFO_LINE_1	VARCHAR2(80)	-	-	X
SHIP_LABEL_INFO_LINE_10	VARCHAR2(80)	-	-	X
SHIP_LABEL_INFO_LINE_2	VARCHAR2(80)	-	-	X

Column Name	Type	Required	Derived	Optional
SHIP_LABEL_I NFO_LINE_3	VARCHAR2(80)	-	-	X
SHIP_LABEL_I NFO_LINE_4	VARCHAR2(80)	-	-	X
SHIP_LABEL_I NFO_LINE_5	VARCHAR2(80)	-	-	X
SHIP_LABEL_I NFO_LINE_6	VARCHAR2(80)	-	-	X
SHIP_LABEL_I NFO_LINE_7	VARCHAR2(80)	-	-	X
SHIP_LABEL_I NFO_LINE_8	VARCHAR2(80)	-	-	X
SHIP_LABEL_I NFO_LINE_9	VARCHAR2(80)	-	-	X
SHIP_TO_ADD RESS_1_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADD RESS_2_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADD RESS_3_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADD RESS_4_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADD RESS_ID	NUMBER	-	-	X
DELIVER_TO_O RG_ID	NUMBER	-	-	X
SHIP_TO_CITY_ EXT	VARCHAR2(30)	-	-	X

Column Name	Type	Required	Derived	Optional
SHIP_TO_COUNTRY_EXT	VARCHAR2(3)	-	-	X
SHIP_TO_COUNTRY_EXT	VARCHAR2(25)	-	-	X
SHIP_TO_NAME_EXT	VARCHAR2(60)	-	-	X
SHIP_TO_POSTAL_CD_EXT	VARCHAR2(15)	-	-	X
SHIP_TO_PROVINCE_EXT	VARCHAR2(10)	-	-	X
SHIP_TO_SITE_USE_ID	NUMBER	-	-	X
SHIP_TO_STATE_EXT	VARCHAR2(10)	-	-	X
START_DATE_TIME	DATE	-	-	X
SUBLINE_ASSIGNED_ID_EXT	VARCHAR2(20)	-	-	X
SUBLINE_CONFIG_CODE_EXT	VARCHAR2(3)	-	-	X
SUBLINE_CUST_ITEM_EXT	VARCHAR2(50)	-	-	X
SUBLINE_CUST_ITEM_ID	NUMBER	-	-	X
SUBLINE_MODEL_NUM_EXT	VARCHAR2(35)	-	-	X
SUBLINE_QUANTITY	NUMBER	-	-	X

Column Name	Type	Required	Derived	Optional
SUBLINE_UOM_CODE	VARCHAR2(3)	-	-	X
SUPPLIER_ITEM_EXT	VARCHAR2(35)	-	-	X
TRANSIT_TIME_EXT	VARCHAR2(22)	-	-	X
TRANSIT_TIME_QUAL_EXT	VARCHAR2(3)	-	-	X
TRANSPORT_LOCATION_QUAL_EXT	VARCHAR2(3)	-	-	X
TRANSPORT_LOCATION_EXT	VARCHAR2(35)	-	-	X
TRANSPORT_METHOD_EXT	VARCHAR2(3)	-	-	X
UOM_CODE	VARCHAR2(3)	-	-	X
WEIGHT_EXT	NUMBER	-	-	X
WEIGHT_QUALIFIER_EXT	VARCHAR2(3)	-	-	X
WEIGHT_UOM_EXT	VARCHAR2(3)	-	-	X
FBO_CONFIGURATION_KEY_1	VARCHAR2(35)	-	-	X
FBO_CONFIGURATION_KEY_2	VARCHAR2(35)	-	-	X
FBO_CONFIGURATION_KEY_3	VARCHAR2(35)	-	-	X
FBO_CONFIGURATION_KEY_4	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
FBO_CONFIGUR ATION_KEY_5	VARCHAR2(35)	-	-	X
MATCH_KEY_A CROSS	VARCHAR2(150)	-	-	X
MATCH_KEY_ WITHIN	VARCHAR2(150)	-	-	X
CRITICAL_KEY _ATTRIBUTES	VARCHAR2(150)	-	-	X
LAST_UPDATE _DATE	DATE	-	X	-
LAST_UPDATE D_BY	NUMBER	-	X	-
CREATION_DA TE	DATE	-	X	-
CREATED_BY	NUMBER	-	X	-
ATTRIBUTE_CA TEGORY	VARCHAR2(30)	-	-	X
ATTRIBUTE1	VARCHAR2(150)	-	-	X
ATTRIBUTE2	VARCHAR2(150)	-	-	X
ATTRIBUTE3	VARCHAR2(150)	-	-	X
ATTRIBUTE4	VARCHAR2(150)	-	-	X
ATTRIBUTE5	VARCHAR2(150)	-	-	X

Column Name	Type	Required	Derived	Optional
ATTRIBUTE6	VARCHAR2(150)	-	-	X
ATTRIBUTE7	VARCHAR2(150)	-	-	X
ATTRIBUTE8	VARCHAR2(150)	-	-	X
ATTRIBUTE9	VARCHAR2(150)	-	-	X
ATTRIBUTE10	VARCHAR2(150)	-	-	X
ATTRIBUTE11	VARCHAR2(150)	-	-	X
ATTRIBUTE12	VARCHAR2(150)	-	-	X
ATTRIBUTE13	VARCHAR2(150)	-	-	X
ATTRIBUTE14	VARCHAR2(150)	-	-	X
ATTRIBUTE15	VARCHAR2(150)	-	-	X
LAST_UPDATE _LOGIN	NUMBER	-	X	-
REQUEST_ID	NUMBER	-	X	-
PROGRAM_AP PLICATION_ID	NUMBER	-	X	-
PROGRAM_ID	NUMBER	-	X	-
PROGRAM_UP DATE_DATE	DATE	-	X	-

Column Name	Type	Required	Derived	Optional
TP_ATTRIBUTE 1	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 2	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 3	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 4	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 5	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 6	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 7	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 8	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 9	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 10	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 11	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 12	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 13	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE 14	VARCHAR2(150)	-	-	X

Column Name	Type	Required	Derived	Optional
TP_ATTRIBUTE 15	VARCHAR2(150)	-	-	X
TP_ATTRIBUTE _CATEGORY	VARCHAR2(30)	-	-	X
LINE_NUMBER	NUMBER	-	-	X
INTMED_SHIP_ TO_ORG_ID	NUMBER	-	-	X
LINE_SOURCE	VARCHAR2(30)	-	-	X
PREFERRED_G RADE	VARCHAR2(4)	-	-	X
CUST_PRODUC TION_SEQ_NU M_BEG	VARCHAR2(35)	-	-	X
CUST_PRODUC TION_SEQ_NU M_END	VARCHAR2(35)	-	-	X
ITEM_DETAIL_ QUANTITY_MI N_EXT	NUMBER	-	-	X
ITEM_DETAIL_ QUANTITY_MA X_EXT	NUMBER	-	-	X
ITEM_DETAIL_ QUANTITY_PRI OR_EXT	NUMBER	-	-	X
REQUIREMENT _PRIORITY_EXT	NUMBER	-	-	X
PROJECT_NUM BER_EXT	VARCHAR2(50)	-	-	X

Column Name	Type	Required	Derived	Optional
RESP_LINE_NUMBER	VARCHAR2(35)	-	-	X
TASK_NUMBER_EXT	VARCHAR2(50)	-	-	X
UPC_EXT	VARCHAR2(50)	-	-	X
ITEM_ENG_CNG_LVL_DATE_EXT	DATE	-	-	X
AUTHORIZATION_CODE_EXT	VARCHAR2(50)	-	-	X
PACKAGING_DESC_EXT	VARCHAR2(80)	-	-	X
BILL_TO_ADDRESS_5_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_6_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_7_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_8_EXT	VARCHAR2(35)	-	-	X
BILL_TO_ADDRESS_9_EXT	VARCHAR2(35)	-	-	X
ONE_TIME_BILL_TO_FLAG_EXT	VARCHAR2(35)	-	-	X
BILL_TO_DESCRIPTION_EXT	VARCHAR2(50)	-	-	X
BILL_TO_FAX_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
BILL_TO_REGI ON_EXT	VARCHAR2(35)	-	-	X
BILL_TO_JURIS DICTION_EXT	VARCHAR2(35)	-	-	X
BILL_TO_TEL_E XT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_5_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_6_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_7_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_8_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_A DDRESS_9_EXT	VARCHAR2(35)	-	-	X
ONE_TIME_INT RMD_ST_FLAG _EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_D ESCRPTION_E XT	VARCHAR2(50)	-	-	X
INTRMD_ST_FA X_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_RE GION_EXT	VARCHAR2(35)	-	-	X
INTRMD_ST_JU RISDICTION_E XT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
INTRMD_ST_TEL_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADDRESS_5_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADDRESS_6_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADDRESS_7_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADDRESS_8_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_ADDRESS_9_EXT	VARCHAR2(35)	-	-	X
ONE_TIME_SHIP_TO_FLAG_EXTENSION	VARCHAR2(35)	-	-	X
SHIP_TO_DESCRIPTION_EXT	VARCHAR2(50)	-	-	X
SHIP_TO_FAX_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_REGION_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_JURISDICTION_EXT	VARCHAR2(35)	-	-	X
SHIP_TO_TEL_EXTENSION	VARCHAR2(35)	-	-	X
SHIP_FROM_ADDRESS_5_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_ADDRESS_6_EXT	VARCHAR2(35)	-	-	X

Column Name	Type	Required	Derived	Optional
SHIP_FROM_ADDRESS_7_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_ADDRESS_8_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_ADDRESS_9_EXT	VARCHAR2(35)	-	-	X
ONE_TIME_SHIP_FROM_FLAG_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_DESCRIPTION_EXT	VARCHAR2(50)	-	-	X
SHIP_FROM_FACTX_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_REGION_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_JURISDICTION_EXT	VARCHAR2(35)	-	-	X
SHIP_FROM_TEL_EXT	VARCHAR2(35)	-	-	X

Column Descriptions

LINE_ID NUMBER

Sequence generated unique identifier.

HEADER_ID NUMBER

Sequence generated unique identifier foreign key to the RLM_INTERFACE_HEADERS_ALL.

AGREEMENT_ID NUMBER

Unique identifier for agreement on which customer purchase order is associated. FK to RA_AGREEMENTS

ATO_DATA_TYPE VARCHAR2(30)

Code to describe what type of data is included for the ATO item:

1=Model and Options

2=Model

3=Options

Validated against FND_LOOKUPS.LOOKUP_TYPE =RLM_ITEM_METHOD_TYPE

BILL_TO_ADDRESS_1_EXT VARCHAR2(35)

Bill to address line 1, as sent by the customer on the N3 segment.

BILL_TO_ADDRESS_2_EXT VARCHAR2(35)

Bill to address line 2, as sent by the customer on the N3 segment.

BILL_TO_ADDRESS_3_EXT VARCHAR2(35)

Bill to address line 3, as sent by the customer on the N3 segment.

BILL_TO_ADDRESS_4_EXT VARCHAR2(35)

Bill to address line 4, as sent by the customer on the N3 segment.

BILL_TO_ADDRESS_ID NUMBER

Bill to address identifier. FK to RA_ADDRESSES.

INVOICE_TO_ORG_ID NUMBER

Unique identifier for invoice-to organization which relates to BILL_TO_ADDRESS_ID.
FK in R12 customer-org data model to HR_ORGANIZATIONS.

BILL_TO_CITY_EXT VARCHAR2(30)

Bill to address city, as sent by the customer on the N4 segment.

BILL_TO_COUNTRY_EXT VARCHAR2(3)

Bill to country, as sent by the customer on the N4 segment.

BILL_TO_COUNTY_EXT VARCHAR2(25)

Bill to county, as sent by the customer on the N4 segment.

BILL_TO_NAME_EXT VARCHAR2(60)

Bill to name, as sent by the customer in the N1 and/or N2 segment.

BILL_TO_POSTAL_CD_EXT VARCHAR2(15)

Bill to address postal code, as sent by the customer on the N4 segment.

BILL_TO_PROVINCE_EXT VARCHAR2(10)

Bill to province, as sent by the customer on the N4 segment.

BILL_TO_SITE_USE_ID NUMBER

Bill to site use identifier. FK to RA_SITE_USES.

BILL_TO_STATE_EXT VARCHAR2(10)

Bill to state, as sent by the customer on the N4 segment.

CARRIER_ID_CODE_EXT VARCHAR2(35)

The carrier id, as sent by the customer on the TD5 segment.

CARRIER_QUALIFIER_EXT VARCHAR2(3)

The carrier qualifier, as sent by the customer on the TD5 segment.

COMMODITY_EXT VARCHAR2(35)

Customer specified commodity code.

COUNTRY_OF_ORIGIN_EXT VARCHAR2(35)

Customer-specified country of origin.

CUST_ASSEMBLY_EXT VARCHAR2(30)

The customer's assembly identification, as sent on the LIN segment of the 866 transaction.

CUST_ASSIGNED_ID_EXT VARCHAR2(20)

Customer assigned identification for differentiation within a transaction set, from LIN01.

CUST_BILL_TO_EXT VARCHAR2(35)

External customer bill to cross reference.

CUST_CONTRACT_NUM_EXT VARCHAR2(35)

The customer's contract number, as sent on the LIN segment of the 830 transaction.

CUSTOMER_DOCK_CODE VARCHAR2(50)

Customer dock code.

CUST_INTERMD_SHIP_TO_EXT VARCHAR2(35)

External intermediate ship to cross reference.

CUST_ITEM_PRICE_EXT NUMBER

Price included on the customer's EDI transaction.

CUST_ITEM_PRICE_UOM_EXT VARCHAR2(3)

UOM corresponding to the price included on the customer's EDI transaction.

CUSTOMER_ITEM_REVISION VARCHAR2(35)

Customer part revision included on schedule.

CUSTOMER_JOB VARCHAR2(50)

Customer job number.

CUST_MANUFACTURER_EXT VARCHAR2(35)

The manufacturer, as sent by the customer on the LIN segment of the 866 transaction.

CUST_MODEL_NUMBER_EXT VARCHAR2(35)

Customer's model number for this sequenced detail.

CUST_MODEL_SERIAL_NUMBER VARCHAR2(35)

Customer's vehicle identification number for this sequenced detail, e.g. VIN or Chassis ID.

CUST_ORDER_NUM_EXT VARCHAR2(35)

The order number, as sent by the customer on the LIN segment of the 866 and 830 transaction.

CUST_PROCESS_NUM_EXT VARCHAR2(35)

The process number, as sent by the customer on the LIN segment of the 866 and 862 transaction.

CUST_PRODUCTION_LINE VARCHAR2(50)

Customer production line.

CUSTOMER_PROD_SEQ_NUM VARCHAR2(350)

Customer production sequence number, or delivery number.

CUST_SET_NUM_EXT VARCHAR2(35)

The customer's set number, as sent on the LIN segment of the 862 and 866 transactions.

CUST_SHIP_FROM_ORG_EXT VARCHAR2(80)

External inventory organization cross reference.

CUST_SHIP_TO_EXT VARCHAR2(35)

External ship to address cross reference.

CUST_UOM_EXT VARCHAR2(10)

External customer unit of measure cross reference.

CUSTOMER_ITEM_EXT VARCHAR2(50)

External customer part number cross reference.

CUSTOMER_ITEM_ID NUMBER

Customer item identifier. FK to MTL_CUSTOMER_ITEMS.

REQUEST_DATE DATE

The date and time the customer wants the material delivered or shipped, based on whether the requirements are delivery based or ship based.

SCHEDULE_DATE DATE

Planned shipment date and time (request_date - delivery lead time if delivery based).

DATE_TYPE_CODE VARCHAR2(30)

Type of start/end date, e.g. ship, deliver, Pull signal, Cumulative. Validated against FND_LOOKUPS.LOOKUP_TYPE =RLM_DATE_TYPE_CODE

DATE_TYPE_CODE_EXT VARCHAR2(30)

External value from which DATE_TYPE_CODE was derived in EDI Gateway code conversion.

DELIVERY_LEAD_TIME NUMBER

For demand lines, the difference between arrival and shipment dates, based on default shipping method between the ship-from and ship-to locations. This is determined by the calculate ship date routine and passed into Order Import.

END_DATE_TIME DATE

Customer-specified optional end date/time, applicable for flexible bucketed requirements, and cumulative information such as authorizations and shipped/received.

EQUIPMENT_CODE_EXT VARCHAR2(3)

The equipment code, as sent by the customer on the TD3 segment.

EQUIPMENT_NUMBER_EXT VARCHAR2(35)

The equipment number, as sent by the customer on the TD3 segment.

HANDLING_CODE_EXT VARCHAR2(3)

Special handling code as sent by the customer in the TD4 segment.

HAZARD_CODE_EXT VARCHAR2(10)

Hazardous material code corresponding to the hazardous material code qualifier as sent by the customer in the TD4 segment.

HAZARD_CODE_QUAL_EXT VARCHAR2(3)

Hazardous material code qualifier as sent by the customer in the TD4 segment.

HAZARD_DESCRIPTION_EXT VARCHAR2(80)

Hazardous material description as sent by the customer in the TD4 segment.

IMPORT_LICENSE_DATE_EXT DATE

Customer's import license date.

IMPORT_LICENSE_EXT VARCHAR2(35)

Customer's import license for shipment destination country.

INDUSTRY_ATTR IBUTE1 VARCHAR2(150)

Record keeping or model year.

INDUSTRY_ATTRIBUTE10 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE11 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE12 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE13 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE14 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE15 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE2 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE3 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE4 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE5 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE6 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE7 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE8 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_ATTRIBUTE9 VARCHAR2(150)

Industry descriptive flexfield.

INDUSTRY_CONTEXT VARCHAR2(30)

Industry descriptive context flexfield.

INTERMEDIATE_SHIP_TO_ID NUMBER

Customer intermediate ship-to destination unique identifier. FK to RA_ADDRESSES

SHIP_TO_ORG_ID NUMBER

Unique identifier for ship-to organization which relates to SHIP_TO_ADDRESS_ID if there is no intermediate ship-to address, or to INTERMEDIATE_SHIP_TO_ID if intermediate ship-to address is specified. FK in R12 customer-org data model to HR_ORGANIZATIONS.

INTRMD_ST_ADDRESS_1_EXT VARCHAR2(35)

Intermediate ship to address line 1, as sent by the customer on the N3 segment.

INTRMD_ST_ADDRESS_2_EXT VARCHAR2(35)

Intermediate ship to address line 2, as sent by the customer on the N3 segment.

INTRMD_ST_ADDRESS_3_EXT VARCHAR2(35)

Intermediate ship to address line 3, as sent by the customer on the N3 segment.

INTRMD_ST_ADDRESS_4_EXT VARCHAR2(35)

Intermediate ship to address line 4, as sent by the customer on the N3 segment.

INTRMD_ST_CITY_EXT VARCHAR2(30)

Intermediate ship to address city, as sent by the customer on the N4 segment.

INTRMD_ST_COUNTRY_EXT VARCHAR2(3)

Intermediate ship to country, as sent by the customer on the N4 segment.

INTRMD_ST_COUNTY_EXT VARCHAR2(25)

Intermediate ship to county, as sent by the customer on the N4 segment.

INTRMD_ST_NAME_EXT VARCHAR2(60)

Intermediate ship to name, as sent by the customer in the N1 and/or N2 segment.

INTRMD_ST_POSTAL_CD_EXT VARCHAR2(15)

Intermediate ship to address postal code, as sent by the customer on the N4 segment.

INTRMD_ST_PROVINCE_EXT VARCHAR2(10)

Intermediate ship to province, as sent by the customer on the N4 segment.

INTRMD_ST_STATE_EXT VARCHAR2(10)

Intermediate ship to state, as sent by the customer on the N4 segment.

INTRMD_ST_SITE_USE_ID NUMBER

Intermediate ship to site use identifier, FK to RA_SITE_USES.

INVENTORY_ITEM_ID NUMBER

Inventory item identifier. FK to MTL_SYSTEM_ITEMS.

INVENTORY_ITEM_SEGMENT1 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT10 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT11 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT12 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT13 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT14 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT15 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT16 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT17 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT18 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT19 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT2 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT20 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT3 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT4 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT5 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT6 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT7 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT8 VARCHAR2(40)

Accounting flexfield.

INVENTORY_ITEM_SEGMENT9 VARCHAR2(40)

Accounting flexfield.

ITEM_CONTACT_CODE_1 VARCHAR2(3)

Contact code 1 included in customer's EDI transaction.

ITEM_CONTACT_CODE_2 VARCHAR2(3)

Contact code 2 included in customer's EDI transaction.

ITEM_CONTACT_VALUE_1 VARCHAR2(80)

Values associated with contact code 1: concatenation of name, communication code, and communication number.

ITEM_CONTACT_VALUE_2 VARCHAR2(80)

Values associated with contact code 2 : concatenation of name, communication code, and communication number.

ITEM_DESCRIPTION_EXT VARCHAR2(80)

Item description included on schedule.

ITEM_DETAIL_QUANTITY NUMBER

Requested quantity.

If Item_Detail_Type = 0,1,2, this is the demand quantity.

If Item_Detail_Type = 3, this is the authorization quantity.

If Item_Detail_Type = 4, this is the shipped, received or cum quantity.

If Item_Detail_Type = 5, this is a miscellaneous quantity, such as ahead/behind, inventory balance, etc.

ITEM_DETAIL_REF_CODE_1 VARCHAR2(3)

Reference code 1 included in customer's EDI transaction.

ITEM_DETAIL_REF_CODE_2 VARCHAR2(3)

Reference code 2 included in customer's EDI transaction.

ITEM_DETAIL_REF_CODE_3 VARCHAR2(3)

Reference code 3 included in customer's EDI transaction.

ITEM_DETAIL_REF_VALUE_1 VARCHAR2(35)

Value associated with reference code 1.

ITEM_DETAIL_REF_VALUE_2 VARCHAR2(35)

Value associated with reference code 2.

ITEM_DETAIL_REF_VALUE_3 VARCHAR2(35)

Value associated with reference code 3.

ITEM_DETAIL_SUBTYPE VARCHAR2(30)

Schedule item detail row sub-type:

for firm or forecast demand, Bucket type associated with the demand date/quantity;

for authorizations, the type of authorization;

for shipment/receipt, cumulative or last.

Validated against FND_LOOKUPS.

ITEM_DETAIL_SUBTYPE_EXT VARCHAR2(30)

External value from which ITEM_DETAIL_SUBTYPE was derived in EDI Gateway Code Conversion.

ITEM_DETAIL_TYPE VARCHAR2(30)

Schedule item detail row type: 0 = Past Due Firm, 1 = Firm Demand, 2 = Forecast Demand, 3 = Authorization, 4 = Shipment/Receipt Info, 5 = Other. Validated against FND_LOOKUPS.LOOKUP_TYPE = RLM_DETAIL_TYPE_CODE

ITEM_DETAIL_TYPE_EXT VARCHAR2(30)

External value from which ITEM_DETAIL_TYPE was derived in EDI Gateway Code Conversion.

ITEM_ENG_CNG_LVL_EXT VARCHAR2(35)

Customer part engineering change level included on schedule

ITEM_MEASUREMENTS_EXT VARCHAR2(240)

Item measurement information as sent by the customer on the MEA segments.

ITEM_NOTE_TEXT VARCHAR2(240)

Free form item note text included in the schedule.

ITEM_REF_CODE_1 VARCHAR2(3)

Reference code 1 included in customer's EDI transaction (not validated).

ITEM_REF_CODE_2 VARCHAR2(3)

Reference code 2 included in customer's EDI transaction (not validated).

ITEM_REF_CODE_3 VARCHAR2(3)

Reference code 3 included in customer's EDI transaction (not validated).

ITEM_REF_VALUE_1 VARCHAR2(35)

Value associated with reference code 1.

ITEM_REF_VALUE_2 VARCHAR2(35)

Value associated with reference code 2.

ITEM_REF_VALUE_3 VARCHAR2(35)

Value associated with reference code 3.

ITEM_RELEASE_STATUS_EXT VARCHAR2(3)

Customer part release status included on schedule.

LADING_QUANTITY_EXT NUMBER

The lading quantity as sent by the customer on the TD1 segment.

LETTER_CREDIT_EXPDT_EXT DATE

Customer's letter of credit expiration date.

LETTER_CREDIT_EXT VARCHAR2(35)

Customer's letter of credit with international bank guaranteeing payment for international shipments.

LINE_REFERENCE VARCHAR2(50)

Unique line identifier within an order for a transmission identifier.

LINK_TO_LINE_REF VARCHAR2(50)

Link to Immediate Parent_line_reference within an order for a transmission identifier.

ORDER_HEADER_ID NUMBER

Order header identifier. FK to OE_ORDER_HEADERS.

ORG_ID NUMBER

Operating unit unique id. FK to HR_ORGANIZATION_UNITS.

OTHER_NAME_CODE_1 VARCHAR2(3)

Other name code 1 included in customer's EDI transaction (not validated).

OTHER_NAME_CODE_2 VARCHAR2(3)

Other name code 2 included in customer's EDI transaction (not validated).

OTHER_NAME_VALUE_1 VARCHAR2(80)

Values associated with other name code 1: concatenation of name, code, and number.

OTHER_NAME_VALUE_2 VARCHAR2(80)

Values associated with other name code 2: concatenation of name, code, and number.

PACK_SIZE_EXT NUMBER

Size of supplier units in pack, as sent by the customer on the PO4 segment.

PACK_UNITS_PER_PACK_EXT NUMBER

Number of inner pack units per out pack unit, as sent by the customer on the PO4 segment.

PACK_UOM_CODE_EXT VARCHAR2(3)

Unit of measure of supplier units in the pack, as sent by the customer in the PO4 segment.

PACKAGING_CODE_EXT VARCHAR2(10)

The packaging code as sent by the customer on TD1 segment.

PARENT_LINK_LINE_REF VARCHAR2(50)

Line reference of top model for a transmission identifier for an order.

PRICE_LIST_ID NUMBER

Unique identifier for price list associated with customer item or agreement on which customer purchase order is associated. FK to RA_PRICE_LISTS

PRIMARY_QUANTITY NUMBER

Quantity in the primary UOM.

PRIMARY_UOM_CODE VARCHAR2(3)

Primary unit of measure. FK to MTL_UNITS_OF_MEASURE

PRIME_CONTRACTOR_PART_EXT VARCHAR2(35)

The prime contractor part number, as sent by the customer on the LIN segment of the 862 segment.

PROCESS_STATUS NUMBER

Indicates the current processing status of a the demand line

1= Waiting to be processed

2= Processed with error

3= Processed successfully

4= Processed with error(s) for headers with lines in-error

CUST_PO_RELEASE_NUM VARCHAR2(35)

Customer purchase order release number included on schedule.

CUST_PO_DATE DATE

Customer purchase order line number included on schedule.

CUST_PO_LINE_NUM VARCHAR2(35)

Customer-specified effectivity date of purchase order number included on schedule.

CUST_PO_NUMBER VARCHAR2(50)

Customer purchase order number.

QTY_TYPE_CODE VARCHAR2(30)

Actual or Cumulative. Validated against FND_LOOKUPS.LOOKUP_TYPE
=RLM_QTY_TYPE_CODE

QTY_TYPE_CODE_EXT VARCHAR2(30)

External value from which QTY_TYPE_CODE was derived in EDI Gateway Code
Conversion.

RETURN_CONTAINER_EXT VARCHAR2(35)

Returnable container specified by customer for item shipment.

RLM_SCHEDULE_LINE_ID NUMBER

Schedule line unique identifier. FK to RLM_SCHEDULE_LINES_ALL, except for
aggregated schedule lines.

ROUTING_DESC_EXT VARCHAR2(35)

The routing description, as sent by the customer on the TD5 segment.

ROUTING_SEQ_CODE_EXT VARCHAR2(3)

Code describing the relationship of a carrier to a specific shipment movement, as sent
by the customer on the TD5 segment.

SCHEDULE_ITEM_NUM NUMBER

Schedule Item Number, the means to identify how item demand and information is
grouped by the customer within the schedule. For sequenced schedules, it is equal to
CUSTOMER_PROD_SEQ_NUM. For EDI planning and shipping schedules, it is
incremented in the EDI Gateway when each 2000 record is encountered on an inbound
SPSI or SSSI transaction. This number is assigned for manually entered schedules. All
interface lines with the same schedule item number are validated together, and pass or
fail validation as a group.

SHIP_DEL_PATTERN_EXT VARCHAR2(3)

Customer ship delivery pattern code for this item; not integrated with delivery rules for
date/quantity calculation.

SHIP_DEL_TIME_CODE_EXT VARCHAR2(3)

Customer ship delivery time code for this item; not integrated with delivery rules for
date/quantity calculation.

SHIP_DEL_RULE_NAME VARCHAR2(30)

Shipment/delivery rule name for this schedule item. Initially populated only if a
successful code conversion in the EDI Gateway has occurred. FK to
RLM_SHIP_DELIVERY_CODES

SHIP_FROM_ADDRESS_1_EXT VARCHAR2(35)

Ship from address line 1, as sent by the customer on the N3 segment.

SHIP_FROM_ADDRESS_2_EXT VARCHAR2(35)

Ship from address line 2, as sent by the customer on the N3 segment.

SHIP_FROM_ADDRESS_3_EXT VARCHAR2(35)

Ship from address line 3, as sent by the customer on the N3 segment.

SHIP_FROM_ADDRESS_4_EXT VARCHAR2(35)

Ship from address line 4, as sent by the customer on the N3 segment.

SHIP_FROM_CITY_EXT VARCHAR2(30)

Ship from address city, as sent by the customer on the N4 segment.

SHIP_FROM_COUNTRY_EXT VARCHAR2(3)

Ship from country, as sent by the customer on the N4 segment.

SHIP_FROM_COUNTY_EXT VARCHAR2(25)

Ship from county, as sent by the customer on the N4 segment.

SHIP_FROM_NAME_EXT VARCHAR2(60)

Ship from name, as sent by the customer in the N1 or N2 segment.

SHIP_FROM_ORG_ID NUMBER

Ship from organization identifier. FK to MTL_PARAMETERS and HR_ORGANIZATIONS.

SHIP_FROM_POSTAL_CD_EXT VARCHAR2(15)

Ship from address postal code, as sent by the customer on the N4 segment.

SHIP_FROM_PROVINCE_EXT VARCHAR2(10)

Ship from province, as sent by the customer on the N4 segment.

SHIP_FROM_STATE_EXT VARCHAR2(10)

Ship from state, as sent by the customer on the N4 segment.

SHIP_LABEL_INFO_LINE_1 VARCHAR2(80)

Pull signal bar-code label routing information - line 1.

SHIP_LABEL_INFO_LINE_10 VARCHAR2(80)

Pull signal bar-code label routing information - line 10.

SHIP_LABEL_INFO_LINE_2 VARCHAR2(80)

Pull signal bar-code label routing information - line 2.

SHIP_LABEL_INFO_LINE_3 VARCHAR2(80)

Pull signal bar-code label routing information - line 3.

SHIP_LABEL_INFO_LINE_4 VARCHAR2(80)

Pull signal bar-code label routing information - line 4.

SHIP_LABEL_INFO_LINE_5 VARCHAR2(80)

Pull signal bar-code label routing information - line 5.

SHIP_LABEL_INFO_LINE_6 VARCHAR2(80)

Pull signal bar-code label routing information - line 6.

SHIP_LABEL_INFO_LINE_7 VARCHAR2(80)

Pull signal bar-code label routing information - line 7.

SHIP_LABEL_INFO_LINE_8 VARCHAR2(80)

Pull signal bar-code label routing information - line 8.

SHIP_LABEL_INFO_LINE_9 VARCHAR2(80)

Pull signal bar-code label routing information - line 9.

SHIP_TO_ADDRESS_1_EXT VARCHAR2(35)

Ship to address line 1, as sent by the customer on the N3 segment.

SHIP_TO_ADDRESS_2_EXT VARCHAR2(35)

Ship to address line 2, as sent by the customer on the N3 segment.

SHIP_TO_ADDRESS_3_EXT VARCHAR2(35)

Ship to address line 3, as sent by the customer on the N3 segment.

SHIP_TO_ADDRESS_4_EXT VARCHAR2(35)

Ship to address line 4, as sent by the customer on the N3 segment.

SHIP_TO_ADDRESS_ID NUMBER

Ship to address identifier. FK to RA_ADDRESSES.

DELIVER_TO_ORG_ID NUMBER

Unique identifier for deliver-to organization which relates to SHIP_TO_ADDRESS_ID.
FK in R12 customer-org data model to HR_ORGANIZATIONS

SHIP_TO_CITY_EXT VARCHAR2(30)

Ship to address city, as sent by the customer on the N4 segment.

SHIP_TO_COUNTRY_EXT VARCHAR2(3)

Ship to country, as sent by the customer on the N4 segment.

SHIP_TO_COUNTY_EXT VARCHAR2(25)

Ship to county, as sent by the customer on the N4 segment.

SHIP_TO_NAME_EXT VARCHAR2(60)

Ship to name, as sent by the customer on the N1 or N2 segment.

SHIP_TO_POSTAL_CD_EXT VARCHAR2(15)

Ship to address postal code, as sent by the customer on the N4 segment.

SHIP_TO_PROVINCE_EXT VARCHAR2(10)

Ship to province, as sent by the customer on the N4 segment.

SHIP_TO_SITE_USE_ID NUMBER

Ship to site use identifier. FK to RA_SITE_USES.

SHIP_TO_STATE_EXT VARCHAR2(10)

Ship to state, as sent by the customer on the N4 segment.

START_DATE_TIME DATE

Customer-specified date and time, as transmitted by the customer on the EDI transaction.

SUBLINE_ASSIGNED_ID_EXT VARCHAR2(20)

Subline customer assigned identification from the SLN01. Related to but not necessarily equivalent to the baseline number, assigned identification from LIN01. For example, 1.1 or 1A might be used as a subline number to relate to baseline number 1.

SUBLINE_CONFIG_CODE_EXT VARCHAR2(3)

The Subline Configuration Code, as sent by the customer on the SLN segment, indicating the relationship of the subline item to the baseline item.

SUBLINE_CUST_ITEM_EXT VARCHAR2(50)

Subline customer item number, as sent by the customer on the SLN segment.

SUBLINE_CUST_ITEM_ID NUMBER

Customer item unique identifier. FK to RLM_SHIP_FROM_CUST_ITEM.

SUBLINE_MODEL_NUM_EXT VARCHAR2(35)

Subline customer model number, as sent by the customer on the SLN segment.

SUBLINE_QUANTITY NUMBER

The subline quantity, as sent by the customer on the SLN segment.

SUBLINE_UOM_CODE VARCHAR2(3)

The subline unit of measure (internal) cross-referenced from the UOM sent by the customer on the SLN segment. FK to MTL_UNITS_OF_MEASURE.

SUPPLIER_ITEM_EXT VARCHAR2(35)

Supplier item number specified by customer.

TRANSIT_TIME_EXT VARCHAR2(22)

The transit time, as sent by the customer on the TD5 segment.

TRANSIT_TIME_QUAL_EXT VARCHAR2(3)

The transit time qualifier, as sent by the customer on the TD5 segment.

TRANSPORT_LOC_QUAL_EXT VARCHAR2(3)

The shipping location qualifier, as sent by the customer on the TD5 segment. This identifies the type of location which specified in the corresponding shipping location.

TRANSPORT_LOCATION_EXT VARCHAR2(35)

The specific shipping location (such as pool point or airport) corresponding to the shipping location qualifier, as sent by the customer on the TD5 segment. This corresponds to the shipping location qualifier.

TRANSPORT_METHOD_EXT VARCHAR2(3)

The transportation method, as sent by the customer on the TD5 segment.

UOM_CODE VARCHAR2(3)

Abbreviated unit of measure code. FK to MTL_UNITS_OF_MEASURE.

WEIGHT_EXT NUMBER

The weight, as sent by the customer on the TD1 segment.

WEIGHT_QUALIFIER_EXT VARCHAR2(3)

The weight qualifier, as sent by the customer on the TD1 segment.

WEIGHT_UOM_EXT VARCHAR2(3)

The unit of measure corresponding to shipment weight, as sent by the customer on the TD1 segment.

FBO_CONFIGURATION_KEY_1 VARCHAR2(35)

For FBO Production Sequence schedules, the 1st sort key for identifying configurations indicated by the customer.

FBO_CONFIGURATION_KEY_2 VARCHAR2(35)

For FBO Production Sequence schedules, the 2nd sort key for identifying configurations indicated by the customer.

FBO_CONFIGURATION_KEY_3 VARCHAR2(35)

For FBO Production Sequence schedules, the 3rd sort key for identifying configurations indicated by the customer.

FBO_CONFIGURATION_KEY_4 VARCHAR2(35)

For FBO Production Sequence schedules, the 4th sort key for identifying configurations indicated by the customer.

FBO_CONFIGURATION_KEY_5 VARCHAR2(35)

For FBO Production Sequence schedules, the 5th sort key for identifying configurations indicated by the customer.

MATCH_ACROSS_KEY VARCHAR2(150)

Audit trail of match across key used when this schedule was processed.

MATCH_WITHIN_KEY VARCHAR2(150)

Audit trail of match within key used when this schedule was processed.

CRITICAL_KEY_ATTRIBUTES VARCHAR2(150)

Audit trail of critical attributes key used when this schedule was processed. An exception (warning) is generated if one of these is missing on Firm demand.

LAST_UPDATE_DATE DATE

Standard Who column.

LAST_UPDATED_BY NUMBER

Standard Who column.

CREATION_DATE DATE

Standard Who column.

CREATED_BY NUMBER

Standard Who column.

ATTRIBUTE_CATEGORY VARCHAR2(30)

Descriptive flexfield context column.

ATTRIBUTE1 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE2 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE3 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE4 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE5 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE6 VARCHAR2(150)

Descriptive flexfield segment column

ATTRIBUTE7 VARCHAR2(150)

Descriptive flexfield segment column.

ATTRIBUTE8 VARCHAR2(150)

Descriptive flexfield segment column.
 ATTRIBUTE9 VARCHAR2(150)
 Descriptive flexfield segment column.
 ATTRIBUTE10 VARCHAR2(150)
 Descriptive flexfield segment column.
 ATTRIBUTE11 VARCHAR2(150)
 Descriptive flexfield segment column.
 ATTRIBUTE12 VARCHAR2(150)
 Descriptive flexfield segment column.
 ATTRIBUTE13 VARCHAR2(150)
 Descriptive flexfield segment column.
 ATTRIBUTE14 VARCHAR2(150)
 Descriptive flexfield segment column
 ATTRIBUTE15 VARCHAR2(150)
 Descriptive flexfield segment column.
 LAST_UPDATE_LOGIN NUMBER
 Standard Who column.
 REQUEST_ID NUMBER
 Standard Who column.
 PROGRAM_APPLICATION_ID NUMBER
 Standard Who column
 PROGRAM_ID NUMBER
 Standard Who column.
 PROGRAM_UPDATE_DATE DATE
 Standard Who column.
 TP_ATTRIBUTE1 VARCHAR2(150)
 Trading partner flexfield segment column.
 TP_ATTRIBUTE2 VARCHAR2(150)
 Trading partner flexfield segment column.
 TP_ATTRIBUTE3 VARCHAR2(150)
 Trading partner flexfield segment column.
 TP_ATTRIBUTE4 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE5 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE6 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE7 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE8 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE9 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE10 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE11 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE12 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE13 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE14 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE15 VARCHAR2(150)

Trading partner flexfield segment column.

TP_ATTRIBUTE_CATEGORY VARCHAR2(30)

Trading partner flexfield context column.

LINE_NUMBER NUMBER

Unique identifier of a line within a schedule.

INTMED_SHIP_TO_ORG_ID NUMBER

Unique identifier for intermediate ship-to organization which relates to INTERMEDIATE_SHIP_TO_ID if intermediate ship-to address is specified. FK to HR_ORGANIZATIONS

LINE_SOURCE VARCHAR2(30)

Schedule Source at line level since lines can also be manually entered.

PREFERRED_GRADE VARCHAR2(4)

Preferred grade.

CUST_PRODUCTION_SEQ_NBR_BEG VARCHAR2(35)

For future use.

CUST_PRODUCTION_SEQ_NBR_END VARCHAR2(35)

For future use.

ITEM_DETAIL_QUANTITY_MIN_EXT NUMBER

For future use.

ITEM_DETAIL_QUANTITY_MAX_EXT NUMBER

For future use.

ITEM_DETAIL_QUANTITY_PRIOR_EXT NUMBER

For future use.

REQUIREMENT_PRIORITY_EXT NUMBER

For future use.

PROJECT_NUMBER_EXT VARCHAR2(50)

For future use.

RESP_LINE_NUMBER VARCHAR2(35)

For future use.

TASK_NUMBER_EXT VARCHAR2(50)

For future use.

UPC_EXT VARCHAR2(50)

For future use.

ITEM_ENG_CNG_LVL_DATE_EXT DATE

For future use.

AUTHORIZATION_CODE_EXT VARCHAR2(50)

For future use.

PACKAGING_DESC_EXT VARCHAR2(80)

For future use.

BILL_TO_ADDRESS_5_EXT VARCHAR2(35)

For future use.

BILL_TO_ADDRESS_6_EXT VARCHAR2(35)

For future use.

BILL_TO_ADDRESS_7_EXT VARCHAR2(35)

For future use.

BILL_TO_ADDRESS_8_EXT VARCHAR2(35)

For future use.

BILL_TO_ADDRESS_9_EXT VARCHAR2(35)

For future use.

ONE_TIME_BILL_TO_FLAG VARCHAR2(35)

For future use.

BILL_TO_DESCRIPTION_EXT VARCHAR2(50)

For future use.

BILL_TO_FAX_EXT VARCHAR2(35)

For future use.35

BILL_TO_REGION_EXT VARCHAR2(35)

For future use.

BILL_TO_JURISDICTION_EXT VARCHAR2(35)

For future use.

BILL_TO_TEL_EXT VARCHAR2(35)

For future use.

INTRMD_ST_ADDRESS_5_EXT VARCHAR2(35)

For future use.

INTRMD_ST_ADDRESS_6_EXT VARCHAR2(35)

For future use.

INTRMD_ST_ADDRESS_7_EXT VARCHAR2(35)

For future use.

INTRMD_ST_ADDRESS_8_EXT VARCHAR2(35)

For future use.

INTRMD_ST_ADDRESS_9_EXT VARCHAR2(35)

For future use.

ONE_TIME_INTRMD_ST_FLAG_EXT VARCHAR2(35)

For future use.

INTRMD_ST_DESCRIPTION_EXT VARCHAR2(50)

For future use.

INTRMD_ST_FAX_EXT VARCHAR2(35)

For future use.

INTRMD_ST_REGION_EXT VARCHAR2(35)

For future use.

INTRMD_ST_JURISDICTION_EXT VARCHAR2(35)

For future use.

INTRMD_ST_TEL_EXT VARCHAR2(35)

For future use.

SHIP_TO_ADDRESS_5_EXT VARCHAR2(35)

For future use.

SHIP_TO_ADDRESS_6_EXT VARCHAR2(35)

For future use.

SHIP_TO_ADDRESS_7_EXT VARCHAR2(35)

For future use.

SHIP_TO_ADDRESS_8_EXT VARCHAR2(35)

For future use.

SHIP_TO_ADDRESS_9_EXT VARCHAR2(35)

For future use.

ONE_TIME_SHIP_TO_FLAG_EXT VARCHAR2(35)

For future use.

SHIP_TO_DESCRIPTION_EXT VARCHAR2(50)

For future use.

SHIP_TO_FAX_EXT VARCHAR2(35)

For future use.

SHIP_TO_REGION_EXT VARCHAR2(35)

For future use.

SHIP_TO_JURISDICTION_EXT VARCHAR2(35)

For future use.

SHIP_TO_TEL_EXT VARCHAR2(35)

For future use.

SHIP_FROM_ADDRESS_5_EXT VARCHAR2(35)

For future use.

SHIP_FROM_ADDRESS_6_EXT VARCHAR2(35)

For future use.

SHIP_FROM_ADDRESS_7_EXT VARCHAR2(35)

For future use.

SHIP_FROM_ADDRESS_8_EXT VARCHAR2(35)

For future use.

SHIP_FROM_ADDRESS_9_EXT VARCHAR2(35)

For future use.

ONE_TIME_SHIP_FROM_FLAG_EXT VARCHAR2(35)

For future use.

SHIP_FROM_DESCRIPTION_EXT VARCHAR2(50)

For future use.

SHIP_FROM_FAX_EXT VARCHAR2(35)

For future use.

SHIP_FROM_REGION_EXT VARCHAR2(35)

For future use.

SHIP_FROM_JURISDICTION_EXT VARCHAR2(35)

For future use.

SHIP_FROM_TEL_TEXT VARCHAR2(35)

For future use

Oracle Shipping Execution Open Interfaces and APIs

This chapter covers the following topics:

- Overview of API Information
- Shipment Processing Using APIs
- Actions, APIs, and Parameters
- Application Parameter Initialization
- Trip Public Application Program Interface
- Stop Public Application Program Interface
- Deliveries Public Application Program Interface
- Exceptions Application Program Interface
- Delivery Details Public Application Program Interface
- Container Public Application Program Interface
- Freight Costs Public Application Program Interface
- Pick Release Application Program Interface
- Document Public Application Program Interface
- Custom Public Application Program Interface
- Example of API Packages and Procedures

Overview of API Information

This chapter contains the following information about Oracle Shipping Execution Public Application Program Interfaces (APIs):

- Shipment Processing Using APIs: Correlation between common Shipping

Transaction Form tasks and Public APIs and some sample scenarios that you can use to process shipments through ship confirmation using APIs.

- API Package and Procedures Example: An example of two simple procedures in the same package that use the APIs.
- Public Application Program Interfaces: Descriptions of the APIs.
 - Actions, APIs, and Parameters: Descriptions of the APIs used for various functions and the API parameters.
 - Application Parameter Initialization: Description of the application parameter initialization call.
 - Trip API: Create and update trip records and perform actions on trips.
 - Exceptions API: Retrieve information on exceptions and carry out actions on exceptions.
 - Stop API: Create and update stop records and perform actions on stops.
 - Deliveries API: Create and update trip stop records and perform actions on trip stops.
 - Delivery Details API: Assign and unassign delivery details to and from deliveries, split a delivery detail, update a delivery detail with new information, and create trips and deliveries for multiple delivery lines.
 - Container API: Create container records, update container records, autopack containers, perform actions on containers.
 - Freight Cost APIs: Create freight cost records, update freight cost records, validate freight cost types, delete freight cost records.
 - Picking API: Creates a record of all pick release batches.
 - Custom API: Customize delivery window calculations and tailor how the location information is displayed in the Oracle Shipping Execution forms and reports.
 - Document API: Creates a record of the attributes of an open document, such as a bill of lading or packing slip.

Note: For details about migrating your shipping process from the Release 10.7 and Release 11 delivery-based Ship Confirm Open Interface to the Release 12 public APIs, see *OracleMetaLink* note

Shipment Processing Using APIs

This section contains:

- The Shipping Transaction Form/Public API Correlation table: This table specifies the public APIs that correspond to common actions that you perform on the Shipping Transactions Form.
- Sample flow scenarios: These scenarios show different paths to ship confirmation using APIs. Each step in the flow refers to a reference number (Ref) in the API Table.

Note: As a general rule, if an attribute is not updatable from the Shipping Transactions Form, then it is not updateable from the public API. If you pass a value for a non-updateable attribute, then the code ignores the value and the existing value in the database is retained.

The shipping flow scenarios refer to the reference numbers shown in this table.

Shipping Transaction Form/Public API Correlation

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
1	Select delivery lines for processing	Find window > Search for Lines > Enter search criteria	Not a specific API. To do this, populate API input parameter tables before submitting the API.	-	-
2.1	Create trips	Data Entry, Trip Data Entry > Enter data	WSH_TRIPS_PUB. CREATE_UP DATE_TRIP	CREATE	If known, provide information such as trip name, ship method, carrier ID, and vehicle number.

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
2.2	Update trips	Find window > Search for Trips > Enter trip name > Add/change/delete data	WSH_TRIPS_PUB. CREATE_UP DATE_TRIP	UPDATE	If known, provide vehicle type and vehicle number
2.3	Perform actions on trips	Find window > Search for Trips > Select action	WSH_TRIPS_PUB.TRIP_ACTION	PLAN, UNPLAN, DELETE, WT-VOL, PICK- RELEASE, PRINT-DOC-S ETS, FIRM, TRIP-CONFIRM, PRINT-MBOL, UNASSIGN-ALL	-
3.1	Create stops	Data Entry, Stop Data Entry > Enter data	WSH_TRIP_STOPS_PUB. CREATE_UP DATE_STOP	CREATE	Provide trip ID or trip name, stop location ID or stop location code, planned departure date, and planned arrival date.
3.2	Update stops	Find window > Search for Stops > Add/change/delete data	WSH_TRIP_STOPS_PUB. CREATE_UP DATE_STOP	UPDATE	Provide change information, for example, planned dates.

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
3.3	Perform actions on stops	Find window > Search for Stops > Select action	WSH_TRIP_S TOPS_PUB. STOP_ACTIO N	PLAN, UNPLAN, ARRIVE, CLOSE, DELETE, PICK- RELEASE, WT-VOL	-
4.1	Create deliveries	Data Entry, Delivery Data Entry > Enter data	WSH_DELIV ERIES_PUB.C REATE_UPD ATE_DELIVE RIES	CREATE	Provide organization, initial ship-from, and ultimate ship-to information.
4.2	Update deliveries	Find window > Search for Deliveries > Add/change / delete data	WSH_DELIV ERIES_PUB. CREATE_UP DATE_DELIV ERIES	UPDATE	-

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
4.3	Perform actions on deliveries	Find window > Search for Deliveries > Select action	WSH_DELIVERIES_PUB.DELIVERY_ACTION	PLAN, UNPLAN, DELETE, WT-VOL, PICK-RELEASE, RE-OPEN, CLOSE, PICK-PACK-SHIP, OUTBOUND-DOCUMENT, PRINT-DOCUMENTS, FIRM, AUTO-PACK, AUTO-PACK-MASTER, GEN-LOAD-SEQU, SELECT-CARRIER, CANCEL-SHIP-METHOD, PRINT-PACK-SLIP, PRINT-BOL	-
4.4	Assign and cancel the assignment of deliveries to and from trips and stops	Find window > Search for Deliveries > Select action	WSH_DELIVERIES_PUB.DELIVERY_ACTION	ASSIGN-TRIP, UNASSIGN-TRIP, AUTOCREATE-TRIP	For ASSIGN-TRIP, provide a trip name.
4.5	Ship confirmed deliveries	Find window > Search for Deliveries > Action: Ship Conform	WSH_DELIVERIES_PUB.DELIVERY_ACTION	CONFIRM	Provide the actual ship date.

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
4.6	Pick and Ship confirm delivery	Find window > Search for Deliveries > Action: Pick and Ship	WSH_DELIVERIES_PUB.DELIVERY_ACTION	PICK-SHIP	-
5.1	Assign and cancel the assignment of delivery lines to and from deliveries	Find window > Search for Lines > Enter delivery line number > Select action	WSH_DELIVERY_DETAIL_S_PUB.DETAIL_TO_DELIVERY	ASSIGN, UNASSIGN	Provide a delivery ID or delivery name.
5.2	Split delivery lines	Find window > Search for Lines > Enter delivery line number > Action: Split Line	WSH_DELIVERY_DETAIL_S_PUB.SPLIT_LINE	-	Provide a split quantity.
5.3	Confirm actual shipped quantities	Find window > Search for Lines > Enter Shipped Quantity and add/change/delete data	WSH_DELIVERY_DETAIL_S_PUB.UPDATE_SHIPPING_ATTRIBUTES	UPDATE	-

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
5.4	Create deliveries automatically	Find window > Search for Lines > Enter criteria > Action: Auto-create Delivery	WSH_DELIVERY_DETAIL_S_PUB. AUTOCREATE_DELIVERIES	-	-
5.5	Create deliveries and trips automatically	Find window > Search for Lines > Enter criteria > Action: Auto-create Trip	WSH_DELIVERY_DETAIL_S_PUB. AUTOCREATE_DEL_TRIP	-	-
6.1	Create containers	Data Entry, LPN Data Entry > Enter data	WSH_CONTAINER_AINER_PUB. CREATE_CONTAINER	-	Provide the container item ID (prefix, base, and suffix)
6.2	Update containers	Find window > Search for Lines/LPN > Add/change/delete data	WSH_CONTAINER_AINER_PUB. UPDATE_CONTAINER	-	-
6.3	Pack and unpack delivery lines into and from containers	Find window > Search for Lines > Enter delivery line number > Select action > Select target LPN	WSH_CONTAINER_AINER_PUB. CONTAINER_ACTIONS	PACK, UNPACK	For PACK, provide the container ID or the container name and delivery line ID.

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
6.4	Assign and cancel the assignment of containers to and from deliveries	Find window > Search for LPN > Enter LPN > Select action	WSH_CONTAINER_PUB. CONTAINER_ACTIONS	ASSIGN, UNASSIGN	-
6.5	Create containers and auto-pack	Find window > Search for Lines > Enter delivery line number > Action: Auto-pack	WSH_CONTAINER_PUB. AUTO_PACK	-	Provide delivery line ID
6.6	Enter serial numbers for serial controlled item at SO Issue	Find window > Search for Lines > Go to Tools > Serial Numbers > Enter the Serial Number(s) > Save	WSH_DELIVERY_DETAIL_PUB. UPDATE_SHIPPING_ATTRIBUTES (Overloaded)	UPDATE	Provide delivery line ID and Serial Number(s).

Ref	Function	STF Equivalent	Package. Procedure	API Action Codes	Data Requirements
7.1	Create and update freight costs	Find window > Search for entity (Trip, Stop, Delivery, or Delivery Line) > Enter entity name> Action: Freight Cost > Enter or add/change/delete data	WSH_FREIG HT_COSTS_P UB.CREATE_ UPDATE_FR EIGHT_COST	CREATE, UPDATE	For trip, stop, delivery, and delivery line, provide the entity name or entity ID.
7.2	Validate freight costs	Find window > Search for entity (Trip, Stop, Delivery, or Delivery Line) > Enter entity name> Action: Freight Cost > Open Freight Name LOV and select one	WSH_FREIG HT_COSTS_P UB. VALIDATE_F REIGHT_COS T_TYPE	-	-
7.3	Delete freight costs	Find window > Search for entity (Trip, Stop, Delivery, or Delivery Line) > Enter entity name> Action: Freight Cost > Delete row	WSH_FREIG HT_COSTS_P UB.DELETE_ FREIGHT_CO STS	-	-

Sample Flow Scenarios

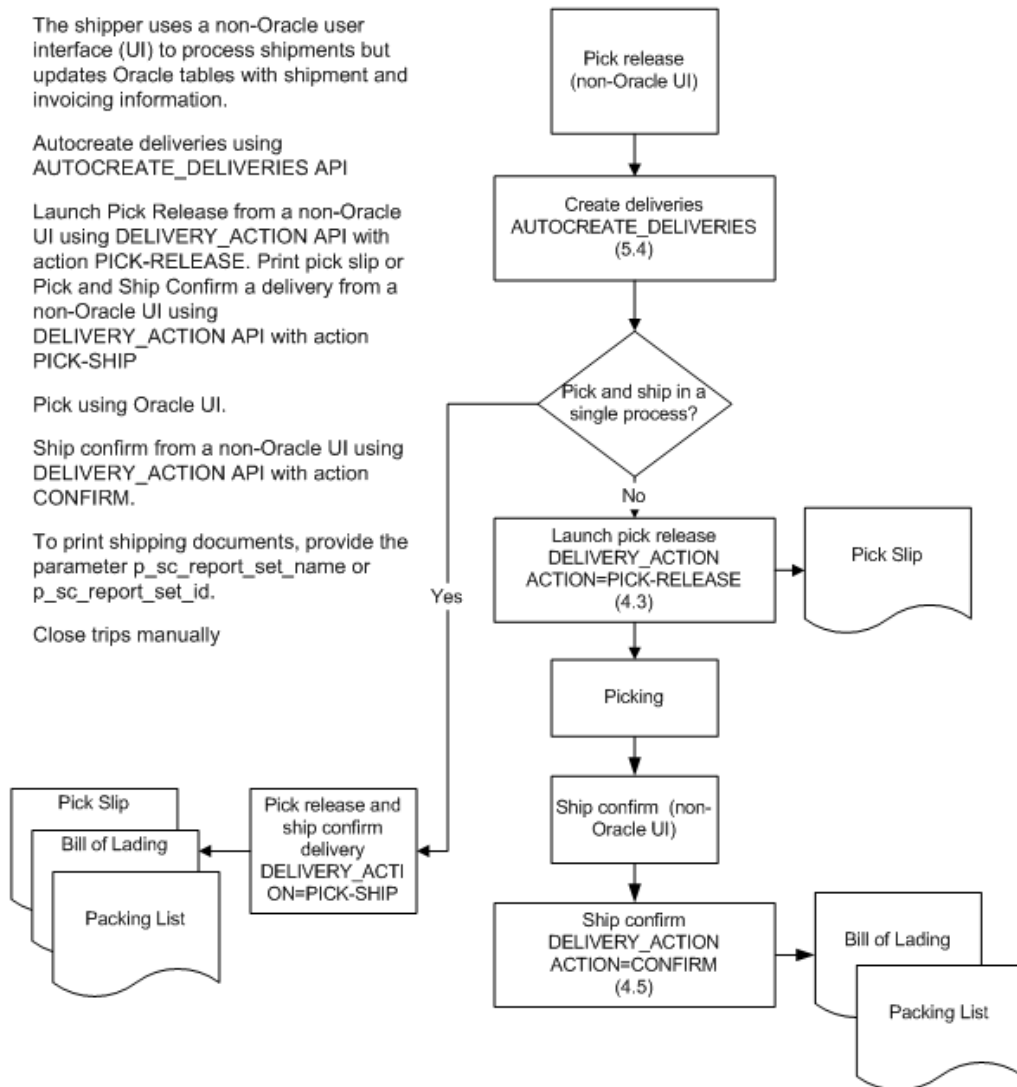
The following scenarios show the different paths to ship confirmation using APIs. Each step in a flow refers to a reference number in the API Table:

- Scenario 1: Pick Release, Ship Confirm/Pick Release with Auto-Ship Confirm as a rule.
- Scenario 2: Pick Release, Ship Confirm with Serial Numbers (for single and multiple serial numbers)
- Scenario 3: Create Delivery, Auto-create Trip, Pick Release Trip
- Scenario 4: Create Trip, Stop, and Delivery; Assign Entities; Pack Items into Containers
- Scenario 5: Auto-create Delivery and Trip, Auto-pack Lines/Manual Pack, Add Freight Costs

Shipping API Flow Scenario 1

Pick Release and Ship Confirm Without Using User Interface

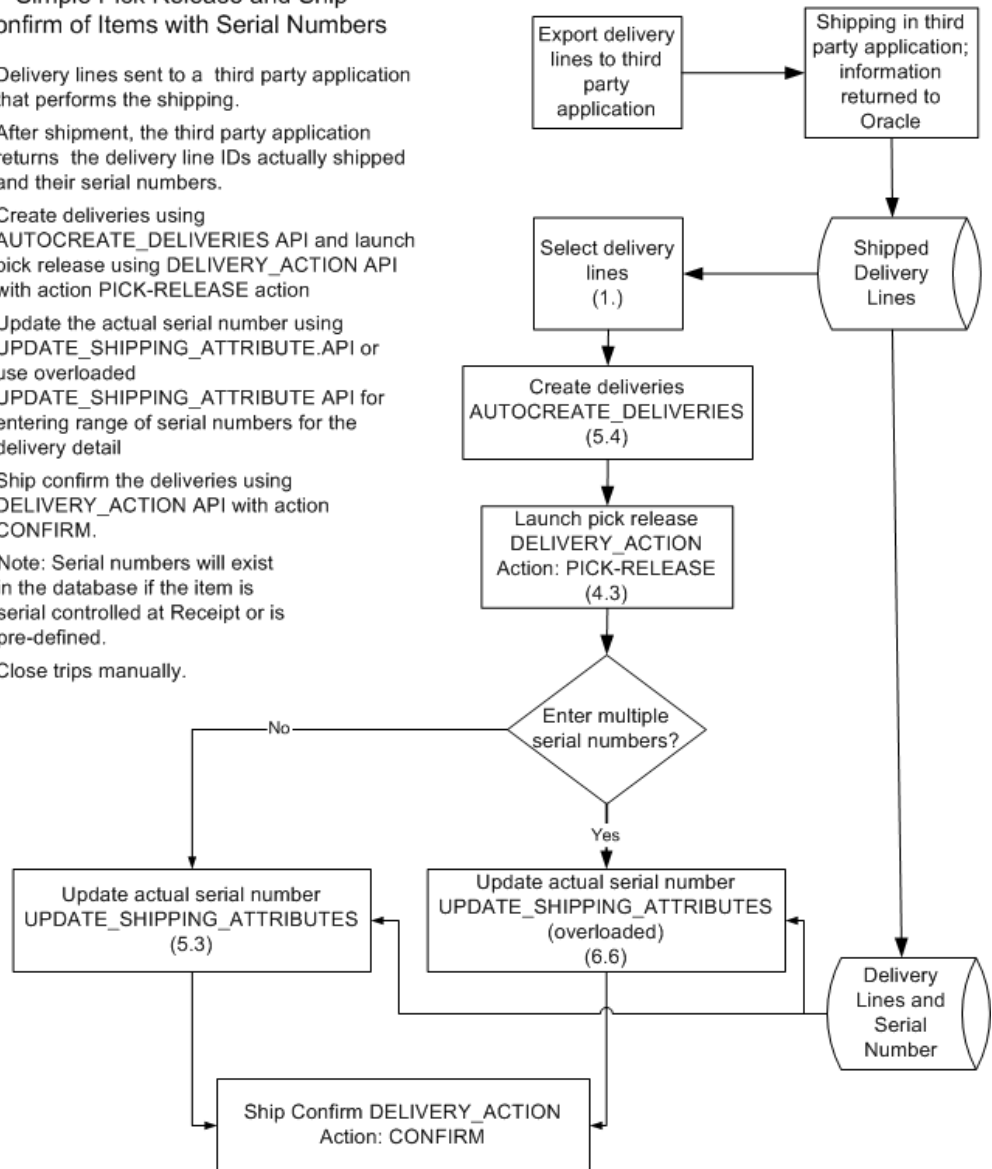
- The shipper uses a non-Oracle user interface (UI) to process shipments but updates Oracle tables with shipment and invoicing information.
- Autocreate deliveries using AUTOCREATE_DELIVERIES API
- Launch Pick Release from a non-Oracle UI using DELIVERY_ACTION API with action PICK-RELEASE. Print pick slip or Pick and Ship Confirm a delivery from a non-Oracle UI using DELIVERY_ACTION API with action PICK-SHIP
- Pick using Oracle UI.
- Ship confirm from a non-Oracle UI using DELIVERY_ACTION API with action CONFIRM.
- To print shipping documents, provide the parameter p_sc_report_set_name or p_sc_report_set_id.
- Close trips manually



Shipping API Flow Scenario 2

Simple Pick Release and Ship Confirm of Items with Serial Numbers

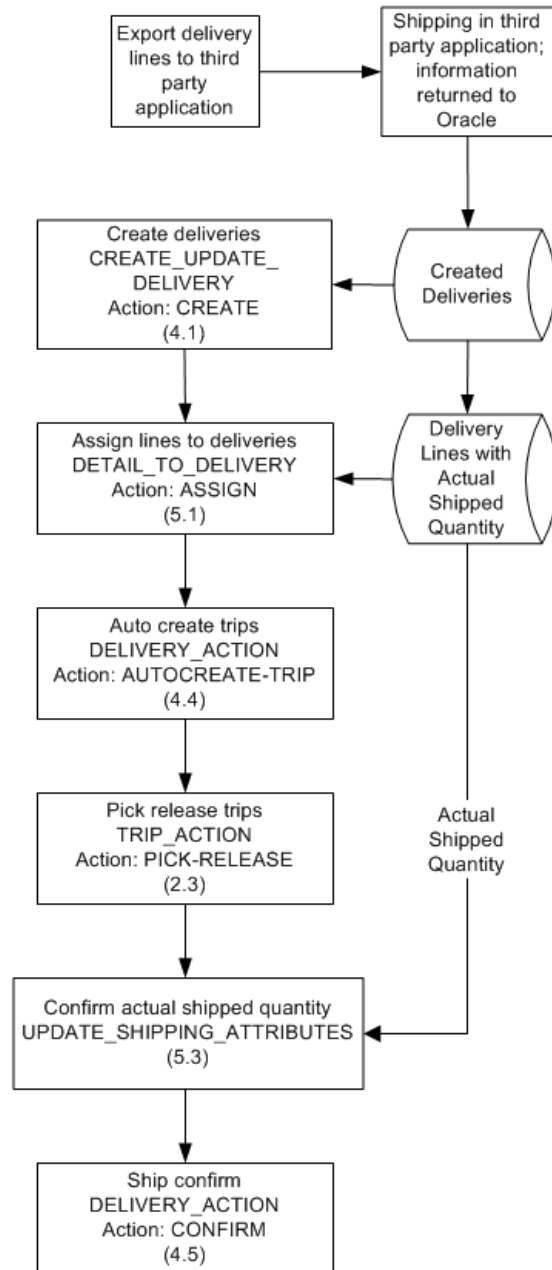
- Delivery lines sent to a third party application that performs the shipping.
 - After shipment, the third party application returns the delivery line IDs actually shipped and their serial numbers.
 - Create deliveries using AUTOCREATE_DELIVERIES API and launch pick release using DELIVERY_ACTION API with action PICK-RELEASE action
 - Update the actual serial number using UPDATE_SHIPPING_ATTRIBUTE.API or use overloaded UPDATE_SHIPPING_ATTRIBUTES API for entering range of serial numbers for the delivery detail
 - Ship confirm the deliveries using DELIVERY_ACTION API with action CONFIRM.
- Note: Serial numbers will exist in the database if the item is serial controlled at Receipt or is pre-defined.
- Close trips manually.



Shipping API Flow Scenario 3

Create Delivery, Auto-Create Trip, and Pick Release Trip

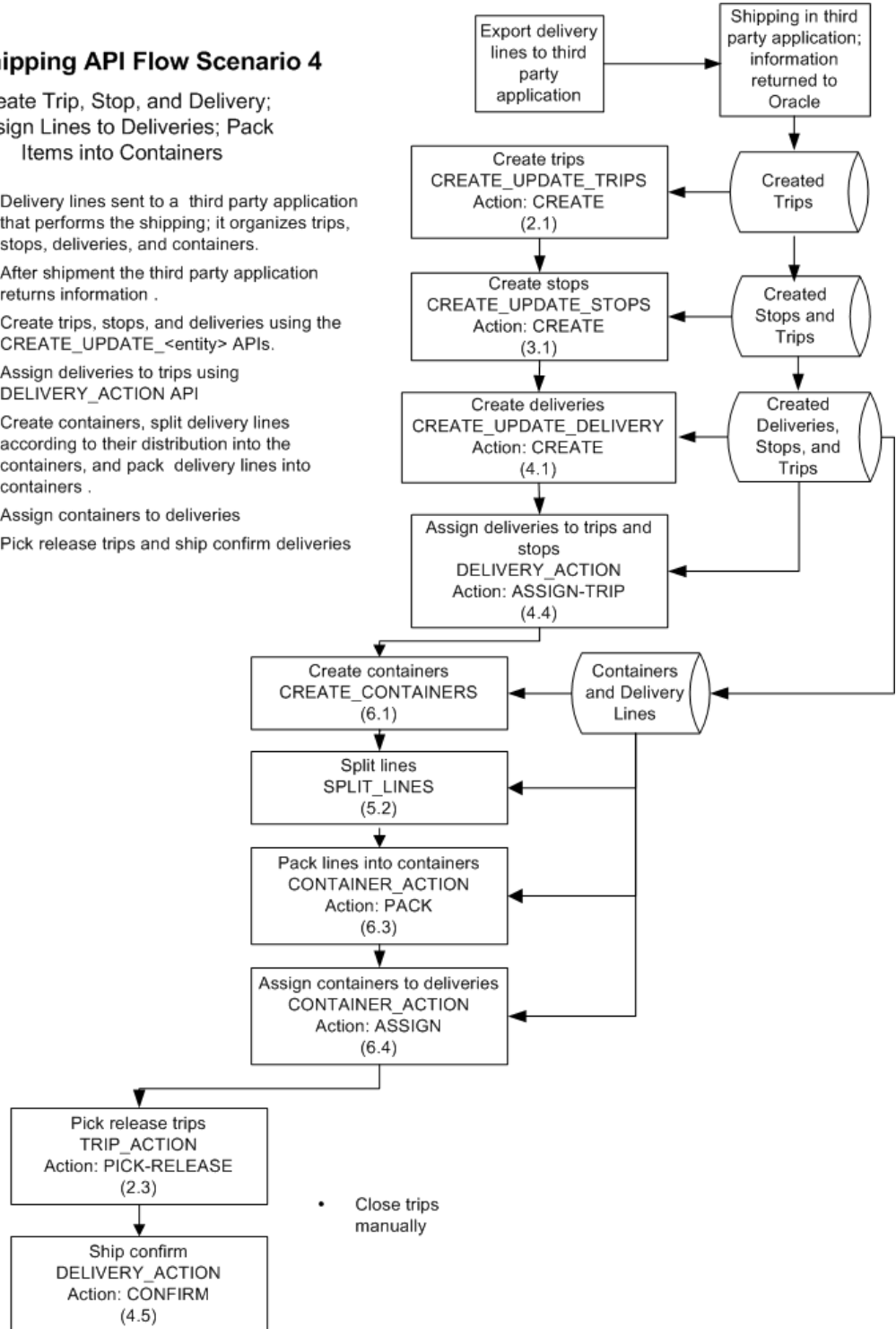
- Delivery lines sent to a third party application that performs the shipping; it organizes the deliveries and confirms actual shipped quantities.
- After shipment, the third party application returns the delivery lines associated with those deliveries, and the actual shipped quantity of each delivery line.
- Create deliveries using CREATE_UPDATE_DELIVERY API.
- Associate delivery lines to deliveries using DETAIL_TO_DELIVERY API.
- Autocreate trips and then pick release.
- Actual shipped quantity is confirmed using UPDATE_SHIPPING_ATTRIBUTE API.
- Ship confirm deliveries.
- After actual shipped quantity entered p_sc_action_flag is Code S: Ship Entered.
- Close trips manually.



Shipping API Flow Scenario 4

Create Trip, Stop, and Delivery;
Assign Lines to Deliveries; Pack
Items into Containers

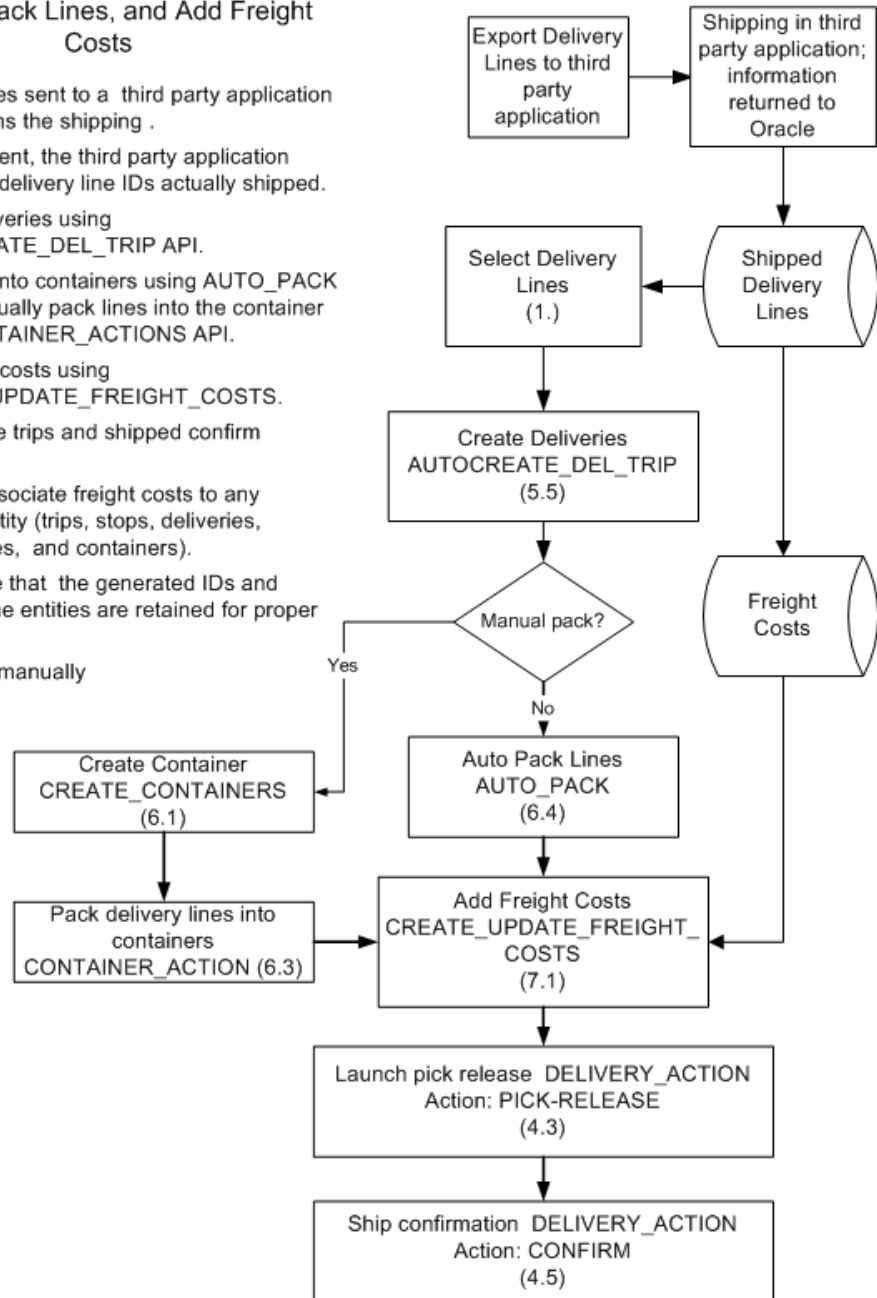
- Delivery lines sent to a third party application that performs the shipping; it organizes trips, stops, deliveries, and containers.
- After shipment the third party application returns information .
- Create trips, stops, and deliveries using the CREATE_UPDATE_<entity> APIs.
- Assign deliveries to trips using DELIVERY_ACTION API
- Create containers, split delivery lines according to their distribution into the containers, and pack delivery lines into containers .
- Assign containers to deliveries
- Pick release trips and ship confirm deliveries



Shipping API Flow Scenario 5

Autocreate Delivery and Trip, Auto/ Manual Pack Lines, and Add Freight Costs

- Delivery lines sent to a third party application that performs the shipping .
- After shipment, the third party application returns the delivery line IDs actually shipped.
- Create deliveries using AUTO_CREATE_DEL_TRIP API.
- Pack lines into containers using AUTO_PACK API or manually pack lines into the container using CONTAINER_ACTIONS API.
- Add freight costs using CREATE_UPDATE_FREIGHT_COSTS.
- Pick release trips and shipped confirm deliveries.
- You can associate freight costs to any shipping entity (trips, stops, deliveries, delivery lines, and containers).
- We assume that the generated IDs and names of the entities are retained for proper association
- Close trips manually



Actions, APIs, and Parameters

This section specifies the APIs you use to perform actions on shipping entities and lists the parameters that the APIs use.

Each time you call an API, include the parameters listed in the Required Parameters

column and, in addition, include the standard parameters p_api_version_number, p_init_msg_list, p_commit, p_validation_level, x_return_status, x_msg_count, and x_msg_data.

Some of the parameters may have default values in their API signatures. Refer to the Record Parameter Description section for each API to see if a parameter has a default value.

Some of the parameters are Table type or Record type. Refer to the Record Parameter Description section for each API to see its table and record definitions.

Trips

The following table shows trip actions, APIs, and parameters:

Trip Actions, APIs, and Parameters

Action	API	Required Parameters	Optional Parameters	Comments
Plan	WSH_TRIPS_PU B.Trip_Action	p_action_code = PLAN p_trip_id or p_trip_name	-	-
Unplan	WSH_TRIPS_PU B.Trip_Action	p_action_code = UNPLAN p_trip_id or p_trip_name	-	-
Launch pick release	WSH_TRIPS_PU B.Trip_Action	p_action_code = PICK- RELEASE p_trip_id or p_trip_name	-	-
Calculate weight and volume	WSH_TRIPS_PU B.Trip_Action	p_action_code = WT-VOL p_trip_id or p_trip_name	p_wv_override_f lag	-

Action	API	Required Parameters	Optional Parameters	Comments
Delete	WSH_TRIPS_PU B.Trip_Action	p_action_code = DELETE p_trip_id or p_trip_name	-	-
Create	WSH_TRIPS_PU B.Create_Update _Trip	p_action_code = CREATE p_trip_info x_trip_id x_trip_name	p_trip_name	-
Update	WSH_TRIPS_PU B.Create_Update _Trip	p_action_code = UPDATE p_trip_info x_trip_id x_trip_name	p_trip_name	For p_trip_info, pass only the attributes that need updating.
Print Document Sets	WSH_TRIPS_PU B.TRIP_ACTION	p_action_code = PRINT-DOC-SE TS p_report_set_na me or p_trip_name	-	-
Firm Routing	WSH_TRIPS_PU B.TRIP_ACTION	p_action_code = FIRM p_trip_id or p_trip_name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Ship Confirm	WSH_TRIPS_PU B.TRIP_ACTION	p_action_code = TRIP-CONFIRM p_trip_id or p_trip_name	-	-
Print Master BOL	WSH_TRIPS_PU B.TRIP_ACTION	p_action_code = PRINT-MBOL p_trip_id or p_trip_name	-	-
Unassign All Deliveries	WSH_TRIPS_PU B.TRIP_ACTION	p_action_code = UNASSIGN-AL L p_trip_id or p_trip_name	-	-

Stops

The following table shows stop actions, APIs, and parameters:

Stop Actions, APIs, and Parameters

Action	API	Required Parameters	Optional Parameters	Comments
Plan	WSH_TRIP_STO PS_PUB. Stop_Action	p_action_code = - PLAN p_stop_id		You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_location_ code or p_stop_location_ id p_planned_dep_ date
Unplan	WSH_TRIP_STO PS_PUB. Stop_Action	p_action_code = - UNPLAN p_stop_id		You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_location_ code or p_stop_location_ id p_planned_dep_ date

Action	API	Required Parameters	Optional Parameters	Comments
Update Status - Arrive	WSH_TRIP_STOPS_PUB. Stop_Action	p_action_code = ARRIVE p_stop_id	p_actual_date p_defer_interface_flag	You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_location_code or p_stop_location_id p_planned_dep_date
Update Status - Close	WSH_TRIP_STOPS_PUB. Stop_Action	p_action_code = CLOSE p_stop_id	p_actual_date p_defer_interface_flag	You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_location_code or p_stop_location_id p_planned_dep_date
Launch pick release	WSH_TRIP_STOPS_PUB. Stop_Action	p_action_code = PICK- RELEASE p_stop_id	-	You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_location_code or p_stop_location_id p_planned_dep_date

Action	API	Required Parameters	Optional Parameters	Comments
Delete	WSH_TRIP_STO PS_PUB. Stop_Action	p_action_code = DELETE p_stop_id	-	You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_location_ code or p_stop_location_ id p_planned_dep_ date
Create	WSH_TRIP_STO PS_PUB. Create_Update_ Stop	p_action_code = CREATE p_stop_info x_stop_id	-	-
Update	WSH_TRIP_STO PS_PUB. Create_Update_ Stop	p_action_code = UPDATE p_stop_info x_stop_id	p_trip_id p_trip_name p_stop_location_ id p_stop_location_ code p_planned_dep_ date	For p_stop_info, pass only the attributes that need updating.
Reset Weight/Vol	WSH_TRIP_STO PS_PUB. Stop_Action	p_action_code = WT-VOL p_stop_id	-	You can substitute all of the following for p_stop_id: p_trip_name or p_trip_id p_stop_ location_code or p_stop_location_ id

Deliveries

The following table shows delivery actions, APIs, and parameters:

Delivery Actions, APIs, and Parameters

Action	API	Required Parameters	Optional Parameters	Comments
Plan	wsh_deliveries_ pub. delivery_action	p_action_code = PLAN p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-
Unplan	wsh_deliveries_ pub. delivery_action	p_action_code = UNPLAN p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Ship confirm	wsh_deliveries_ pub. delivery_action	p_action_code = CONFIRM p_delivery_id or p_delivery_name x_trip_id x_trip_name	p_sc_action_flag p_sc_close_trip_ flag p_sc_create_bol p_sc_stage_del_ flag p_sc_trip_ship_ method p_sc_actual_dep_ date p_sc_report_set_ id p_sc_report_set_ name p_sc_defer_inter face	-
Reopen	wsh_deliveries_ pub. delivery_action	p_action_code = RE-OPEN p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-
Close	wsh_deliveries_ pub. delivery_action	p_action_code = CLOSE p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Assign to trip	wsh_deliveries_ pub. delivery_action	<p>p_action_code = ASSIGN-TRIP</p> <p>p_delivery_id or p_delivery_name</p> <p>p_asg_trip_id or p_asg_trip_name</p> <p>x_trip_id</p> <p>x_trip_name</p>	<p>p_asg_pickup_stop_id</p> <p>p_asg_pickup_location_id</p> <p>p_asg_pickup_stop_seq</p> <p>p_asg_pickup_location_code</p> <p>p_asg_pickup_arrival_date</p> <p>p_asg_pickup_departure_date</p> <p>p_asg_dropoff_stop_id</p> <p>p_asg_dropoff_location_id</p> <p>p_asg_dropoff_stop_seq</p> <p>p_asg_dropoff_location_code</p> <p>p_asg_dropoff_arrival_date</p> <p>p_asg_dropoff_departure_date</p>	-
Unassign from trip	wsh_deliveries_ pub. delivery_action	<p>p_action_code = UNASSIGN-TRIP</p> <p>p_delivery_id or p_delivery_name</p> <p>x_trip_id</p> <p>x_trip_name</p>	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Auto-create trip	wsh_deliveries_ pub. delivery_action	p_action_code = AUTOCREATE- TRIP p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-
Calculate weight and volume	wsh_deliveries_ pub. delivery_action	p_action_code = WT-VOL p_delivery_id or p_delivery_name x_trip_id x_trip_name	p_wv_override_f lag	-
Launch pick release	wsh_deliveries_ pub. delivery_action	p_action_code = PICK- RELEASE p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-
Delete	wsh_deliveries_ pub. delivery_action	p_action_code = DELETE p_delivery_id or p_delivery_name x_trip_id x_trip_name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Create	wsh_deliveries_ pub. create_update_ elivery	p_action_code = CREATE p_delivery_info x_delivery_id x_name	p_delivery_name	-
Update	wsh_deliveries_ pub. create_update_ elivery	p_action_code = UPDATE p_delivery_info x_delivery_id x_name	p_delivery_name	For p_delivery_info, pass only the attributes that need updating.
Auto-pack	WSH_CONTAI NER_PUB.Auto_ Pack	p_entity_tab p_entity_type p_group_id_tab p_pack_cont_ flag x_cont_inst_tab	-	-
Pick and Ship	wsh_deliveries_ pub. delivery_action	p_action_code = PICK- SHIP p_delivery_id p_delivery_ name x_trip_idx_ trip_ name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Pick, Pack and Ship	wsh_deliveries_ pub. delivery_action	p_action_code = PICK-PACK-SHI P p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Send Outbound Message	wsh_deliveries_ pub. delivery_action	p_action_code = OUTBOUND-D OCUMENT p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Print Document Sets	wsh_deliveries_ pub. delivery_action	p_action_code = PRINT-DOC-SE TS p_delivery_id p_delivery_name p_sc_report_set_name or p_sc_report_set_idx_trip_idx_trip_name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Firm Routing and Contents	wsh_deliveries_ pub. delivery_action	p_action_code = FIRM p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Auto-Pack	wsh_deliveries_ pub. delivery_action	p_action_code = AUTO-PACK p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Auto-Pack Master	wsh_deliveries_ pub. delivery_action	p_action_code = AUTO-PACK-MASTER p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Generate Loading Sequence	wsh_deliveries_ pub. delivery_action	p_action_code = GEN-LOAD-SEQ p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Select Carrier	wsh_deliveries_ pub. delivery_action	p_action_code = SELECT-CARRIER p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Cancel Ship Method	wsh_deliveries_ pub. delivery_action	p_action_code = CANCEL-SHIP-METHOD p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Print Pack Slip	wsh_deliveries_ pub. delivery_action	p_action_code = PRINT-PACK-SLIP p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-
Print BOL	wsh_deliveries_ pub. delivery_action	p_action_code = PRINT-BOL p_delivery_id p_delivery_name x_trip_idx_trip_name	-	-

Delivery Details

The following table shows delivery detail actions, APIs, and parameters:

Delivery Detail Actions, APIs, and Parameters

Action	API	Required Parameters	Optional Parameters	Comments
Assign to delivery	wsh_delivery_details_pub. detail_to_delivery	p_TabOfDelDets p_action = ASSIGN, p_delivery_id p_delivery_name	-	-
Unassign from delivery	wsh_delivery_details_pub. detail_to_delivery	p_TabOfDelDets p_action = UNASSIGN p_delivery_id p_delivery_name	-	-
Split line	wsh_delivery_details_pub. split_line	p_from_detail_id x_split_quantity x_new_detail_id x_split_quantity 2	-	-
Update	wsh_delivery_details_pub. UpdateShipping_Attributes	p_changed_attributes p_source_code	p_container_flag	For p_changed_attributes, pass only the attributes that need updating.
Autocreate delivery	wsh_delivery_details_pub. Autocreate_Deliveries	p_line_rows x_del_rows	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Autocreate trip	wsh_delivery_details_pub. Autocreate_delivery_trip	p_line_rows x_del_rows x_trip_id x_trip_name	-	-
Pack into a container	WSH_CONTAINER_PUB. Container_Actions	p_detail_tab p_action_code =PACK p_container_name or p_cont_instance_id	p_container_flag p_delivery_flag	-
Unpack from a container	WSH_CONTAINER_PUB. Container_Actions	p_detail_tab p_action_code =UNPACK p_container_name or p_cont_instance_id	p_container_flag p_delivery_flag	-
Auto-pack	WSH_CONTAINER_PUB.Auto_Pack	p_entity_tab p_entity_type p_group_id_tab p_pack_cont_flag x_cont_inst_tab	-	-

Action	API	Required Parameters	Optional Parameters	Comments
Create container	WSH_CONTAINER_PUB.Create_Containers	<p>p_container_item_id</p> <p>p_container_item_name</p> <p>p_container_item_seg</p> <p>p_organization_id</p> <p>p_organization_code</p> <p>p_name_prefix</p> <p>p_name_suffix</p> <p>p_base_number</p> <p>p_num_digits</p> <p>p_quantity</p> <p>p_container_name</p> <p>x_container_ids</p>	-	-
Update container	WSH_CONTAINER_PUB.Update_Container	p_container_rec	-	For p_container_rec, pass only the attributes that need updating.
Assign container to a delivery	WSH_CONTAINER_PUB.Container_Actions	<p>p_detail_tab</p> <p>p_action_code = ASSIGN</p> <p>p_delivery_id or p_delivery_name</p>	<p>p_container_flag</p> <p>p_delivery_flag</p>	-

Action	API	Required Parameters	Optional Parameters	Comments
Unassign container from a delivery	WSH_CONTAINER_PUB.Container_Actions	p_detail_tab p_action_code = UNASSIGN p_delivery_id or p_delivery_name	p_container_flag p_delivery_flag	-
Enter serial numbers at Sales Order issue.	WSH_DELIVERY_DETAILS_PUB.Update_Shipping_Attributes (overloaded)	p_changed_attributes p_source_code p_serial_range_tab	p_container_flag	For p_changed_attributes, pass only the attributes that need update. For p_serial_range_tab pass the delivery_detail_id, from_serial_number, to_serial_number and quantity.

Application Parameter Initialization

In your scripts that call Public APIs, include a call to `fnf_global.apps_initialize` immediately after the BEGIN. The call syntax is

```
fnf_global.apps_initialize(user_id => l_user_id,
resp_id => l_resp_id,
resp_appl_id => l_resp_appl_id,
security_group_id => l_security_group_id);
```

Finding Variable Values

To find the values for the variables, perform these steps:

1. Navigate to the Transaction form.
Help > Diagnostics > Examine.
2. In Block, enter \$PROFILES\$

3. In Field, enter USER_ID and press Tab. Use the value in Value for l_user_id.
4. In Field, enter RESP_ID and press Tab. Use the value in Value for l_resp_id.
5. In Field, enter RESP_APPL_ID and press Tab. Use the value in Value for l_resp_appl_id.
6. In Field, enter SECURITY_GROUP_ID and press Tab. Use the value in Value for l_security_group_id.

Trip Public Application Program Interface

The Trip Public Application Program Interface (API) is a public API that consists of the following two procedures in package **WSH_TRIPS_PUB**:

Create_Update_Trip: Enables you to create a new trip record and update an existing trip record.

Trip_Action: Enables you to perform certain actions on a trip.

This section describes how to use the Trip Public API and how it functions in Oracle Shipping Execution.

Create_Update_Trip API Features

The Create_Update_Trip API has the following features.

The Create_Update_Trip procedure enables you to create a new trip record or update an existing trip record in the WSH_TRIPS table. The TRIP_ID, NAME and return status of a new trip are passed as OUT parameters, while the trip NAME of an existing trip for update is passed as an IN parameter.

Functional Overview

This API creates a new trip record in WSH_TRIPS as specified by IN parameter p_action_code value CREATE. It inserts the trip information into WSH_TRIPS and returns the TRIP_ID and NAME of the new trip. It also updates an existing trip record in WSH_TRIPS as specified by IN parameter p_action_code value UPDATE. The NAME of the trip being updated is passed through IN parameter p_trip_name.

The API validates trip information such as Trip Name, Arrive After Trip Name, and Ship Method before performing the actions of creating or updating a trip record. It also checks that the insert or update statements were successful, and if not returns an error.

Procedure Parameter Descriptions

The following chart describes all parameters used by the public procedure WSH_TRIPS_PUB.CREATE_UPDATE_TRIP. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_TRIPS_PUB.CREATE_UPDATE_TRIP Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_action_code	IN	Varchar2	x
p_trip_info	IN OUT	Record	x
p_trip_name	IN	Varchar2	-
x_trip_id	OUT	Number	-
x_trip_name	OUT	Varchar2	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_action_code

Specifies whether API should create a new trip or update existing trip information based on its values CREATE or UPDATE.

p_trip_info

Attributes of the trip entity of type Trip_Pub_Rec_Type. These attributes are inserted/updated in WSH_TRIPS. Definition of Trip_Pub_Rec_Type follows.

p_trip_name

IN parameter to specify the Name of trip to be updated.

Default Value: FND_API.G_MISS_CHAR

x_trip_id

Trip ID of new trip being created.

x_trip_name

Name of new trip being created.

Record Parameter Descriptions

TRIP_PUB_REC_TYPE RECORD DEFINITION

To encapsulate WSH_TRIPS table definition and Value column equivalents for ID columns in a PL/SQL record, define TRIP_PUB_REC_TYPE to pass trip information to the Create_Update_Trip routine.

TRIP_PUB_REC_TYPE Record Definition

Attribute	Type	Default Value
trip_id	Number	fnd_api.g_miss_num
name	Varchar2(30)	fnd_api.g_miss_char
arrive_after_trip_id	Number	fnd_api.g_miss_num
arrive_after_trip_name	Varchar2(30)	fnd_api.g_miss_char
vehicle_item_id	Number	fnd_api.g_miss_num
vehicle_item_desc	Varchar2(240)	fnd_api.g_miss_char
vehicle_organization_id	Number	fnd_api.g_miss_num
vehicle_organization_code	Varchar2(3)	fnd_api.g_miss_char
vehicle_number	Varchar2(30)	fnd_api.g_miss_char
vehicle_num_prefix	Varchar2(10)	fnd_api.g_miss_char
carrier_id	Number	fnd_api.g_miss_num
ship_method_code	Varchar2(30)	fnd_api.g_miss_char
ship_method_name	Varchar2(80)	fnd_api.g_miss_char
route_id	Number	fnd_api.g_miss_num
routing_instructions	Varchar2(2000)	fnd_api.g_miss_char
seal_code	Varchar2(30)	fnd_api.g_miss_char
attribute_category	Varchar2(150)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char
creation_date	Date	fnd_api.g_miss_char
created_by	Number	fnd_api.g_miss_num
last_update_date	Date	fnd_api.g_miss_date
last_updated_by	Number	fnd_api.g_miss_num
last_update_login	Date	fnd_api.g_miss_date
program_application_id	Number	fnd_api.g_miss_num
program_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
program_update_date	Date	fnd_api.g_miss_date
request_id	Number	fnd_api.g_miss_num
service_level	Varchar2(30)	fnd_api.g_miss_char
mode_of_transport	Varchar2(30)	fnd_api.g_miss_char
operator	Varchar2(150)	fnd_api.g_mis_char

Record Parameter Attribute Validations

trip_id

Should be a valid unique element of wsh_trips.trip_id.

name

Should be a valid unique element of wsh_trips.name

arrive_after_trip_id

Should be a valid element of wsh_trips.trip_id.

arrive_after_trip_name

Should be a valid element of wsh_trips.name

vehicle_item_id

Should be a valid vehicle item in mtl_system_items.inventory_item_id as designated by mtl_system_items.vehicle_tem_flag.

vehicle_item_desc

Key flex field description for vehicle item.

vehicle_organization_id

Should be a valid element of org_organization_definitions.organization_id and be assigned to the specific item in mtl_system_items.organization_id.

vehicle_organization_code

Should be a valid element of org_organization_definitions.organization_code for that specific organization_id.

ship_method_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

ship_method_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type SHIP_METHOD.

seal_code

Indicates the Departure seal code of the trip stops.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Trip_Action API Features

The Trip_Action API has the following features:

The Trip_Action procedure enables you to carry out actions on a trip. For IN parameters, it accepts the trip identifiers, an action code, and any additional parameters needed for specific actions, and returns a completion status.

Functional Overview

This API validates the trip identifiers and performs the following actions specified in p_action_code by calling the corresponding private procedures.

- PLAN/UNPLAN: Called to plan or unplan trips.
- WT-VOL: Called to calculate weight and volume for the trip.
- PICK-RELEASE: Called to launch pick release to release lines related to the trip.
- PRINT-DOC-SETS: Called to print document sets.
- DELETE: Called to delete trips.

Procedure Parameter Descriptions

WSH_TRIPS_PUB.TRIP_ACTION

The following table describes all parameters used by the public procedure WSH_TRIPS_PUB.TRIP_ACTION. All of the inbound and outbound parameters are listed. Additional information about these parameters follows the table.

WSH_TRIPS_PUB.TRIP_ACTION Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	Y
p_init_msg_list	IN	Varchar2	N
x_return_status	OUT	Varchar2	N
x_msg_count	OUT	Number	N
x_msg_data	OUT	Varchar2	N
p_action_code	IN	Varchar2	Y
p_trip_id	IN OUT	Record	N
p_trip_name	IN	Varchar2	N
p_wv_override_flag	IN	Varchar2	N

p_api_version_number

Compares the version number of the incoming API call to the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. Values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this text contains the actual message.

p_action_code

Specifies which of the actions (PLAN, UNPLAN, WT-VOL, PICK RELEASE, or DELETE) the API should perform.

p_trip_id

ID of the trip on which actions are being performed.

p_trip_name

Name of the trip on which actions are being performed.

p_wv_override_flag

Additional parameter for WT-VOL. Regardless of whether this is set to Y or N, the system recalculates weight and volume for all deliveries on the trip. The default value is N.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Trip_Action (overloaded version) API features

The WSH_TRIPS_PUB.TRIP_ACTION API consists of the following entities:

The Trip_Action procedure enables you to carry out actions on a trip. This overloaded API accepts a record of parameters for various actions. The Valid action codes are PLAN , UNPLAN, WT-VOL, PICK-RELEASE , TRIP-CONFIRM, PRINT-DOC-SETS, GENERATE-ROUTING-RESPONSE, FIRM, IGNORE_PLAN, INCLUDE_PLAN, GENERATE-PACK-SLIP, PRINT-PACK-SLIP, GENERATE-BOL, PRINT-BOL, PRINT-MBOL, REMOVE-CONSOL, UNASSIGN-ALL, CREATE, UPDATE, and DELETE.

Functional Overview

This API validates the trip identifiers and performs the following actions specified in p_action_param_rec by calling the corresponding private procedures.

- PLAN/UNPLAN: Is called to plan or unplan trips.
- WT-VOL: Is called to calculate weight and volume for the trip.
- PICK-RELEASE: Is called to launch pick release to release lines related to the trip.
- TRIP-CONFIRM : Is called to Confirm the Trip
- PRINT-DOC-SETS: Is called to print document sets.
- GENERATE-PACK-SLIP :Is called to generate the Packing Slip Number.
- PRINT-PACK-SLIP : Is called to generate the Packing Slip Report.
- GENERATE-BOL : Is called to generate the Bill of Lading Number.
- PRINT-BOL : Is called to print the Bill of Lading report.
- PRINT-MBOL : : Is called to print the Master Bill of Lading report.
- CREATE : Is called to create trips.
- UPDATE : Is called to update trips.
- DELETE: Is called to delete trips.

Procedure Parameters Descriptions

The following table lists all parameters used by the public WSH_TRIPS_PUB.TRIP_ACTION. All of the inbound and outbound parameters are listed. Additional information on these parameters follows the table.

Parameter Name	Usage	Type	Required	Derived
p_api_version_number	IN	NUMBER	x	No
p_init_msg_list	IN	VARCHAR2	-	No
x_return_status	OUT	VARCHAR2	-	No

Parameter Name	Usage	Type	Required	Derived
x_msg_count	OUT	NUMBER	-	No
x_msg_data	OUT	NUMBER	-	No
p_action_param_rec	IN	WSH_TRIPS_PUB. Action_Param_Re ctype	x	No
p_trip_id	IN	NUMBER	-	No
p_trip_name	IN	VARCHAR2	-	No

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR

- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_action_param_rec Specifies which of the actions of PLAN , UNPLAN, WT-VOL, PICK-RELEASE, TRIP-CONFIRM, PRINT-DOC-SETS, GENERATE-ROUTING-RESPONSE, FIRM, IGNORE_PLAN, INCLUDE_PLAN, GENERATE-PACK-SLIP, PRINT-PACK-SLIP, GENERATE-BOL, PRINT-BOL, PRINT-MBOL, REMOVE-CONSOL, UNASSIGN-ALL, CREATE, UPDATE and DELETE the API should perform.

p_trip_id

Trip ID of trip on which actions are being performed.

p_trip_name

Name of trip on which actions are being performed.

Record Parameter Descriptions

ACTION_PARAM_RECTYPE RECORD DEFINITION

To encapsulate WSH_TRIP table definition and Value column equivalents for ID columns in a PL/SQL record, define ACTION_PARAM_RECTYPE to pass trip information to the trip_action routine.

Parameter p_action_param_rec of type Action_Param_Rectype.

Parameter Name	Type	Required	Derived
action_code	VARCHAR2	x	No
Organization_id	NUMBER	x	No
report_set_name	VARCHAR2	-	No
report_set_id	NUMBER	-	No
Override_flag	VARCHAR2	-	No
actual_date	DATE	-	No

Parameter Name	Type	Required	Derived
action_flag	VARCHAR2	-	No
Autointransit_flag	VARCHAR2	-	No
autoclose_flag	VARCHAR2	-	No
stage_del_flag	VARCHAR2	-	No
ship_method	VARCHAR2	-	No
bill_of_lading_flag	VARCHAR2	-	No
defer_interface_flag	VARCHAR2	-	No
actual_departure_date	Date	-	No

action_code

Specifies which of the actions of PLAN , UNPLAN, WT-VOL, PICK-RELEASE , TRIP-CONFIRM, PRINT-DOC-SETS, GENERATE-ROUTING-RESPONSE, FIRM, IGNORE_PLAN, INCLUDE_PLAN, GENERATE-PACK-SLIP, PRINT-PACK-SLIP, GENERATE-BOL, PRINT-BOL, PRINT-MBOL, ,REMOVE-CONSOL, UNASSIGN-ALL, CREATE, UPDATE and DELETE the API should perform.

organization_id

Should be a valid element of org_organization_definitions.organization_id.

report_set_name

Should be a valid element of wsh_report_sets.report_set_name.

report_set_id

Should be a valid element of wsh_report_sets.report_set_id.

override_flag

Should be a valid value 'Y' or 'N'.

actual_date

Actual date of arrival/departure of trip stop. Should be a valid date.

action_flag

Should be a valid value 'S', 'B', 'L', 'T', 'A', 'C' or 'O'.

'S' => Ship Entered, Ship Unspecified Full

'B' => Ship Entered, Backorder Unspecified

'L' => cycle count unspecified quantities

'T' => stage unspecified quantities

'A' => Ship all

'O' => Cycle count all

'C' => Backorder all.

autointransit_flag

Should be a valid value 'Y' or 'N'.

autoclose_flag

Should be a valid value 'Y' or 'N'.

stage_del_flag

Should be a valid value 'Y' or 'N'.

ship_method

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

bill_of_lading_flag

Should be a valid value 'Y' or 'N'.

defer_interface_flag

Ship Confirm flag to submit/defer concurrent requests to INV and OE interfaces.

Default Value: Y.

Standard Validation

Oracle Shipping Execution validates all required columns in the WSH_TRIPS_PUB.TRIP_ACTION API. For specific information on the data referenced by these columns, see *Oracle Electronic Technical Reference Manual (eTRM)*.

Error Handling

If any validation fails, the API will return the error status to the calling module. The WSH_TRIPS_PUB.TRIP_ACTION API processes the rows and reports the following values for every record:

Condition	Message Return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

Stop Public Application Program Interface

The Stop public API consists of the following procedures in package **WSH_TRIP_STOPS_PUB**:

Create_Update_Stop: Enables you to create a new stop record and update an existing stop record.

Stop_Action: Enables you to perform certain actions on a stop.

This section describes how to use the Stop Public API and how it functions in Oracle Shipping Execution.

Create_Update_Stop API Features

The Create_Update_Stop API has the following features.

The Create_Update_Stop procedure enables you to create a new stop record or update an existing stop record in the WSH_TRIP_STOPS table. The STOP_ID and return status of a new stop are passed as OUT parameters, while the TRIP_ID of an existing stop for update is passed as an IN parameter.

Functional Overview

This API creates a new stop record in WSH_TRIP_STOPS as specified by the IN parameter p_action_code value CREATE. It inserts stop information into WSH_TRIP_STOPS and returns the STOP_ID of the new stop. It also updates an existing stop record in WSH_TRIP_STOPS as specified by IN parameter p_action_code value UPDATE. The updated stop record is identified by the Stop Id or Trip Id and Name, Stop Location Id and Code and Planned Departure Date. Additional IN parameters are provided to update Trip Id and Name, Stop Location Id and Code and Planned Departure Date values.

The API validates information such as Trip Name, Location Code, and units of measure before creating or updating a stop record. It also checks that the insert or update statements were successful and, if not successful, it returns an error.

Procedure Parameter Descriptions

WSH_TRIP_STOPS_PUB.CREATE_UPDATE_STOP

The following table describes all parameters used by the public procedure WSH_TRIP_STOPS_PUB.CREATE_UPDATE_STOP. All of the inbound and outbound parameters are listed. Additional information about these parameters follows.

WSH_TRIP_STOPS_PUB.CREATE_UPDATE_STOP Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	Y
p_init_msg_list	IN	Varchar2	N
x_return_status	OUT	Varchar2	N
x_msg_count	OUT	Number	N
x_msg_data	OUT	Varchar2	N
p_action_code	IN	Varchar2	Y
p_stop_info	IN OUT	Record	Y
p_trip_id	IN	Number	N
p_trip_name	IN	Varchar2	N
p_stop_location_id	IN	Number	N
p_location_code	IN	Varchar2	N
p_planned_dep_date	IN	Varchar2	N
x_stop_id	OUT	Number	N

p_api_version_number

Compares the version number of the incoming API call to the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the `x_msg_count` is greater than 1, then the list of messages must be retrieved using the call `FND_MSG_PUB.GET`. Values are:

- `p_msg_index => I`
- `p_encoded => F`
- `p_data => 1_message`
- `p_msg_index_out => 1_msg_index_out`

where `1_message` and `1_msg_index_out` are local variables of types `Varchar2(2000)` and `Number` respectively.

Default Value: `FND_API.G_FALSE`

x_return_status

Requests that the API return the status of the data for you after it completes its function. Values include:

- Success: `FND_API.G_RET_STS_SUCCESS`
- Error: `FND_API.G_RET_STS_ERROR`
- Unexpected Error: `FND_API.G_RET_STS_UNEXP_ERROR`

x_msg_count

Indicates the number of error messages API has encountered.

x_msg_data

Displays an error message text. If the `x_msg_count` is equal to 1, then `x_msg_data` contains the actual message.

p_action_code

Specifies whether the API should create a new stop or update existing stop information depending on its `CREATE` and `UPDATE` values.

p_stop_info

Attributes of the stop entity of type `Trip_Stop_Pub_Rec_Type`. These attributes are inserted/updated in `WSH_TRIP_STOPS`. Definition of `Trip_Stop_Pub_Rec_Type` follows.

p_trip_id

IN parameter to specify trip ID of trip stop record being updated.

Default Value: `FND_API.G_MISS_NUM`

p_trip_name

IN parameter to specify trip name of trip stop record that needs to be updated.

Default Value: FND_API.G_MISS_CHAR

p_stop_location_id

IN parameter to specify stop location ID of trip stop record being updated.

Default Value: FND_API.G_MISS_NUM

p_location_code

IN parameter to specify stop location code of stop record that needs to be updated.

Default Value: FND_API.G_MISS_CHAR

p_planned_dep_date

IN parameter to specify the planned date of departure.

Default Value: FND_API.G_MISS_DATE

x_stop_id

OUT parameter to specify the new stop ID of the new stop record being created.

Record Parameter Descriptions

TRIP_STOP_PUB_REC_TYPE RECORD DEFINITION

To encapsulate WSH_TRIP_STOPS table definition and Value column equivalents for ID columns in a PL/SQL record, define TRIP_STOP_PUB_REC_TYPE to pass trip information to the Create_Update_Stop routine.

TRIP_STOP_PUB_REC_TYPE RECORD DEFINITION

Attribute	Type	Default Value
stop_id	Number	fnd_api.g_miss_num
trip_id	Number	fnd_api.g_miss_num
trip_name	Varchar2(30)	fnd_api.g_miss_char
stop_location_id	Number	fnd_api.g_miss_num
stop_location_code	Varchar2(20)	fnd_api.g_miss_char
planned_arrival_date	Date	fnd_api.g_miss_date
planned_departure_date	Date	fnd_api.g_miss_date

Attribute	Type	Default Value
actual_arrival_date	Date	fnd_api.g_miss_date
actual_departure_date	Date	fnd_api.g_miss_date
departure_gross_weight	Number	fnd_api.g_miss_num
departure_net_weight	Number	fnd_api.g_miss_num
weight_uom_code	Varchar2(3)	fnd_api.g_miss_char
weight_uom_desc	Varchar2(25)	fnd_api.g_miss_char
departure_volume	Number	fnd_api.g_miss_num
volume_uom_code	Varchar2(3)	fnd_api.g_miss_char
volume_uom_desc	Varchar2(25)	fnd_api.g_miss_char
departure_seal_code	Varchar2(30)	fnd_api.g_miss_char
departure_fill_percent	Number	fnd_api.g_miss_char
tp_attribute_category	Varchar2(150)	fnd_api.g_miss_char
tp_attribute1	Varchar2(150)	fnd_api.g_miss_char
tp_attribute2	Varchar2(150)	fnd_api.g_miss_char
tp_attribute3	Varchar2(150)	fnd_api.g_miss_char
tp_attribute4	Varchar2(150)	fnd_api.g_miss_char
tp_attribute5	Varchar2(150)	fnd_api.g_miss_char
tp_attribute6	Varchar2(150)	fnd_api.g_miss_char
tp_attribute7	Varchar2(150)	fnd_api.g_miss_char
tp_attribute8	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
tp_attribute9	Varchar2(150)	fnd_api.g_miss_char
tp_attribute10	Varchar2(150)	fnd_api.g_miss_char
tp_attribute11	Varchar2(150)	fnd_api.g_miss_char
tp_attribute12	Varchar2(150)	fnd_api.g_miss_char
tp_attribute13	Varchar2(150)	fnd_api.g_miss_char
tp_attribute14	Varchar2(150)	fnd_api.g_miss_char
tp_attribute15	Varchar2(150)	fnd_api.g_miss_char
attribute_category	Varchar2(150)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char
creation_date	Date	fnd_api.g_miss_date
created_by	Number	fnd_api.g_miss_num
last_update_date	Date	fnd_api.g_miss_date
last_updated_by	Number	fnd_api.g_miss_num
last_update_login	Date	fnd_api.g_miss_date
program_application_id	Number	fnd_api.g_miss_num
program_id	Number	fnd_api.g_miss_num
program_update_date	Date	fnd_api.g_miss_date
request_id	Number	fnd_api.g_miss_num

Record Parameter Attribute Validations

stop_id

Should be a valid unique element of wsh_trip_stops.stop_id.

trip_id

Should be a valid unique element of wsh_trips.trip_id.

trip_name

Should be a valid unique element of wsh_trips.name

stop_location_id

Should be a valid element of hr_locations.location_id

stop_location_code

Should be a valid element of hr_locations.location_code.

weight_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a weight uom class.

weight_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a weight uom class.

volume_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a volume uom class.

volume_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a volume uom class.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Stop_Action API Features

The Stop_Action API has the following features:

The Trip_Action procedure enables you to carry out actions on a stop. It accepts as IN parameters the trip identifiers, an action code and any additional parameters needed for specific actions, and it returns a completion status.

Functional Overview

This API validates the stop identifiers and performs the following actions specified in p_action_code by calling the corresponding private procedures.

- PLAN/UNPLAN: Called to plan or unplan stops.
- ARRIVE/CLOSE: Called to update departure or arrival dates and to close a stop. This action needs an additional parameter P_ACTUAL_DATE for actual departure or arrival updates.
- PICK-RELEASE: Launches pick release to release lines related to a stop.
- DELETE: Called to delete stops.

Procedure Parameter Descriptions

WSH_TRIP_STOP_PUB.STOP_ACTION

The following chart describes all parameters used by the public procedure WSH_TRIP_STOP_PUB.STOP_ACTION. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_TRIP_STOP_PUB.STOP_ACTION Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_action_code	IN	Varchar2	x
p_stop_id	in	Number	-
p_trip_id	IN	Number	-
p_trip_name	IN	Varchar2	-
p_stop_location_id	IN	Varchar2	-
p_stop_location_code	IN	Varchar2	-
p_planned_dep_date	IN	Date	-
p_actual_date	IN	Date	-
p_defer_interface_flag	IN	Varchar2	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_action_code

Specifies which of the actions of PLAN, UNPLAN, ARRIVE, CLOSE, PICK RELEASE, DELETE the API should perform.

p_stop_id

Stop ID of trip stop on which actions need to be performed.

p_stop_id

Stop ID of trip stop on which actions need to be performed.

p_trip_id

Trip ID of trip stop on which actions need to be performed.

p_trip_name

Trip name of trip stop on which actions need to be performed.

p_stop_location_id

Stop location ID of trip stop on which actions need to be performed.

p_stop_location_code

Stop location code of trip stop on which actions need to be performed.

p_planned_dep_date

Planned departure date of trip stop on which actions need to be performed.

p_actual_date

Actual date of arrival/departure of trip stop.

p_defer_interface_flag

Flag to submit/defer concurrent requests to INV and OE interfaces.

Deliveries Public Application Program Interface

The Deliveries Public Application Program Interface (API) is a public API that consists of the following procedures in **WSH_DELIVERIES_PUB**:

- **Create_Update_Delivery:** Enables you to create a new trip stop record and update an existing trip stop record.
- **Delivery_Action:** Enables you to perform certain actions on a trip stop.
- **Generate_Documents:** Generates the document set for a delivery.

This section describes how to use the Deliveries Public API and how it functions in Oracle Shipping Execution.

Create_Update_Delivery API Features

The Create_Update_Delivery API has the following features.

The Create_Update_Delivery procedure enables you to create a new delivery record or update an existing delivery record in WSH_NEW_DELIVERIES table. The DELIVERY_ID, NAME and return status of a new delivery are passed as OUT parameters, while the DELIVERY_ID or NAME of an existing delivery for update is passed as an IN parameter.

Functional Overview

This API creates a new delivery record in WSH_NEW_DELIVERIES as specified by IN parameter p_action_code value CREATE. Inserts delivery information into WSH_NEW_DELIVERIES and returns the DELIVERY_ID of the new delivery. It also updates an existing delivery record in WSH_NEW_DELIVERIES as specified by IN parameter p_action_code value UPDATE. Delivery record being updated is identified by Delivery Name. An additional IN parameter is provided to identify delivery by Name for update.

The API validates delivery information such as Name, Customer Number, and Ship

Method before performing the actions CREATE or UPDATE. It also checks that the insert or update statements were successful, and if not, returns an error.

Procedure Parameter Descriptions

WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY

The following chart describes all parameters used by the public procedure WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_action_code	IN	Varchar2	x
p_delivery_info	IN OUT	Record	x
p_delivery_name	IN	Varchar2	-
x_delivery_id	OUT	Number	-
x_name	OUT	Varchar2	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Displays error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_action_code

Specifies whether API should create a new delivery or update existing delivery information depending on its values CREATE and UPDATE.

p_delivery_info

Attributes of the delivery entity of type Delivery_Pub_Rec_Type. These attributes are inserted/updated in WSH_NEW_DELIVERIES. Definition of Delivery_Pub_Rec_Type follows.

p_delivery_name

IN parameter to specify name of delivery record being updated.

Default Value: FND_API.G_MISS_CHAR

x_delivery_id

OUT parameter to specify delivery ID of new delivery record being created.

x_name

OUT parameter to specify name of new delivery record being created.

Record Parameter Descriptions

DELIVERY_PUB_REC_TYPE RECORD DEFINITION

To encapsulate WSH_NEW_DELIVERIES table definition and Value column equivalents for ID columns in a PL/SQL record, define DELIVERIES_PUB_REC_TYPE to pass delivery information to the Create_Update_Delivery routine.

DELIVERY_PUB_REC_TYPE RECORD DEFINITION

Attribute	Type	Default
delivery_id	Number	fnd_api.g_miss_num
name	Varchar2(30)	fnd_api.g_miss_char
delivery_type	Varchar2(30)	fnd_api.g_miss_char
loading_sequence	Number	fnd_api.g_miss_num
loading_order_flag	Varchar2(2)	fnd_api.g_miss_char
loading_order_desc	Varchar2(20)	fnd_api.g_miss_char
initial_pickup_date	Date	fnd_api.g_miss_date
initial_pickup_location_id	Number	fnd_api.g_miss_num
initial_pickup_location_code	Varchar2(20)	fnd_api.g_miss_char
organization_id	Number	fnd_api.g_miss_num
organization_code	Varchar2(3)	fnd_api.g_miss_char
ultimate_dropoff_location_id	Number	fnd_api.g_miss_num
ultimate_dropoff_location_code	Varchar2(20)	fnd_api.g_miss_char
ultimate_dropoff_date	Date	fnd_api.g_miss_date
customer_id	Number	fnd_api.g_miss_num
customer_number	Varchar2(30)	fnd_api.g_miss_char

Attribute	Type	Default
intmed_ship_to_location_id	Number	fnd_api.g_miss_num
intmed_ship_to_location_code	Varchar2(20)	fnd_api.g_miss_char
pooled_ship_to_location_id	Number	fnd_api.g_miss_num
pooled_ship_to_location_code	Varchar2(20)	fnd_api.g_miss_char
carrier_id	Number	fnd_api.g_miss_num
carrier_code	Varchar2(25)	fnd_api.g_miss_char
ship_method_code	Varchar2(30)	fnd_api.g_miss_char
ship_method_name	Varchar2(80)	fnd_api.g_miss_char
freight_terms_code	Varchar2(30)	fnd_api.g_miss_char
freight_terms_name	Varchar2(80)	fnd_api.g_miss_char
fob_code	Varchar2(30)	fnd_api.g_miss_char
fob_name	Varchar2(80)	fnd_api.g_miss_char
fob_location_id	Number	fnd_api.g_miss_num
fob_location_code	Varchar2(20)	fnd_api.g_miss_char
waybill	Varchar2(30)	fnd_api.g_miss_char
dock_code	Varchar2(30)	fnd_api.g_miss_char
acceptance_flag	Varchar2(1)	fnd_api.g_miss_char
accepted_by	Varchar2(150)	fnd_api.g_miss_char
accepted_date	Date	fnd_api.g_miss_date

Attribute	Type	Default
acknowledged_by	Varchar2(150)	fnd_api.g_miss_char
confirmed_by	Varchar2(150)	fnd_api.g_miss_char
confirm_date	Date	fnd_api.g_miss_date
asn_date_sent	Date	fnd_api.g_miss_date
asn_status_code	Varchar2(15)	fnd_api.g_miss_char
asn_seq_number	Number	fnd_api.g_miss_num
gross_weight	Number	fnd_api.g_miss_num
net_weight	Number	fnd_api.g_miss_num
weight_uom_code	Varchar2(3)	fnd_api.g_miss_char
weight_uom_desc	Varchar2(25)	fnd_api.g_miss_char
volume	Number	fnd_api.g_miss_num
volume_uom_code	Varchar2(3)	fnd_api.g_miss_char
volume_uom_desc	Varchar2(25)	fnd_api.g_miss_char
additional_shipment_info	Varchar2(500)	fnd_api.g_miss_char
currency_code	Varchar2(15)	fnd_api.g_miss_char
currency_name	Varchar2(80)	fnd_api.g_miss_char
prorate_wt_flag	Varchar2(1)	fnd_api.g_miss_char
attribute_category	Varchar2(150)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char
tp_attribute_category	Varchar2(150)	fnd_api.g_miss_char
tp_attribute1	Varchar2(150)	fnd_api.g_miss_char
tp_attribute2	Varchar2(150)	fnd_api.g_miss_char
tp_attribute3	Varchar2(150)	fnd_api.g_miss_char
tp_attribute4	Varchar2(150)	fnd_api.g_miss_char
tp_attribute5	Varchar2(150)	fnd_api.g_miss_char
tp_attribute6	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default
tp_attribute7	Varchar2(150)	fnd_api.g_miss_char
tp_attribute8	Varchar2(150)	fnd_api.g_miss_char
tp_attribute9	Varchar2(150)	fnd_api.g_miss_char
tp_attribute10	Varchar2(150)	fnd_api.g_miss_char
tp_attribute11	Varchar2(150)	fnd_api.g_miss_char
tp_attribute12	Varchar2(150)	fnd_api.g_miss_char
tp_attribute13	Varchar2(150)	fnd_api.g_miss_char
tp_attribute14	Varchar2(150)	fnd_api.g_miss_char
tp_attribute15	Varchar2(150)	fnd_api.g_miss_char
global_attribute_category	Varchar2(30)	fnd_api.g_miss_char
global_attribute1	Varchar2(150)	fnd_api.g_miss_char
global_attribute2	Varchar2(150)	fnd_api.g_miss_char
global_attribute3	Varchar2(150)	fnd_api.g_miss_char
global_attribute4	Varchar2(150)	fnd_api.g_miss_char
global_attribute5	Varchar2(150)	fnd_api.g_miss_char
global_attribute6	Varchar2(150)	fnd_api.g_miss_char
global_attribute7	Varchar2(150)	fnd_api.g_miss_char
global_attribute8	Varchar2(150)	fnd_api.g_miss_char
global_attribute9	Varchar2(150)	fnd_api.g_miss_char
global_attribute10	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default
global_attribute11	Varchar2(150)	fnd_api.g_miss_char
global_attribute12	Varchar2(150)	fnd_api.g_miss_char
global_attribute13	Varchar2(150)	fnd_api.g_miss_char
global_attribute14	Varchar2(150)	fnd_api.g_miss_char
global_attribute15	Varchar2(150)	fnd_api.g_miss_char
global_attribute16	Varchar2(150)	fnd_api.g_miss_char
global_attribute17	Varchar2(150)	fnd_api.g_miss_char
global_attribute18	Varchar2(150)	fnd_api.g_miss_char
global_attribute19	Varchar2(150)	fnd_api.g_miss_char
global_attribute20	Varchar2(150)	fnd_api.g_miss_char
creation_date	Date	fnd_api.g_miss_date
created_by	Number	fnd_api.g_miss_num
last_update_date	Date	fnd_api.g_miss_date
last_updated_by	Number	fnd_api.g_miss_num
last_update_login	Number	fnd_api.g_miss_num
program_application_id	Number	fnd_api.g_miss_num
program_id	Number	fnd_api.g_miss_num
program_update_date	Date	fnd_api.g_miss_date
request_id	Number	fnd_api.g_miss_num

Record Parameter Attribute Validations
delivery_id

Should be a valid unique element of wsh_new_deliveries.delivery_id.

delivery_name

Should be a valid unique element of wsh_new_deliveries.name.

delivery_type

Should be either STANDARD.

loading_order_flag

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type LOADING_ORDER.

loading_order_desc

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type LOADING_ORDER.

initial_pickup_location_id

Should be a valid element of hr_locations.location_id.

initial_pickup_location_code

Should be a valid element of hr_locations.location_code.

organization_id

Should be a valid element of org_organization_definitions.organization_id.

organization_code

Should be a valid element of org_organization_definitions.organization_code.

ultimate_dropoff_location_id

Should be a valid element of hr_locations.location_id.

ultimate_dropoff_location_code

Should be a valid element of hr_locations.location_code.

customer_id

Should be a valid element of ra_customers.customer_id.

customer_code

Should be a valid element of ra_customers.customer_code.

intmed_ship_to_location_id

Should be a valid element of hr_locations.location_id.

intmed_ship_to_location_code

Should be a valid element of hr_locations.location_code.

pooled_ship_to_location_id

Should be a valid element of hr_locations.location_id.

pooled_ship_to_location_code

Should be a valid element of hr_locations.location_code.

ship_method_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

ship_method_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type SHIP_METHOD.

freight_terms_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FREIGHT_TERMS.

freight_terms_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FREIGHT_TERMS.

fob_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FOB.

fob_terms_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FOB.

fob_location_id

Should be a valid element of hr_locations.location_id.

fob_location_code

Should be a valid element of hr_locations.location_code.

weight_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a weight uom class

weight_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a weight uom class.

volume_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a volume uom class.

volume_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a volume uom class.

currency_code

Should be a valid element of fnd_currencies_vl.currency_code.

currency_name

Should be a valid element of `fnd_currencies_vl.currency_name`.

prorate_wt_flag

Should be Y or N as per proration of weight required or not.

Error Handling

Refer to parameters `p_init_msg_list`, `x_msg_count`, and `x_msg_data` on retrieving error messages.

Delivery_Action API Features

The `Delivery_Action` API has the following features:

The `Delivery_Action` procedure enables you to carry out actions on a delivery. It accepts as IN parameters the delivery identifiers, an action code and any additional parameters needed for specific actions, and it returns a completion status.

Note: Before you call this public API, initialize your environment (using `fnd_global.apps_initialize`) in the script or package that you use to call it.

Functional Overview

This API validates the delivery identifiers and performs the following actions specified in `p_action_code` by calling the corresponding private procedures:

- **PLAN/UNPLAN:** Is called to plan or unplan deliveries.
- **WT-VOL:** Is called to calculate weight and volume for the delivery. This action needs an additional parameter `P_WV_OVERRIDE_FLAG`. Regardless if this is set to Y or N, weight and volume re-calculations are done for all deliveries on the trip.
- **CONFIRM:** Is called to confirm delivery. Requires additional parameters starting with `P_SC`
- **RE-OPEN, CLOSE:** Is called to update status of delivery. This action needs an additional parameter `P_ACTUAL_DATE` for actual departure or arrival updates.
- **ASSIGN-TRIP:** Is called to assign a trip to delivery. It requires additional parameters starting with `P_ASG`.
- **UNASSIGN-TRIP:** Is called to unassign trip from delivery.
- **AUTOCREATE-TRIP:** Is called to autcreate trip for the delivery.

- PICK-RELEASE: Launches pick release to release lines related to a delivery.
- DELETE: Is called to delete a delivery.

Procedure Parameter Descriptions

WSH_DELIVERIES_PUB.DELIVERY_ACTION

The following chart describes all parameters used by the public procedure WSH_DELIVERIES_PUB.DELIVERY_ACTION. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERIES_PUB.DELIVERY_ACTION Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_action_code	IN	Varchar2	x
p_delivery_id	IN	Number	-
p_delivery_name	IN	Varchar2	-
p_asg_trip_id	IN	Number	x
p_asg_trip_name	IN	Varchar2	-
p_asg_pickup_stop_id	IN	Number	-
p_asg_pickup_loc_id	IN	Number	-
p_asg_pickup_loc_code	IN	Varchar2	-

Parameter	Usage	Type	Required
p_asg_pickup_arr_date	IN	Date	-
p_asg_pickup_dep_date	IN	Number	-
p_asg_dropoff_stop_id	IN	Number	-
p_asg_dropoff_loc_id	IN	Number	-
p_asg_dropoff_loc_code	IN	Varchar2	-
p_asg_dropoff_arr_date	IN	Date	-
p_asg_dropoff_dep_date	IN	Date	-
p_sc_action_flag	IN	Varchar2	-
p_sc_close_trip_flag	IN	Varchar2	-
p_sc_create_bol_flag	IN	Varchar2	-
p_sc_stage_del_flag	IN	Varchar2	-
p_sc_trip_ship_method	IN	Varchar2	-
p_sc_actual_dep_date	IN	Varchar2	-
p_sc_report_set_id	IN	Number	-
p_sc_report_set_name	IN	Varchar2	-
p_sc_rule_id	IN	Number	-
p_sc_rule_name	IN	Varchar2	-

Parameter	Usage	Type	Required
p_sc_defer_interface_flag	IN	Varchar2	-
p_wv_override_flag	IN	Varchar2	-
x_trip_id	OUT	Varchar2	-
x_trip_name	OUT	Varchar2	-
p_actual_date	IN	Date	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_action_code

Specifies which of the actions of PLAN, UNPLAN, CONFIRM, RE-OPEN, CLOSE, ASSIGN-TRIP, UNASSIGN-TRIP, AUTOCREATE-TRIP, WT-VOL, PICK RELEASE, DELETE the API should perform.

p_delivery_name

Name of delivery on which actions need to be performed.

p_delivery_id

Delivery ID on which actions need to be performed.

Default Value: NULL.

p_asg_trip_id

Trip identifier for assignment of trip to delivery. Used when assigning and un-assigning a delivery to a trip.

Default Value: NULL.

p_trip_name

Name of trip for assignment to delivery. Used when assigning a trip.

Default Value: NULL.

p_asg_pickup_stop_id

Stop ID for pickup assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_pickup_loc_id

Stop location for pickup assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_pickup_loc_code

Stop location code for pickup assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_pickup_arr_date

Stop location arrival date for pickup assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_pickup_dep_date

Stop location departure date for pickup assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_dropoff_stop_id

Stop ID for dropoff assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_dropoff_loc_id

Stop location for dropoff assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_dropoff_loc_code

Stop location code for dropoff assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_dropoff_arr_date

Stop location arrival date for dropoff assignment. Used when assigning a trip.

Default Value: NULL.

p_asg_dropoff_dep_date

Stop location departure date for dropoff assignment. Used when assigning a trip.

Default Value: NULL.

p_sc_action_flag

Ship Confirm option - S, B, T, A, C. Used when ship confirming a delivery.

Default Value: S

p_sc_close_trip_flag

Ship Confirm close trip flag. Used when ship confirming a delivery.

Default Value: N

p_sc_create_bol_flag

Ship Confirm create Bill of Lading flag. Used when ship confirming a delivery.

Default Value: N.

p_sc_rule_id

Used when action is CONFIRM.

Default Value: Null.

p_sc_rule_name

Used when action is CONFIRM.

Default Value: Null.

p_sc_stage_del_flag

Ship Confirm create delivery for stage quantity flag. Used when ship confirming a delivery.

Default Value: Y.

p_sc_trip_ship_method

Ship Confirm trip ship method. Used when ship confirming a delivery.

Default Value: NULL.

p_defer_interface_flag

Ship Confirm flag to submit/defer concurrent requests to INV and OE interfaces.

Default Value: Y

p_wv_override_flag

Override flag for weight volume calculations. Regardless if this is set to Y or N, weight and volume re-calculations are done for all deliveries on the trip

Default Value: N

x_trip_id

ID of autocreated trip.

x_trip_name

Name of autocreated trip.

Error Handling

Refer to the parameters p_init_msg_list, x_msg_count, and x_msg_data for information on retrieving error messages.

Generate_Documents API Features

The Generate_Documents API generates the document set for a delivery. It takes the report_set_name, organization_code, and a table of delivery_names as the input parameter and passes the return_status, message_data and message_count as the output parameters.

Functional Overview

This API is for backward compatibility. A call to this API generates the document set specified in the IN parameter report_set_name for all the deliveries in the IN parameter delivery_names.

Setting Up the Generate_Documents API

Parameter Descriptions

The following chart describes all parameters used by the public Generate_Documents API. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

Generate_Documents Parameters

Parameter	Usage	Type	Required
p_report_set_name	IN	VARCHAR2	-
p_organization_code	IN	VARCHAR2	-
p_delivery_name	IN	WSH_UTIL_CORE. Column_Tab_Type	-
x_msg_count	OUT	NUMBER	-
x_msg_data	OUT	VARCHAR2	-
x_return_status	OUT	VARCHAR2	-

p_report_set_name

This is the report_Set that needs to be generated for all the deliveries specified in p_delivery_name parameter.

p_organization_code

This is the organization code for which the document _set needs to be generated.

p_delivery_name

This is the list of delivery_names for which the document set needs to be generated.

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Contains error message text. If X_msg_count is 1, then this contains the complete message.

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include the following:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

Exceptions Application Program Interface

Exceptions Application Program Interface (API) is a public API that consists of the following procedure in the package WSH_EXCEPTIONS_PUB.

Exception_Action: Enables you to carry out actions on an exception. It accepts as IN parameters the exception identifiers, an action code and returns a completion status.

Get_Exceptions: Enables you to access detailed information about an exception and stores it in an exception table.

Exception_Action API Features

The Exception_Action procedure enables you to carry out actions on an exception. It accepts as IN parameters the exceptions identifiers in a table and an action, and it returns a completion status.

Note: Before you call this public API, initialize your environment (using `fnd_global.apps_initialize`) in the script or package that you use to call it.

Functional Overview

This API validates its input parameters `p_action` and performs the following actions depending on its value:

- PURGE: Used to purge existing exceptions
- LOG: Used to log an exception based on the values in the input exceptions record (`p_exception_rec`)
- CHANGE_STATUS: Used to change the status of an exception

Setting Up the Exception_Actions API

Parameter Descriptions

The following chart describes all parameters used by the public Exception_Actions API. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

Exception_Actions Parameters

Parameter	Usage	Type	Required	Derived	Optional
p_api_version_number	IN	NUMBER	x	-	-
p_init_msg_list	IN	VARCHAR2	-	-	x
p_commit	IN	VARCHAR2	-	-	x
p_action	IN	VARCHAR2	x	-	-
p_validation_level	IN	NUMBER	-	-	x
p_exceptions_rec	IN OUT	XC_ACTION_REC_TYPE	x	-	-
x_msg_count	OUT	NUMBER	-	x	-
x_msg_data	OUT	VARCHAR2	-	x	-
x_return_status	OUT	VARCHAR2	-	x	-

p_api_version_number

Used to compare the version number of the incoming API call to the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000 and Number respectively.

- Default Value: FND_API.G_FALSE

p_commit

Requests that the API update information for you after it completes its function. Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Displays error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_validation_level

This parameter is not currently used.

p_action

This parameter specifies the actions that needs to be performed on the exception. Valid values for this parameter are 'LOG', 'PURGE' and 'CHANGE_STATUS'.

p_exception_rec

This record structure has all the exception specific entries.

The table below describes all parameters that are used by the XC_ACTION_REC_TYPE TYPE record. Additional information on these parameters follows.

XC_ACTION_REC_TYPE

XC_ACTION_REC_TYPE Parameters

Parameter	Type	Default Value
request_id	NUMBER	
batch_id	NUMBER	
exception_id	NUMBER	
exception_name	VARCHAR2	
logging_entity	VARCHAR2	
logging_entity_id	NUMBER	
manually_logged	VARCHAR2	
message	VARCHAR2	
logged_at_location_code	VARCHAR2	
exception_location_code	VARCHAR2	
severity	VARCHAR2	
delivery_name	VARCHAR2	
trip_name	VARCHAR2	
stop_location_id	NUMBER	
delivery_detail_id	NUMBER	
container_name	VARCHAR2	
ord_id	NUMBER	
inventory_item_id	NUMBER	
lot_number	VARCHAR2	

Parameter	Type	Default Value
sublot_number	VARCHAR2	
revision	VARCHAR2	
serial_number	VARCHAR2	
unit_of_measure	VARCHAR2	
quantity	NUMBER	
unit_of_measure2	VARCHAR2	
quantity2	NUMBER	
subinventory	VARCHAR2	
locator_id	NUMBER	
error_message	VARCHAR2	
attribute_category	VARCHAR2	
attribute1	VARCHAR2	
attribute2	VARCHAR2	
attribute3	VARCHAR2	
attribute4	VARCHAR2	
attribute5	VARCHAR2	
attribute6	VARCHAR2	
attribute7	VARCHAR2	
attribute8	VARCHAR2	
attribute9	VARCHAR2	

Parameter	Type	Default Value
attribute10	VARCHAR2	
attribute11	VARCHAR2	
attribute12	VARCHAR2	
attribute13	VARCHAR2	
attribute14	VARCHAR2	
attribute15	VARCHAR2	
departure_date	DATE	
arrival_date	DATE	
exception_type	VARCHAR2	
status	VARCHAR2	
departure_date_to	DATE	
arrival_date_to	DATE	
creation_date	DATE	
creation_date_to	DATE	
data_older_no_of_days	NUMBER	
new_status	VARCHAR2	
caller	VARCHAR2	
phase	NUMBER	

The purge action uses the following fields:

- exception_type: Specifies which type of exception needs to be purged.
- status: Specifies the status of exceptions that should be selected to be purged

- `departure_date_to`: Specifies that exceptions with `departure_dates` before this date should be purged
- `arrival_date_to`: Specifies that exceptions with `arrival_dates` before this date should be purged
- `creation_date`: Specifies that exceptions with `creation_date` same as this date should be purged
- `creation_date_to`: Specifies that exceptions with `creation_dates` before this date should be purged
- `data_older_no_of_days`: Specifies the maximum age of exceptions that should be purged
- `departure_date`: Specifies the `departure_date` of exceptions that should be purged
- `arrival_date`: Specifies the arrival date of exceptions that should be purged
- `logged_at_location_code`: Specifies the `location_code` where the exceptions have been logged that should be purged
- `exception_location_code`: Specifies the `location_code` of the exceptions that should be purged
- `severity`: Specifies the severity of the exceptions that should be purged
- `delivery_name`: Specifies the delivery name of the exception that should be purged
- `exception_name`: Specifies the name of the exception that should be purged
- `logging_entity`: Specifies the entity whose exceptions need to be purged
- `request_id`: Specifies the `request_id` whose exceptions needs to be purged

The following fields are used for the `change_status` action:

- `exception_name`: Specifies the `exception_name` whose status needs to be changed
- `logging_entity`: Specifies the logging entity whose exceptions need to change status
- `logging_entity_id`: Specifies the logging entity id whose exceptions need to change status
- `new_status`: Specifies the new status of the exceptions that satisfy the previous fields

Logging exceptions use the following fields. The value of these fields is directly assigned to the corresponding columns in the `WSH_EXCEPTION` table when the new

exception record is created.

- request_id
- batch_id
- exception_id
- exception_name
- logging_entity
- logging_entity_id
- manually_logged
- message
- logged_at_location_code
- exception_location_code
- severity
- delivery_name
- trip_name
- stop_location_id
- delivery_detail_id
- container_name
- org_id
- inventory_item_id
- lot_number
- subplot_number
- revision
- serial_number
- unit_of_measure
- quantity

- unit_of_measure2
- quantity2
- subinventory
- locator_id
- error_message
- attribute_category
- attribute1
- attribute2
- attribute3
- attribute4
- attribute5
- attribute6
- attribute7
- attribute8
- attribute9
- attribute10
- attribute11
- attribute12
- attribute13
- attribute14
- attribute15
- departure_date
- arrival_date

Get_Exceptions API Features

The Get_Exceptions API fetches the exceptions based on the entity_name and the entity_id passed to it. The fetched details are stored in the table x_exceptions_tab which is the output parameter of the API.

Functional Overview

The Get_Exceptions API takes in p_logging_entity_id and p_logging_entity_name as IN parameters and passes out the exceptions that have been logged for this entity. The valid values of p_logging_entity_name are 'DELIVERY', 'TRIP', 'STOP', 'CONTAINER' and 'DETAIL'. Thus, if p_logging_entity is 'DELIVERY', then p_logging_entity_id should be a valid delivery_id.

Setting Up the Get_Exceptions API

Parameter Descriptions

The following table describes all parameters used by the Get_Exceptions publicAPI. All of the inbound and outbound parameters are listed. Additional information about these parameters follows the table.

Get_Exceptions Parameters

Parameter	Usage	Type	Required	Derived	Optional
p_api_version_number	IN	NUMBER	x	-	-
p_init_message	IN	VARCHAR2	-	-	x
x_return_status	OUT	VARCHAR2	-	-	x
x_msg_count	OUT	NUMBER	-	-	x
x_msg_data	OUT	VARCHAR2	-	x	-
p_logging_entity_id	IN	NUMBER	x	-	-

Parameter	Usage	Type	Required	Derived	Optional
p_logging_entity	IN	VARCHAR2	x	-	-
x_exceptions_table	OUT	XC_TAB_TY PE	x	-	-

p_api_version_number

Used to compare the version number of the incoming API call with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Displays error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_logging_entity_name

Specifies the entity for which we are querying the exception. Valid values are 'DELIVERY', 'TRIP', 'STOP','CONTAINER' and 'DETAIL'.

P_logging_entity_id

The entity Id for which the exceptions details are being fetched.

x_exceptions_tab

The OUT parameter where the details of the exception are stored.

The x_exceptions_tab fields are as follows:

x_exceptions_tab Fields

Parameter	Type	Default Value
exception_id	NUMBER	-
exception_name	VARCHAR2	-
status	VARCHAR2	-

Delivery Details Public Application Program Interface

The Delivery Details public API consists of the following procedures in package **WSH_DELIVERY_DETAILS_PUB**:

Detail_To_Delivery: Enables you to assign or cancel the assignment of a delivery detail to or from a delivery

Split_Line: Splits a delivery detail into 2 delivery details with the requested quantity summed up to the original delivery detail.

Update_Shipping_Attributes: Updates a delivery detail with new information with the **Autocreate_Deliveries** and **Autocreate_Del_Trip** procedures.

Autocreate_Deliveries:Automatically creates deliveries for multiple delivery lines.

Autocreate_Del_Trip:Automatically creates trips and deliveries for multiple delivery lines.

Get_Detail_Status: Provides one of the following statuses for a delivery detail passed: SIC (ship confirmed), PK (packed), or OK (other).

Note: The creation of a delivery detail is not provided in the open API since it should be part of the order entry process.

This section describes how to use the Delivery Details Public API and how it functions in Oracle Shipping Execution.

Detail_To_Delivery API Features

The Detail_To_Delivery procedure performs 2 actions: assign delivery details to a delivery and cancel the assignment of delivery details from a delivery.

Functional Overview

This procedure assigns delivery details to a delivery as specified by IN parameter p_action value ASSIGN. If the action is ASSIGN, either the Delivery ID or the Delivery Name need to be specified. It also cancels delivery details from a delivery based on the IN parameter p_action value UNASSIGN. If the action is UNASSIGN, then you do not need to pass a Delivery ID or a Delivery Name because the procedure will cancel all of the delivery details from the deliveries.

For ASSIGN, the procedure checks the delivery detail status, delivery status, group by attributes defined in Shipping Parameters, Ship To Locations and Ship From Locations. If one of the delivery details is assigned to a delivery, then the procedure continues to the next delivery detail, performs the ASSIGN action to the delivery details, which are not already assigned, and returns an error in the return status. The calling procedure decides whether to commit or rollback. ASSIGN is allowed only for the deliveries in the status of OPEN or PACKED and not Planned. Security rules check to ensure that ship-to and ship-from location information and other optional group by attributes are common between the delivery details and the delivery.

If one of the delivery details is unassigned, then the UNASSIGN return status posts a warning, and the action continues to the next delivery detail. To allow UNASSIGN action, the delivery status must be OPEN and not planned. If there is freight cost for the delivery detail, then a warning is posted in the return status.

Procedure Parameter Descriptions

WSH_DELIVERY_DETAILS_PUB.DETAIL_TO_DELIVERY

The following chart describes all parameters used by the public procedure WSH_DELIVERY_DETAILS_PUB.DETAIL_TO_DELIVERY. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERY_DETAILS_PUB.DETAIL_TO_DELIVERY Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	Y

Parameter	Usage	Type	Required
p_init_msg_list	IN	Varchar2	N
x_return_status	OUT	Varchar2	N
x_msg_count	OUT	Number	N
x_msg_data	OUT	Varchar2	N
p_TabOfDelDets	IN	Table	Y
p_action	IN	Varchar2	Y
p_delivery_id	OUT	Number	N
p_delivery_name	OUT	Varchar2	N

p_api_version_number

Compares the version number of the incoming API call with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. Values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Values include:

- Success: FND_API.G_RET_STS_SUCCESS

- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates the number of error messages that the API has encountered.

x_msg_data

Returns an error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_TabOfDelDets

Table of Delivery Detail ID's of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

p_action

Valid values for the Action are ASSIGN and UNASSIGN.

If the action is ASSIGN, the user needs to pass either the Delivery ID or the Delivery Name to identify which Delivery the Delivery Details are assigned to. If both the Delivery ID and Delivery Name are passed in, only the Delivery ID will be used to identify the Delivery, the Delivery Name will be ignored.

p_delivery_id

The Delivery ID to identify the Delivery, not needed if the action is UNASSIGN.

p_delivery_name

The Delivery Name to identify the Delivery, not needed if the action is UNASSIGN

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Split_Line API Features

The Split_Line API has the following features:

The Split_Line procedure enables you to split a Delivery Detail into two Delivery Details. A Delivery Detail needs to be split when ship partially, or when a backorder occurs.

Functional Overview

This API splits a Delivery Detail into two Delivery Details. Most of the fields in the new Delivery Detail have the same values except that the requested quantity is the split quantity, the shipped quantity is NULL, the backorder quantity is NULL and that the split_from_detail_id points to the original Delivery Detail. After the split, the calling procedure needs to make another call to update the shipped quantity and backordered

quantity to complete the whole transaction.

This procedure checks that the Delivery Detail is valid, the split quantity is not greater than the requested quantity before calling WSH_DELIVERY_DETAILS_ACTIONS.Split_Delivery_Details. Most of the validation is performed in WSH_DELIVERY_DETAILS_ACTIONS.Split_Delivery_Details, which checks that the quantity meets the decimal quantity standard.

Procedure Parameter Descriptions

WSH_DELIVERY_DETAILS_PUB.SPLIT_LINE

The following chart describes all parameters used by the public procedure WSH_DELIVERY_DETAILS_PUB.SPLIT_LINE. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERY_DETAILS_PUB.SPLIT_LINE Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
p_validation_level	IN	Number	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_from_detail_id	IN	Varchar2	x
x_new_detail_id	OUT	Number	-
x_split_quantity	IN OUT	Number	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list. If the `x_msg_count` is greater than 1, then the list of messages must be retrieved using the call `FND_MSG_PUB.GET`. The values are:

- `p_msg_index => I`
- `p_encoded => F`
- `p_data => 1_message`
- `p_msg_index_out => 1_msg_index_out`

where `1_message` and `1_msg_index_out` are local variables of types `Varchar2(2000)` and `Number` respectively.

Default Value: `FND_API.G_FALSE`.

p_commit

Commits the transaction if this parameter is set to `TRUE`.

Default Value: `FND_API.G_FALSE`

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: `FND_API.G_RET_STS_SUCCESS`
- Error: `FND_API.G_RET_STS_ERROR`
- Unexpected Error: `FND_API.G_RET_STS_UNEXP_ERROR`

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the `x_msg_count` is equal to 1, then this contains the actual message.

p_from_detail_id

The Delivery Detail ID of the line to be split.

x_new_detail_id

Delivery Detail ID of the new line being created after splitting the original line.

x_split_quantity

Quantity out of the original Delivery Detail quantity allocated to the new Delivery Detail created after the split.

Error Handling

Refer to parameters `p_init_msg_list`, `x_msg_count`, and `x_msg_data` on retrieving error messages.

Update_Shipping_Attributes API Features

The Update_Shipping API has the following features:

The Update_Shipping_Attributes procedure enables you to modify data in `wsh_delivery_details`.

Functional Overview

This procedure is called from the source system to modify data in `wsh_delivery_details` to reflect any changes made by the source system including pick release, ship confirm, split source line, and other update activities.

The procedure validates the input parameters and updates the delivery details.

Procedure Parameter Descriptions

`WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES`

The following chart describes all parameters used by the public procedure `WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES`. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES Parameters

Parameter	Usage	Type	Required
<code>p_api_version_number</code>	IN	Number	Y
<code>p_init_msg_list</code>	IN	Varchar2	N
<code>p_commit</code>	IN	Varchar2	N
<code>x_return_status</code>	OUT	Varchar2	N
<code>x_msg_count</code>	OUT	Number	N
<code>x_msg_data</code>	OUT	Varchar2	N

Parameter	Usage	Type	Required
p_changed_attributes	IN	Record	Y
p_source_code	IN	Varchar2	Y
p_serial_range_tab	IN	wsh_delivery_details _grp.serialrangetype	N

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE.

p_commit

Commits the transaction if this parameter is set to TRUE.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_changed_attributes

Attributes of ChangedAttributesTabType that are updated in wsh_delivery_details. WSH_DELIVERY_DETAILS_PUB.ChangedAttributesTabType is a table of ChangedAttributeRecType indexed by Binary_Integer. Definition of ChangedAttributeRecType follows.

p_source_code

Code for source system which updates wsh_delivery_details table. Source system can only update delivery details that are created by the same source system. For Update_Shipping_Attributes this should be OE.

p_serial_range_tab

This is of type wsh_glbl_var_strct_grp.ddSerialRangeTabType.

Where, wsh_glbl_var_strct_grp.ddSerialRangeTabType is of type
TYPE ddSerialRangeTabType IS TABLE OF ddSerialRangeRecType
INDEX BY BINARY_INTEGER;

p_serial_range_tab

Attribute	Type	Default Value
delivery_detail_id	NUMBER	NULL
from_serial_number	VARCHAR2	NULL
to_serial_number	VARCHAR2	NULL
quantity	NUMBER	NULL

delivery_detail_id

The delivery detail id for which the serial numbers are entered.

from_serial_number

The starting serial number.

to_serial_number

The ending serial number.

quantity

The serial quantity, between the from_serial_number and the to_serial_number.

Record Parameter Descriptions**ChangedAttributeRecType RECORD DEFINITION**

To encapsulate WSH_DELIVERY_DETAILS table definition and Value column equivalents for ID columns in a PL/SQL record, define ChangedAttributeRecType.

ChangedAttributesRecType RECORD DEFINITION

Attribute	Type	Default Value
source_header_id	Number	fnd_api.g_miss_num
source_line_id	Number	fnd_api.g_miss_num
sold_to_org_id	Number	fnd_api.g_miss_num
customer_number	Number	fnd_api.g_miss_char
sold_to_contact_id	Number	fnd_api.g_miss_num
ship_from_org_id	Number	fnd_api.g_miss_char
ship_from_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_org_id	Number	fnd_api.g_miss_num
ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_contact_id	Number	fnd_api.g_miss_num
deliver_to_org_id	Number	fnd_api.g_miss_num
deliver_to_org_code	Varchar2(3)	fnd_api.g_miss_char
deliver_to_contact_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char

Attribute	Type	Default Value
intmed_ship_to_contact_id	Number	fnd_api.g_miss_num
ship_tolerance_above	Number	fnd_api.g_miss_num
ship_tolerance_below	Number	fnd_api.g_miss_num
ordered_quantity	Number	fnd_api.g_miss_num
ordered_quantity2	Number	fnd_api.g_miss_num
order_quantity_uom	Varchar2(3)	fnd_api.g_miss_char
ordered_quantity_uom2	Varchar2(25)	fnd_api.g_miss_char
ordered_qty_unit_of_measure	Varchar2(25)	fnd_api.g_miss_char
ordered_qty_unit_of_measure 2	Varchar2(25)	fnd_api.g_miss_char
subinventory	Varchar2(10)	fnd_api.g_miss_char
prefered_grade	Varchar2(4)	fnd_api.g_miss_char
revision	Varchar2(3)	fnd_api.g_miss_char
lot_number	Varchar2(30)	fnd_api.g_miss_char
sublot_number	Varchar2(30)	fnd_api.g_miss_char
customer_requested_lot_flag	Varchar2(1)	fnd_api.g_miss_char
serial_number	Varchar2(30)	fnd_api.g_miss_char
locator_id	Number	fnd_api.g_miss_num
date_requested	Date	fnd_api.g_miss_date
date_scheduled	Date	fnd_api.g_miss_date
master_container_item_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
detail_container_item_id	Number	fnd_api.g_miss_num
shipping_method_code	Varchar2(30)	fnd_api.g_miss_char
carrier_id	Number	fnd_api.g_miss_num
freight_terms_code	Varchar2(30)	fnd_api.g_miss_char
freight_terms_name	Varchar2(30)	fnd_api.g_miss_char
freight_carrier_code	Varchar2(30)	fnd_api.g_miss_char
shipment_priority_code	Varchar2(30)	fnd_api.g_miss_char
fob_code	Varchar2(30)	fnd_api.g_miss_char
fob_name	Varchar2(30)	fnd_api.g_miss_char
dep_plan_required_flag	Varchar2(1)	fnd_api.g_miss_char
customer_prod_seq	Varchar2(50)	fnd_api.g_miss_char
customer_dock_code	Varchar2(30)	fnd_api.g_miss_char
gross_weight	Number	fnd_api.g_miss_num
net_weight	Number	fnd_api.g_miss_num
weight_uom_code	Varchar2(3)	fnd_api.g_miss_char
weight_uom_desc	Varchar2(50)	fnd_api.g_miss_char
volume	Number	fnd_api.g_miss_num
volume_uom_code	Varchar2(3)	fnd_api.g_miss_char
volume_uom_desc	Varchar2(50)	fnd_api.g_miss_char
top_model_line_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
ship_set_id	Number	fnd_api.g_miss_num
ato_line_id	Number	fnd_api.g_miss_num
arrival_set_id	Number	fnd_api.g_miss_num
ship_model_complete_flag	Varchar2(1)	fnd_api.g_miss_char
cust_po_number	Varchar2(50)	fnd_api.g_miss_char
released_status	Varchar2(1)	fnd_api.g_miss_char
packing_instructions	Varchar2(2000)	fnd_api.g_miss_char
shipping_instructions	Varchar2(2000)	fnd_api.g_miss_char
container_name	Varchar2(50)	fnd_api.g_miss_char
container_flag	Varchar2(1)	fnd_api.g_miss_char
delivery_detail_id	Number	fnd_api.g_miss_num
shipped_quantity	Number	fnd_api.g_miss_num
shipped_quantity2	Number	fnd_api.g_miss_num
cycle_count_quantity	Number	fnd_api.g_miss_num
cycle_count_quantity2	Number	fnd_api.g_miss_num
tracking_number	Varchar2(30)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char
to_serial_number	Varchar2(30)	fnd_api.g_miss_char

Record Parameter Attribute Validations

source_header_id

Should indicate the source of the header content, for example OM or OKE.

sold_to_org_id

Should be a valid unique element of ra_customers.customer_id.

customer_number

Should be a valid unique element of ra_customers.customer_number.

sold_to_contact_id

Should indicate the person to contact at the sold to location.

ship_from_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_from_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_contact_id

Should indicate the person to contact at the ship to location.

deliver_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

deliver_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

deliver_to_contact_id

Should indicate the person to contact at the delivery to location.

intmed_ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

intmed_ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

intmed_ship_to_contact_id

Should indicate the person to contact at the intermediate ship to location.

ship_tolerance_above

Should be a non-negative number.

ship_tolerance_below

Should be a non-negative number.

ordered_quantity

Should be a non-negative whole number.

ordered_quantity2

Should be a non-negative whole number.

order_quantity_uom

Should be a valid element of mtl_items_uoms_view.uom_code.

ordered_quantity_uom2

Should be a valid element of mtl_items_uoms_view.uom_code.

preferred_grade

Should indicate the preferred grade for the line item.

ordered_qty_unit_of_measure

Should be a valid element of mtl_items_uoms_view.unit_of_measure.

ordered_qty_unit_of_measure2

Should be a valid element of mtl_items_uoms_view.unit_of_measure.

subinventory

Should indicate the subinventory from which the line item is picked.

revision

Should indicate the revision number for the line item.

lot_number

Should indicate the lot number for the line item.

customer_requested_lot_flag

Should indicate that the line item requires a customer requested lot number.

serial_number

Should indicate the serial number for the line item.

locator_id

Should indicate the inventory locator from which the line item is picked.

date_requested

Should indicate the date requested for the line item.

date_scheduled

Should indicate the date scheduled for the line item.

master_container_item_id

Should indicate the master container name into which the line is packed.

detail_container_item_id

Should indicate the parent container name into which the line is packed.

shipping_method_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

carrier_id

Should indicate the carrier name transporting the line item.

freight_terms_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FREIGHT_TERMS.

freight_terms_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FREIGHT_TERMS.

freight_carrier_code

Should indicate the freight carrier code for the carrier transporting the line item.

shipment_priority_code

Should indicate the shipment priority assigned to the line item.

fob_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FOB.

fob_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FOB.

dep_plan_required_flag

Should be Y or N.

customer_prod_seq

Should indicate the customer production sequence number assigned to the line item.

customer_dock_code

Should indicate the customer dock code for the line item.

gross_weight

Should be a non-negative number.

net_weight

Should be a non-negative number.

weight_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a weight uom class

weight_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a weight uom class.

volume

Should be a non-negative number.

volume_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a volume uom class.

volume_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a volume uom class.

top_model_line_id

If the item is included as part of a model, this should indicate the top model line name for the line item.

ship_set_id

If the item is included as part of a ship set, this should indicate the ship set number to which the line item is assigned.

ato_line_id

If the item is included as part of a ATO model, this should indicate the model line name to which the line item is assigned.

arrival_set_id

If the item is included as part of a arrival ship set, this should indicate the arrival set number to which the line item is assigned.

ship_model_complete_flag

Should be Y or N.

cust_po_number

Should indicate the purchase order number for the line item.

released_status

Should be one of R, S, Y, or X.

packing_instructions

Should indicate the packing instructions included for the line item.

shipping_instructions

Should indicate the shipping instructions included for the line item.

container_name

Should indicate the container name of the container.

container_flag

Should indicate that the line item is a container.

delivery_detail_id

Should indicate the delivery detail id for the line.

tracking_number

Should indicate the tracking number for the line item.

shipped_quantity

Should be a non-negative number.

shipped_quantity2

Should be a non-negative number.

cycle_count_quantity

Should be a non-negative number.

cycle_count_quantity2

Should be a non-negative number.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Autocreate_Deliveries API Features

The Autocreate_Deliveries API has the following features:

The Autocreate_Deliveries procedure is called from the source system to automatically create deliveries for multiple delivery lines.

The procedure checks that the attribute parameters passed in are valid and perform value to ID conversion while validating.

Procedure Parameter Descriptions

WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DELIVERIES

The following chart describes all parameters used by the public procedure WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DELIVERIES. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DELIVERIES Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-

Parameter	Usage	Type	Required
x_msg_data	OUT	Varchar2	-
p_line_rows	IN	Table	x
x_del_rows	OUT	Table	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initializes the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE.

p_commit

Commits the transaction if this parameter is set to TRUE.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_line_rows

Table of Delivery Details of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

x_del_rows

Table of Deliveries of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Autocreate_Del_Trip API Features

The Autocreate_Del_Trip API has the following features:

The Autocreate_Del_Trip procedure is called from the source system to automatically create trips and deliveries for multiple delivery lines.

The procedure also checks that the attribute parameters passed in are valid and perform value to ID conversion while validating.

Procedure Parameter Descriptions

WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DEL_TRIP

The following chart describes all parameters used by the public procedure WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DEL_TRIP. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DEL_TRIP Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-

Parameter	Usage	Type	Required
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_line_rows	IN	Table	x
x_del_rows	OUT	Table	x
x_trip_id	OUT	Number	x
x_trip_name	OUT	Varchar2	x

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE.

p_commit

Commits the transaction if this parameter is set to TRUE.

Default Value: FND_API.G_FALSE.

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_line_rows

Table of Delivery Details of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

x_del_rows

Table of Deliveries of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

x_trip_id

ID of autocreated trip.

x_trip_name

Name of autocreated trip.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Get_Detail_Status API Features

The GET_DETAIL_STATUS API consists of the following feature:

Get_Detail_Status API provides the status of delivery detail.

Functional Overview

Get_Detail_Status procedure provides any one of the following statuses for a delivery detail passed:

- SIC (Ship Confirmed) - If a delivery detail is assigned to Confirmed/In-Transit/Closed delivery
- PK (Packed) - If a delivery detail is packed

- OK - Other than above two conditions

Procedure Parameter Descriptions

WSH_DELIVERY_DETAILS_PUB.Get_Detail_Status

The following chart describes all parameters used by the public procedure WSH_DELIVERY_DETAILS_PUB.Get_Detail_Status. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

Parameter Name	Usage	Type	Required	Derived
p_delivery_detail_id	IN	Number	x	
x_line_status	OUT	VARCHAR2	-	
x_return_status	OUT	VARCHAR2	-	

p_delivery_detail_id

Delivery detail id for which the status is required.

x_line_status

Status of delivery (SIC/PK/OK)

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

Standard Validation

No Validation done for input parameter passed.

Error Handling

None.

The API returns the following return statuses to the calling module:

Condition	Message return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

Update_Shipping_Attributes (overloaded version) API

The Update_Shipping_Attributes API consists of the following features:

The Update_Shipping_Attributes procedure allows you to modify data in wsh_delivery_details table for delivery detail(s) passed.

Functional Overview

The Update_Shipping_Attributes (overloaded version) API allows you to update the delivery detail information in wsh_delivery_details table.

Procedure Parameter Descriptions

WSH_DELIVERY_DETAILS.Update_Shipping_Attributes

The following chart describes all parameters used by the public procedure WSH_DELIVERY_DETAILS.Update_Shipping_Attributes. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

Parameter Name	Usage	Type	Required	Derived
p_api_version_number	IN	NUMBER	x	
p_init_msg_list	IN	VARCHAR2	-	
p_commit	IN	VARCHAR2	-	
x_return_status	OUT	VARCHAR2	-	
x_msg_count	OUT	NUMBER	-	
x_msg_data	OUT	VARCHAR2	-	

Parameter Name	Usage	Type	Required	Derived
p_changed_attributes	IN	WSH_DELIVERY_DETAILS_PUB.ChangedAttributeTabType	x	
p_source_code	IN	VARCHAR2	x	
p_container_flag	IN	VARCHAR2	-	
p_serial_range_tab	IN	WSH_GLBL_VAR_STRCT_GRP.ddSerialRangeTabType	-	

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are: where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively. Default Value: FND_API.G_FALSE

- p_msg_index => I
- p_encoded => F
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

p_commit

Requests that the API update information for you after it completes its function. Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates the number of error messages the API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_changed_attributes

Attributes of ChangedAttributesTabType that are updated in wsh_delivery_details. WSH_DELIVERY_DETAILS_PUB.ChangedAttributesTabType is a table of ChangedAttributesRecType indexed by Binary_Integer. Definition of ChangedAttributesRecType follows.

p_source_code

Code for source system which updates wsh_delivery_details table. Source system can only update delivery details that are created by the same source system. For Update_Shipping_Attributes this should be OE.

p_container_flag

Y or N depending on whether the delivery detail is Container/content line.

Record Parameter Descriptions

ChangedAttributesRecType RECORD DEFINITION. To encapsulate WSH_DELIVERY_DETAILS table definition and Value column equivalents for ID columns in a PL/SQL record, define ChangedAttributesRecType.

Attribute	Type	Default Value
source_header_id	Number	fnd_api.g_miss_num
source_line_id	Number	fnd_api.g_miss_num
sold_to_org_id	Number	fnd_api.g_miss_num
customer_number	Number	fnd_api.g_miss_char
sold_to_contact_id	Number	fnd_api.g_miss_num
ship_from_org_id	Number	fnd_api.g_miss_char

Attribute	Type	Default Value
ship_from_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_org_id	Number	fnd_api.g_miss_num
ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_contact_id	Number	fnd_api.g_miss_num
deliver_to_org_id	Number	fnd_api.g_miss_num
deliver_to_org_code	Varchar2(3)	fnd_api.g_miss_char
deliver_to_contact_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
intmed_ship_to_contact_id	Number	fnd_api.g_miss_num
ship_tolerance_above	Number	fnd_api.g_miss_num
ship_tolerance_below	Number	fnd_api.g_miss_num
ordered_quantity	Number	fnd_api.g_miss_num
ordered_quantity2	Number	fnd_api.g_miss_num
order_quantity_uom	Varchar2(3)	fnd_api.g_miss_char
ordered_quantity_uom2	Varchar2(25)	fnd_api.g_miss_char
ordered_qty_unit_of_measure	Varchar2(25)	fnd_api.g_miss_char
ordered_qty_unit_of_measure 2	Varchar2(25)	fnd_api.g_miss_char
subinventory	Varchar2(10)	fnd_api.g_miss_char
preferred_grade	Varchar2(4)	fnd_api.g_miss_char

Attribute	Type	Default Value
revision	Varchar2(3)	fnd_api.g_miss_char
lot_number	Varchar2(30)	fnd_api.g_miss_char
sublot_number	Varchar2(30)	fnd_api.g_miss_char
customer_requested_lot_flag	Varchar2(1)	fnd_api.g_miss_char
serial_number	Varchar2(30)	fnd_api.g_miss_char
locator_id	Number	fnd_api.g_miss_num
date_requested	Date	fnd_api.g_miss_date
date_scheduled	Date	fnd_api.g_miss_date
master_container_item_id	Number	fnd_api.g_miss_num
detail_container_item_id	Number	fnd_api.g_miss_num
shipping_method_code	Varchar2(30)	fnd_api.g_miss_char
carrier_id	Number	fnd_api.g_miss_num
freight_terms_code	Varchar2(30)	fnd_api.g_miss_char
freight_terms_name	Varchar2(30)	fnd_api.g_miss_char
freight_carrier_code	Varchar2(30)	fnd_api.g_miss_char
shipment_priority_code	Varchar2(30)	fnd_api.g_miss_char
fob_code	Varchar2(30)	fnd_api.g_miss_char
fob_name	Varchar2(30)	fnd_api.g_miss_char
dep_plan_required_flag	Varchar2(1)	fnd_api.g_miss_char
customer_prod_seq	Varchar2(50)	fnd_api.g_miss_char

Attribute	Type	Default Value
customer_dock_code	Varchar2(30)	fnd_api.g_miss_char
gross_weight	Number	fnd_api.g_miss_num
net_weight	Number	fnd_api.g_miss_num
weight_uom_code	Varchar2(3)	fnd_api.g_miss_char
weight_uom_desc	Varchar2(50)	fnd_api.g_miss_char
volume	Number	fnd_api.g_miss_num
volume_uom_code	Varchar2(3)	fnd_api.g_miss_char
volume_uom_desc	Varchar2(50)	fnd_api.g_miss_char
top_model_line_id	Number	fnd_api.g_miss_num
ship_set_id	Number	fnd_api.g_miss_num
ato_line_id	Number	fnd_api.g_miss_num
arrival_set_id	Number	fnd_api.g_miss_num
ship_model_complete_flag	Varchar2(1)	fnd_api.g_miss_char
cust_po_number	Varchar2(50)	fnd_api.g_miss_char
released_status	Varchar2(1)	fnd_api.g_miss_char
packing_instructions	Varchar2(2000)	fnd_api.g_miss_char
shipping_instructions	Varchar2(2000)	fnd_api.g_miss_char
container_name	Varchar2(50)	fnd_api.g_miss_char
container_flag	Varchar2(1)	fnd_api.g_miss_char
delivery_detail_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
shipped_quantity	Number	fnd_api.g_miss_num
shipped_quantity2	Number	fnd_api.g_miss_num
cycle_count_quantity	Number	fnd_api.g_miss_num
cycle_count_quantity2	Number	fnd_api.g_miss_num
tracking_number	Varchar2(30)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
to_serial_number	Varchar2(30)	fnd_api.g_miss_char

Record Parameter Attribute Validations

source_header_id

Should indicate the unique element of oe_order_headers_all.header_id

source_line_id

Should indicate the unique element of oe_order_lines_all.line_id

sold_to_org_id

Should be a valid unique element of hz_cust_accounts.cust_account_id.

customer_number

Should be a valid unique element of hz_cust_accounts.account_number

sold_to_contact_id

Should indicate contact person at the sold to location.

ship_from_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_from_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_contact_id

Should indicate contact person at the ship to location.

deliver_to_org_id Should be a valid element of org_organization_definitions.organization_id.

deliver_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

deliver_to_contact_id

Should indicate contact person at the delivery to location.

intmed_ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

intmed_ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

intmed_ship_to_contact_id

Should indicate contact person at the intermediate ship to location.

ship_tolerance_above

Should be a non-negative number.

ship_tolerance_below

Should be a non-negative number.

ordered_quantity

Should be a non-negative whole number.

ordered_quantity2

Should be a non-negative whole number.

order_quantity_uom

Should be a valid element of mtl_items_uoms_view.uom_code.

ordered_quantity_uom2

Should be a valid element of mtl_items_uoms_view.uom_code.

preferred_grade

Should indicate the preferred grade for the line item.

ordered_qty_unit_of_measure Should be a valid element of mtl_items_uoms_view.unit_of_measure.

ordered_qty_unit_of_measure2

Should be a valid element of mtl_items_uoms_view.unit_of_measure.

subinventory

Should indicate the subinventory from which the line item is picked. revision Should indicate the revision number for the line item.

lot_number

Should indicate the lot number for the line item.

customer_requested_lot_flag

Should indicate that the line item requires a customer requested lot number.

serial_number

Should indicate the serial number for the line item.

locator_id

Should indicate the inventory locator from which the line item is picked.

date_requested

Should indicate the date requested for the line item.

date_scheduled

Should indicate the date scheduled for the line item.

master_container_item_id

Should indicate the master container name into which the line is packed.

detail_container_item_id

Should indicate the parent container name into which the line is packed.

shipping_method_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

carrier_id

Should indicate the name of the carrier transporting the line item.

freight_terms_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FREIGHT_TERMS.

freight_terms_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FREIGHT_TERMS.

freight_carrier_code

Should indicate the freight carrier code for the carrier transporting the line item.

shipment_priority_code

Should indicate the shipment priority assigned to the line item.

fob_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FOB.

fob_name Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FOB.

dep_plan_required_flag

Should be Y or N.

customer_prod_seq

Should indicate the customer production sequence number assigned to the line item.

customer_dock_code

Should indicate the customer dock code for the line item.

gross_weight

Should be a non-negative number.

net_weight

Should be a non-negative number.

weight_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a weight uom class.

weight_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a weight uom class.

volume

Should be a non-negative number.

volume_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a volume uom class.

volume_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a volume uom class.

top_model_line_id

If the item is included as part of a model, this should indicate the top model line name for the line item.

ship_set_id

If the item is included as part of a ship set, this should indicate the ship set number to which the line item is assigned.

ato_line_id

If the item is included as part of a ATO model, this should indicate the model line name to which the line item is assigned.

arrival_set_id

If the item is included as part of a arrival ship set, this should indicate the arrival set number to which the line item is assigned.

ship_model_complete_flag Should be Y or N.

cust_po_number

Should indicate the purchase order number for the line item.

released_status

Should be one of R, S, Y, or X.

packing_instructions

Should indicate the packing instructions included for the line item.

shipping_instructions

Should indicate the shipping instructions included for the line item.

container_name

Should indicate the container name of the container.

container_flag

Should indicate that the line item is a container.

delivery_detail_id

Should indicate the delivery detail id for the line.

tracking_number

Should indicate the tracking number for the line item.

shipped_quantity

Should be a non-negative number.

shipped_quantity2

Should be a non-negative number.

cycle_count_quantity

Should be a non-negative number.

cycle_count_quantity2

Should be a non-negative number.

Standard Validation

Oracle Shipping Execution validates all required columns in the Update_Shipping_Attributes API. For specific information on the data referenced by these columns, see *Oracle Electronic Technical Reference Manual (eTRM)*

API checks whether delivery detail information passed can be updated in wsh_delivery_details table. If delivery details passed to the API are valid then API updates the same in wsh_delivery_details table. If any invalid data is passed to the API appropriate Warning/Error is returned from API.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

If any validation fails, the API returns the error status to the calling module. API return the following return statuses to the calling module :

Condition	Message return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

Init_Changed_Attribute_Rec API Features

The INIT_CHANGED_ATTRIBUTE_REC API consists of the following feature:

The API is used to initialize the attributes of Changed_Attribute_Rec record type to FND_API.G_MISS values.

Functional Overview

The API is used to initialize the attributes of Changed_Attribute_Rec record type to FND_API.G_MISS values. This API is used before calling WSH_DELIVERY_DETAILS_PUB.update_shipping_attributes. Pass Changed_Attribute_Rec record type to this API for initializing the values with FND_API.G_MISS_CHAR/ FND_API.G_MISS_NUM/ FND_API.G_MISS_DATE.

Procedure Parameter Descriptions

The following table lists all parameters used by the public API INIT_CHANGED_ATTRIBUTE_REC. All of the inbound and outbound parameters are listed. Additional information on these parameters follows the table.

Parameter Name	Usage	Type	Required	Derived
p_init_rec	IN OUT	WSH_DELIVERY_DETAILS_PUB.ChangedAttributeRecType	x	-
x_return_status	OUT	VARCHAR2	-	-

p_init_rec

Attributes of ChangedAttributesRecType that has to be initialized. Definition of ChangedAttributesRecType follows.

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

Record Parameter Descriptions

ChangedAttributesRecType RECORD DEFINITION. To encapsulate WSH_DELIVERY_DETAILS table definition and Value column equivalents for ID columns in a PL/SQL record, define ChangedAttributesRecType.

Attribute	Type	Default Value
source_header_id	Number	fnd_api.g_miss_num
source_line_id	Number	fnd_api.g_miss_num
sold_to_org_id	Number	fnd_api.g_miss_num
customer_number	Number	fnd_api.g_miss_char
sold_to_contact_id	Number	fnd_api.g_miss_num
ship_from_org_id	Number	fnd_api.g_miss_char
ship_from_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_org_id	Number	fnd_api.g_miss_num
ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_contact_id	Number	fnd_api.g_miss_num
deliver_to_org_id	Number	fnd_api.g_miss_num
deliver_to_org_code	Varchar2(3)	fnd_api.g_miss_char
deliver_to_contact_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
intmed_ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
intmed_ship_to_contact_id	Number	fnd_api.g_miss_num
ship_tolerance_above	Number	fnd_api.g_miss_num
ship_tolerance_below	Number	fnd_api.g_miss_num
ordered_quantity	Number	fnd_api.g_miss_num
ordered_quantity2	Number	fnd_api.g_miss_num
order_quantity_uom	Varchar2(3)	fnd_api.g_miss_char
ordered_quantity_uom2	Varchar2(25)	fnd_api.g_miss_char
ordered_qty_unit_of_measure	Varchar2(25)	fnd_api.g_miss_char
ordered_qty_unit_of_measure2	Varchar2(25)	fnd_api.g_miss_char
subinventory	Varchar2(10)	fnd_api.g_miss_char
prefered_grade	Varchar2(4)	fnd_api.g_miss_char
revision	Varchar2(3)	fnd_api.g_miss_char
lot_number	Varchar2(30)	fnd_api.g_miss_char
sublot_number	Varchar2(30)	fnd_api.g_miss_char
customer_requested_lot_flag	Varchar2(1)	fnd_api.g_miss_char

Attribute	Type	Default Value
serial_number	Varchar2(30)	fnd_api.g_miss_char
locator_id	Number	fnd_api.g_miss_num
date_requested	Date	fnd_api.g_miss_date
date_scheduled	Date	fnd_api.g_miss_date
master_container_item_id	Number	fnd_api.g_miss_num
detail_container_item_id	Number	fnd_api.g_miss_num
shipping_method_code	Varchar2(30)	fnd_api.g_miss_char
carrier_id	Number	fnd_api.g_miss_num
freight_terms_code	Varchar2(30)	fnd_api.g_miss_char
freight_terms_name	Varchar2(30)	fnd_api.g_miss_char
freight_carrier_code	Varchar2(30)	fnd_api.g_miss_char
shipment_priority_code	Varchar2(30)	fnd_api.g_miss_char
fob_code	Varchar2(30)	fnd_api.g_miss_char
fob_name	Varchar2(30)	fnd_api.g_miss_char
dep_plan_required_flag	Varchar2(1)	fnd_api.g_miss_char
customer_prod_seq	Varchar2(50)	fnd_api.g_miss_char

Attribute	Type	Default Value
customer_dock_code	Varchar2(30)	fnd_api.g_miss_char
gross_weight	Number	fnd_api.g_miss_num
net_weight	Number	fnd_api.g_miss_num
weight_uom_code	Varchar2(3)	fnd_api.g_miss_char
weight_uom_desc	Varchar2(50)	fnd_api.g_miss_char
volume	Number	fnd_api.g_miss_num
volume_uom_code	Varchar2(3)	fnd_api.g_miss_char
volume_uom_desc	Varchar2(50)	fnd_api.g_miss_char
top_model_line_id	Number	fnd_api.g_miss_num
ship_set_id	Number	fnd_api.g_miss_num
ato_line_id	Number	fnd_api.g_miss_num
arrival_set_id	Number	fnd_api.g_miss_num
ship_model_complete_flag	Varchar2(1)	fnd_api.g_miss_char
cust_po_number	Varchar2(50)	fnd_api.g_miss_char
released_status	Varchar2(1)	fnd_api.g_miss_char
packing_instructions	Varchar2(2000)	fnd_api.g_miss_char

Attribute	Type	Default Value
shipping_instructions	Varchar2(2000)	fnd_api.g_miss_char
container_name	Varchar2(50)	fnd_api.g_miss_char
container_flag	Varchar2(1)	fnd_api.g_miss_char
delivery_detail_id	Number	fnd_api.g_miss_num
shipped_quantity	Number	fnd_api.g_miss_num
shipped_quantity2	Number	fnd_api.g_miss_num
cycle_count_quantity	Number	fnd_api.g_miss_num
cycle_count_quantity2	Number	fnd_api.g_miss_num
tracking_number	Varchar2(30)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char
to_serial_number	Varchar2(30)	fnd_api.g_miss_char

Record Parameter Attribute Validations

source_header_id

Should be a valid unique element of oe_order_headers_all.header_id.

source_line_id

Should be a valid unique element of oe_order_lines_all.line_id.

sold_to_org_id

Should be a valid unique element of hz_cust_accounts.cust_account_id.

customer_number

Should be a valid unique element of hz_cust_accounts.account_number.

sold_to_contact_id

Should indicate contact person at the sold to location.

ship_from_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_from_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_contact_id

Should indicate the contact person to contact at the ship to location.

deliver_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

deliver_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

deliver_to_contact_id

Should indicate contact person at the delivery to location.

intmed_ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

intmed_ship_to_org_code Should be a valid element of
org_organization_definitions.organization_code.

intmed_ship_to_contact_id

Should indicate contact person at the intermediate ship to location.

ship_tolerance_above

Should be a non-negative number.

ship_tolerance_below

Should be a non-negative number.

ordered_quantity

Should be a non-negative whole number.

ordered_quantity2

Should be a non-negative whole number.

order_quantity_uom

Should be a valid element of mtl_items_uoms_view.uom_code.

ordered_quantity_uom2

Should be a valid element of mtl_items_uoms_view.uom_code.

preferred_grade

Should indicate the preferred grade for the line item.

ordered_qty_unit_of_measure

Should be a valid element of mtl_items_uoms_view.unit_of_measure.

ordered_qty_unit_of_measure2

Should be a valid element of mtl_items_uoms_view.unit_of_measure.

subinventory

Should indicate the subinventory from which the line item is picked.

revision

Should indicate the revision number for the line item.

lot_number

Should indicate the lot number for the line item.

customer_requested_lot_flag

Should indicate that the line item requires a customer requested lot number.

serial_number

Should indicate the serial number for the line item.

locator_id

Should indicate the inventory locator from which the line item is picked.

date_requested

Should indicate the date requested for the line item.

date_scheduled

Should indicate the date scheduled for the line item.

master_container_item_id

Should indicate the master container name into which the line is packed.

detail_container_item_id

Should indicate the parent container name into which the line is packed.

shipping_method_code Should be a valid element of
fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

carrier_id

Should indicate the name of the carrier name transporting the line item.

freight_terms_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type
FREIGHT_TERMS.

freight_terms_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FREIGHT_TERMS.

freight_carrier_code

Should indicate the freight carrier code for the carrier transporting the line item.

shipment_priority_code

Should indicate the shipment priority assigned to the line item.

fob_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FOB.

fob_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FOB.

dep_plan_required_flag

Should be Y or N.

customer_prod_seq

Should indicate the customer production sequence number assigned to the line item.

customer_dock_code

Should indicate the customer dock code for the line item.

gross_weight

Should be a non-negative number.

net_weight

Should be a non-negative number.

weight_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a weight uom class

weight_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a weight uom class.

volume

Should be a non-negative number.

volume_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a volume uom class.

volume_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a volume uom class.

top_model_line_id

If the item is included as part of a model, this should indicate the top model line name for the line item.

ship_set_id

If the item is included as part of a ship set, this should indicate the ship set number to which the line item is assigned.

ato_line_id

If the item is included as part of a ATO model, this should indicate the model line name to which the line item is assigned.

arrival_set_id

If the item is included as part of a arrival ship set, this should indicate the arrival set number to which the line item is assigned.

ship_model_complete_flag

Should be Y or N.

cust_po_number

Should indicate the purchase order number for the line item.

released_status

Should be one of R, S, Y, or X.

packing_instructions

Should indicate the packing instructions included for the line item.

Error Handling

If any Oracle Exception occurs, the API will return the error status to the calling module.

The API returns the following statuses to the calling module:

Condition	Message Return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

AUTOCREATE_DEL_TRIP (overloaded version) API Features

The WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DEL_TRIP API consists of the following features:

- It is called from the source system to automatically create trips and deliveries for multiple delivery lines.
- It also checks that the attribute parameters passed in are valid and perform the value to ID conversion while validating.
- The procedure also returns all deliveries and trips created for all delivery detail lines passed to the API.

Functional Overview

AUTOCREATE_DEL_TRIP (overloaded version) API is called from the source system to automatically create trips and deliveries for multiple delivery lines. The procedure also checks that the attribute parameters passed in are valid and perform value to ID conversion while validating. It returns all deliveries and trips created for all delivery detail lines passed to the API

Procedure Parameter Descriptions

The following table lists all parameters used by the public API WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DEL_TRIP. All of the inbound and outbound parameters are listed. Additional information on these parameters follows the table.

Parameter Name	Usage	Type	Required	Derived
p_api_version_number	IN	Number	x	
p_init_msg_list	IN	Varchar2	-	
p_commit	IN	Varchar2	-	
x_return_status	OUT	Varchar2	-	
x_msg_count	OUT	Number	-	
x_msg_data	OUT	Varchar2	-	
p_line_rows	IN	Table	x	

Parameter Name	Usage	Type	Required	Derived
x_del_rows	OUT	Table	-	
x_trip_rows	OUT	Table	-	

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE.

p_commit

Commits the transaction if this parameter is set to TRUE.

Default Value: FND_API.G_FALSE.

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the `x_msg_count` is equal to 1, then this contains the actual message.

p_line_rows

Table of Delivery Details of type `WSH_UTIL_CORE.ID_TAB_TYPE`, which is a table of type Number indexed by binary integers.

x_del_rows

Table of Deliveries of type `WSH_UTIL_CORE.ID_TAB_TYPE` which is a table of type Number indexed by binary integers.

x_trip_rows

Table of trips of type `WSH_UTIL_CORE.ID_TAB_TYPE` which is a table of type Number indexed by binary integers.

Standard Validation

Oracle Transportation Execution validates all required columns in the the API `WSH_DELIVERY_DETAILS_PUB.AUTOCREATE_DEL_TRIP`. For specific information on the data referenced by these columns, see:

Oracle Electronic Technical Reference Manual (eTRM)

API checks that the attribute parameters passed in are valid and perform value to ID conversion while validating.

Error Handling

If any validation fails, the API will return the error status to the calling module. The API processes the rows and reports the following values for every record:

Condition	Message Return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

Container Public Application Program Interface

The Container Public Application Program Interface (API) is a public API that consists of the four following procedures in package `WSH_CONTAINER_PUB`:

Create_Containers: Enables you to create a new container record.

Update_Container: Updates an existing container record.

Auto_Pack: Lets you autopack containers.

Container_Actions: Enables you to perform certain actions on a container.

This section describes how to use the Container Public API and how it functions in Oracle Shipping Execution.

Create_Containers API Features

The Create_Containers API has the following features.

The Create_Containers procedure takes in a container item id or container item name and other necessary parameters to create one or more containers and creates the required containers. It returns a table of container instance ids (delivery detail ids) along with the standard out parameters.

Functional Overview

This API creates a new container record in WSH_DELIVERY_DETAILS. It Inserts container information into WSH_DELIVERY_DETAILS and returns a table of the newly created container IDs.

Procedure Parameter Descriptions

WSH_CONTAINER_PUB.CREATE_CONTAINERS

The following chart describes all parameters used by the public procedure WSH_CONTAINER_PUB.CREATE_CONTAINERS. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_CONTAINER_PUB.CREATE_CONTAINERS Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
p_validation_level	IN	Number	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-

Parameter	Usage	Type	Required
x_msg_data	OUT	Varchar2	-
p_container_item_id	IN	Number	-
p_container_item_name	IN	Varchar2	x
p_organization_id	IN	Number	-
p_organization_code	IN	Varchar2	-
p_name_prefix	IN	Varchar2	-
p_name_suffix	IN	Varchar2	-
p_base_number	IN	Number	x
p_num_digits	IN	Number	-
p_quantity	IN	Number	-
p_container_name	IN	Varchar2	-
p_action	IN	Varchar2	-
x_container_ids	OUT	Table	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message

- `p_msg_index_out => 1_msg_index_out`

where `1_message` and `1_msg_index_out` are local variables of types `Varchar2(2000)` and `Number` respectively.

Default Value: `FND_API.G_FALSE`

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: `FND_API.G_RET_STS_SUCCESS`
- Error: `FND_API.G_RET_STS_ERROR`
- Unexpected Error: `FND_API.G_RET_STS_UNEXP_ERROR`

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the `x_msg_count` is equal to 1, then this contains the actual message.

p_container_item_id

Key flexfield Id for the container.

p_container_item_name

Flexfield name for the container.

p_organization_id

Organization ID for the container.

p_organization_code

Organization code for the container.

p_name_prefix

Prefix of the container name.

p_name_suffix

Suffix of the container name.

p_name_prefix

Prefix of the container name.

p_base_number

Starting number for the numeric portion of the container name.

p_num_digits

Precision for the number of digits.

p_quantity

Number of containers created.

p_container_name

Container name if creating just one container.

x_container_ids

Table of the newly created container IDs of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Update Container API Features

The Trip_Action API has the following features:

The Update_Container procedure enables you to update an existing container. It accepts as IN parameters the container information and the name/ID of the container being updated.

Functional Overview

The API updates an existing container record in WSH_DELIVERY_DETAILS with the attributes input in the container rec type. The name or the ID of the container being updated is passed as IN parameters p_container_name and p_cont_instance_id respectively.

The API validates the input record attributes before updating a container record.

WSH_DELIVERY_DETAILS

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
p_validation_level	IN	Varchar2	-

Parameter	Usage	Type	Required
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_container_rec	IN	Record	x

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_container_rec

Attributes of the container entity of type ChangedAttributeRecType. These attributes are updated in WSH_DELIVERY_DETAILS. Definition of ChangedAttributeRecType follows.

Record Parameter Descriptions

TRIP_STOP_PUB_REC_TYPE RECORD DEFINITION

To encapsulate WSH_DELIVERY_DETAILS table definition and Value column equivalents for ID columns in a PL/SQL record, define ChangedAttributeRecType and use this to pass trip information to the Update_Container routine.

TRIP_STOP_PUB_REC_TYPE RECORD DEFINITION

Attribute	Type	Default Value
source_header_id	Number	fnd_api.g_miss_num
source_line_id	Number	fnd_api.g_miss_num
sold_to_org_id	Number	fnd_api.g_miss_num
customer_number	Number	fnd_api.g_miss_char
sold_to_contact_id	Number	fnd_api.g_miss_num
ship_from_org_id	Number	fnd_api.g_miss_char
ship_from_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_org_id	Number	fnd_api.g_miss_num
ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
ship_to_contact_id	Number	fnd_api.g_miss_num
deliver_to_org_id	Number	fnd_api.g_miss_num
deliver_to_org_code	Varchar2(3)	fnd_api.g_miss_char

Attribute	Type	Default Value
deliver_to_contact_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_id	Number	fnd_api.g_miss_num
intmed_ship_to_org_code	Varchar2(3)	fnd_api.g_miss_char
intmed_ship_to_contact_id	Number	fnd_api.g_miss_num
ship_tolerance_above	Number	fnd_api.g_miss_num
ship_tolerance_below	Number	fnd_api.g_miss_num
ordered_quantity	Number	fnd_api.g_miss_num
order_quantity_uom	Varchar2(3)	fnd_api.g_miss_char
ordered_qty_unit_of_measure	Varchar2(25)	fnd_api.g_miss_char
subinventory	Varchar2(10)	fnd_api.g_miss_char
revision	Varchar2(3)	fnd_api.g_miss_char
lot_number	Varchar2(30)	fnd_api.g_miss_char
customer_requested_lot_flag	Varchar2(1)	fnd_api.g_miss_char
serial_number	Varchar2(30)	fnd_api.g_miss_char
locator_id	Number	fnd_api.g_miss_num
date_requested	Date	fnd_api.g_miss_date
date_scheduled	Date	fnd_api.g_miss_date
master_container_item_id	Number	fnd_api.g_miss_num
detail_container_item_id	Number	fnd_api.g_miss_num
shipping_method_code	Varchar2(30)	fnd_api.g_miss_char

Attribute	Type	Default Value
carrier_id	Number	fnd_api.g_miss_num
freight_terms_code	Varchar2(30)	fnd_api.g_miss_char
freight_terms_name	Varchar2(30)	fnd_api.g_miss_char
freight_carrier_code	Varchar2(30)	fnd_api.g_miss_char
shipment_priority_code	Varchar2(30)	fnd_api.g_miss_char
fob_code	Varchar2(30)	fnd_api.g_miss_char
fob_name	Varchar2(30)	fnd_api.g_miss_char
dep_plan_required_flag	Varchar2(1)	fnd_api.g_miss_char
customer_prod_seq	Varchar2(50)	fnd_api.g_miss_char
customer_dock_code	Varchar2(30)	fnd_api.g_miss_char
gross_weight	Number	fnd_api.g_miss_num
net_weight	Number	fnd_api.g_miss_num
weight_uom_code	Varchar2(3)	fnd_api.g_miss_char
weight_uom_desc	Varchar2(50)	fnd_api.g_miss_char
volume	Number	fnd_api.g_miss_num
volume_uom_code	Varchar2(3)	fnd_api.g_miss_char
volume_uom_desc	Varchar2(50)	fnd_api.g_miss_char
top_model_line_id	Number	fnd_api.g_miss_num
ship_set_id	Number	fnd_api.g_miss_num
ato_line_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
arrival_set_id	Number	fnd_api.g_miss_num
ship_model_complete_flag	Varchar2(1)	fnd_api.g_miss_char
cust_po_number	Varchar2(50)	fnd_api.g_miss_char
released_status	Varchar2(1)	fnd_api.g_miss_char
packing_instructions	Varchar2(2000)	fnd_api.g_miss_char
shipping_instructions	Varchar2(2000)	fnd_api.g_miss_char
container_name	Varchar2(50)	fnd_api.g_miss_char
container_flag	Varchar2(1)	fnd_api.g_miss_char
delivery_detail_id	Number	fnd_api.g_miss_num

Record Parameter Attribute Validations

sold_to_org_id

Should be a valid unique element of ra_customers.customer_id.

customer_number

Should be a valid unique element of ra_customers.customer_number

ship_from_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_from_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

delier_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

deliver_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

intmed_ship_to_org_id

Should be a valid element of org_organization_definitions.organization_id.

intmed_ship_to_org_code

Should be a valid element of org_organization_definitions.organization_code.

ship_tolerance_above

Should be a non-negative number.

ship_tolerance_below

Should be a non-negative number.

ordered_quantity

Should be a non-negative whole number.

order_quantity_uom

Should be a valid element of mtl_items_uoms_view.uom_code.

ordered_quantity_unit_of_measure

Should be a valid element of mtl_items_uoms_view.unit_of_measure.

shipping_method_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type SHIP_METHOD.

freight_terms_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FREIGHT_TERMS.

freight_terms_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FREIGHT_TERMS.

fob_code

Should be a valid element of fnd_lookup_values_vl.lookup_code for lookup_type FOB.

fob_name

Should be a valid element of fnd_lookup_values_vl.meaning for lookup_type FOB.

dep_plan_required_flag

Should be Y or N.

gross_weight

Should be a non-negative number.

net_weight

Should be a non-negative number.

weight_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a weight uom class

weight_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a weight uom class.

volume

Should be a non-negative number.

volume_uom_code

Should be a valid element of mtl_units_of_measure.uom_code for a volume uom class.

volume_uom_desc

Should be a valid element of mtl_units_of_measure.unit_of_measure for a volume uom class.

ship_model_complete_flag

Should be Y or N.

released_status

Should be one of R, S, Y or X.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Auto_Pack API Features

The Auto_Pack API has the following features.

The Auto_Pack procedure takes in a table of ids of either delivery lines or container or deliveries and autopacks the lines/containers/deliveries into detail containers and returns a table of container instance ids created during the autopacking process

Functional Overview

This API takes in a table of ids of either delivery lines or container or deliveries and autopacks the lines/containers/deliveries into detail containers. The grouping id table is used only if the input table of entities are lines or containers only. The packing of lines and containers into parent containers is determined by the grouping id for each line/container.

If the grouping id table is not input, the API determines the grouping ids for the lines/containers based on the grouping attributes of the lines/containers. The lines/containers are then autopacked into detail containers and the detail containers are

packed into parent/master containers based on whether the p_pack_cont_flag is set to 'Y' or 'N'. The API returns a table of container instance ids created during the autopacking operation. If the detail containers are packed into parent containers, the output table of ids will contain both the detail and parent containers' delivery detail ids.

Procedure Parameter Descriptions

WSH_CONTAINER_PUB.AUTO_PACK

The following chart describes all parameters used by the public procedure WSH_CONTAINER_PUB.AUTO_PACK. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_CONTAINER_PUB.AUTO_PACK Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
p_validation_level	IN	Number	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_entity_tab	IN	Table	-
p_entity_type	IN	Varchar2	x
p_group_id_tab	IN	Table	-
p_pack_cont_flag	IN	Varchar2	x
x_cont_inst_tab	IN	Table	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_entity_tab

Table of ids of either lines or containers or deliveries that need to be autopacked of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

p_entity_type

Type of entity id contained in the entity_tab that needs to be autopacked ('L' - lines, 'C' - containers or 'D' - deliveries).

p_group_id_tab

Table of ids (numbers that determine the grouping of lines for packing into containers) of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

p_pack_cont_flag

A 'Y' or 'N' value to determine whether to autopack the detail containers that are created into parent containers.

x_cont_inst_tab

Table of container IDs created during the autopacking process of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Container_Actions API Features

The Container_Actions API has the following features.

The Container_Actions procedure enables you to carry out actions on a container. It accepts as IN parameters a table of delivery detail ids, an action code, and any additional parameters needed for specific actions, and returns a completion status.

Functional Overview

This API takes in a table of delivery detail ids and name and/or delivery detail id of the container to pack. If the action code is assigned then delivery id and delivery name must be specified. The API determines what action to perform based on the action code and then calls appropriate private pack/assign/unpack/unassign API. The input table of ids could be lines or containers. The delivery lines and containers are separated from the input table and validated before the appropriate private APIs are called.

Procedure Parameter Descriptions**WSH_CONTAINER_PUB.CONTAINER_ACTIONS**

The following chart describes all parameters used by the public procedure WSH_CONTAINER_PUB.CONTAINER_ACTIONS. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_CONTAINER_PUB.CONTAINER_ACTIONS Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x

Parameter	Usage	Type	Required
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
p_validation_level	IN	Number	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_detail_tab	IN	Table	x
p_container_instance_id	IN	Number	-
p_container_flag	IN	Table	-
p_delivery_flag	IN	Varchar2	-
p_delivery_id	IN	Number	-
p_delivery_name	IN	Varchar2	-
p_action_code	IN	Varchar2	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message

- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_detail_tab

Input table of delivery detail ids of type WSH_UTIL_CORE.ID_TAB_TYPE which is a table of type Number indexed by binary integers.

p_container_instance_id

Delivery detail ID of parent container being packed

p_container_name

Container name if ID is not known

p_container_flag

Y or N depending on whether to unpack or not respectively.

p_delivery_flag

Y or N depending on whether the container needs to be unassigned from a delivery or not respectively.

p_delivery_id

Delivery ID the container is assigned to.

p_delivery_name

Delivery name the container is assigned to.

p_action_code

Action code Pack, Assign, Unpack, or Unassign to specify which action to perform.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Freight Costs Public Application Program Interface

The Freight Costs Public Application Program Interface (API) is a public API that consists of the following three procedures in package WSH_FREIGHT_COSTS_PUB:

Create_Update_Freight_Costs: Enables you to create a new freight cost record and update an existing freight cost record.

Validate_Freight_Cost_Type: Validates that the freight cost type exists.

Delete_Freight_Costs: Enables you to delete a freight cost record.

This section describes how to use the Freight Costs Public API and how it functions in Oracle Shipping Execution.

Create_Update_Freight_Costs API Features

The Create_Update_Freight_Cost API has the following features.

The Create_Update_Freight_Costs procedure enables you to create a new freight cost record or update an existing freight cost record in WSH_FREIGHT_COSTS table. The FREIGHT_COST_ID, and return status of a new freight cost record are passed as OUT parameters, while a freight cost record of freight cost information is passed as an IN parameter.

Functional Overview

This API creates a new freight cost record in WSH_FREIGHT_COSTS as specified by IN parameter p_action_code value CREATE. It inserts the freight cost information into WSH_FREIGHT_COSTS and returns the FREIGHT_COST_ID of the new freight cost record. It also updates an existing freight cost record in WSH_FREIGHT_COSTS as specified by IN parameter p_action_code value UPDATE.

The API validates freight information such as freight cost type, unit amount, conversion rate, and currency before performing the actions of creating or updating a freight cost record. It also checks that the insert or update statements were successful, and if not returns an error.

Also, ensure that only one shipping entity—a trip, stop, delivery, delivery leg, or delivery detail—is passed in to be associated with a freight cost record. Each freight cost must be associated with only one shipping entity.

Procedure Parameter Descriptions

WSH_FREIGHT_COSTS_PUB.CREATE_UPDATE_FREIGHT_COSTS

The following chart describes all parameters used by the public procedure WSH_FREIGHT_COSTS_PUB.CREATE_UPDATE_FREIGHT_COSTS. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_FREIGHT_COSTS_PUB.CREATE_UPDATE_FREIGHT_COSTS Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_pub_freight_costs	IN OUT	Record	x
p_action_code	IN	Varchar2	x
x_freight_cost_id	OUT	Number	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message

- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_pub_freight_costs

Attributes of the freight cost entity of type PubFreightCostRecType. These attributes are inserted/updated in WSH_FREIGHT_COSTS. Definition of PubFreightCostRecType follows.

p_action_code

Specifies whether API should create a new freight cost record or update existing freight cost record information based on its values CREATE or UPDATE.

x_freight_cost_id

ID of new freight cost record being created.

Record Parameter Descriptions

PubFreightCostRecType RECORD DEFINITION

To encapsulate WSH_FREIGHT_COSTS table definition and Value column equivalents for ID columns in a PL/SQL record, define PubFreightCostRecType and use this to pass freight cost record information to the Create_Update_Freight_Cost routine.

PubFreightCostRecType RECORD DEFINITION

Attribute	Type	Default Value
freight_cost_id	Number	fnd_api.g_miss_num
freight_cost_type_id	Number	fnd_api.g_miss_num
unit_amount	Number	fnd_api.g_miss_num
currency_code	Varchar2(15)	fnd_api.g_miss_char
conversion_date	Date	fnd_api.g_miss_date
conversion_rate	Number	fnd_api.g_miss_num
conversion_type_code	Varchar2(30)	fnd_api.g_miss_char
trip_id	Number	fnd_api.g_miss_num
trip_name	Varchar2(30)	fnd_api.g_miss_char
stop_id	Number	fnd_api.g_miss_num
stop_location_id	Number	fnd_api.g_miss_num
planned_dep_date	Date	fnd_api.g_miss_date
delivery_id	Number	fnd_api.g_miss_num
delivery_name	Varchar2(30)	fnd_api.g_miss_char
delivery_leg_id	Number	fnd_api.g_miss_num
delivery_detail_id	Number	fnd_api.g_miss_num
attribute_category	Varchar2(150)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char
creation_date	Date	fnd_api.g_miss_date
created_by	Number	fnd_api.g_miss_num
last_update_date	Date	fnd_api.g_miss_date
last_updated_by	Number	fnd_api.g_miss_num
last_update_login	Date	fnd_api.g_miss_date
program_application_id	Number	fnd_api.g_miss_num
program_id	Number	fnd_api.g_miss_num

Attribute	Type	Default Value
program_update_date	Date	fnd_api.g_miss_date
request_id	Number	fnd_api.g_miss_num
freight_cost_type	Varchar2(30)	fnd_api.g_miss_char
action_code	Varchar2(30)	fnd_api.g_miss_char

Record Parameter Attribute Validations

freight_cost_id

Should be a unique valid element of wsh_freight_cost_types.freight_cost_id.

freight_cost_type_id

Should be a valid element of wsh_freight_cost_types.freight_cost_type_id.

unit_amount

Should be a non-negative number.

currency_code

Should be a valid element of fnd_currencies_vl.currency_code.

conversion_rate

Should be a non-negative number.

trip_id

Should be a valid unique element of wsh_trips.trip_id.

trip_name

Should be a valid unique element of wsh_trips.name

stop_id

Should be a valid unique element of wsh_trip_stops.stop_id.

stop_location_id

Should be a valid element of wsh_trip_stops.stop_location_id.

planned_dep_date

Should be a valid element of wsh_trip_stops.planned_departure_date.

delivery_id

Should be a valid unique element of wsh_new_deliveries.delivery_id.

delivery_name

Should be a valid unique element of wsh_new_deliveries.name

delivery_leg_id

Should be a valid element of wsh_delivery_legs.delivery_leg_id.

delivery_detail_id

Should be a valid element of wsh_delivery_details.delivery_detail_id.

freight_cost_type

Should be a valid element of wsh_freight_cost_types.name.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Validate_Freight_Cost_Type API Features

The Validate_Freight_Cost_Type API has the following features.

The Validate_Freight_Cost_Type procedure enables you to validate an existing freight cost type from the WSH_FREIGHT_COST_TYPES table. The freight cost type is passed as an IN parameter, while the freight cost id and a return status are passed as an OUT parameter.

Procedure Parameter Descriptions**WSH_FREIGHT_COSTS_PUB.DELETE_FREIGHT_COSTS**

The following chart describes all parameters used by the public procedure WSH_FREIGHT_COSTS_PUB.DELETE_FREIGHT_COSTS. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_FREIGHT_COSTS_PUB.DELETE_FREIGHT_COSTS Parameters

Parameter	Usage	Type	Required
p_freight_cost_type	IN	Varchar2	x
x_freight_cost_type_id	IN OUT	Varchar2	-
x_return_status	OUT	Varchar2	-

p_freight_cost_type

Freight cost type that needs to be validated.

x_freight_cost_type_id

Type Id of the freight cost being validated.

x_return_status

Requests that the API return the status of the data for you after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR.

Delete_Freight_Costs API Features

The Delete_Freight_Costs API has the following features.

The Delete_Freight_Costs procedure enables you to delete an existing freight cost record from the WSH_FREIGHT_COSTS table. A freight cost record of freight cost information of the freight cost record being deleted is passed as an IN parameter.

Functional Overview

The Delete_Freight_Costs procedure enables you to delete a new freight cost record from WSH_FREIGHT_COSTS table. It identifies the freight record being deleted by using the freight_cost_id attribute in the freight cost record that is being passed as an IN parameter.

Procedure Parameter Descriptions

WSH_FREIGHT_COSTS_PUB.DELETE_FREIGHT_COSTS

The following chart describes all parameters used by the public procedure WSH_FREIGHT_COSTS_PUB.DELETE_FREIGHT_COSTS. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_FREIGHT_COSTS_PUB.DELETE_FREIGHT_COSTS Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-

Parameter	Usage	Type	Required
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_pub_freight_costs	IN OUT	Record	x

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_pub_freight_costs

Attributes of the freight cost entity of type PubFreightCostRecType. These attributes are inserted/updated in WSH_FREIGHT_COSTS. Definition of PubFreightCostRecType follows.

Record Parameter Descriptions

PubFreightCostRecType RECORD DEFINITION

To encapsulate WSH_FREIGHT_COSTS table definition and Value column equivalents for ID columns in a PL/SQL record, we define PubFreightCostRecType and use this to pass Freight Cost information to the Delete_Freight_Cost routine.

PubFreightCostRecType RECORD DEFINITION

Attribute	Type	Default Value
freight_cost_id	Number	fnd_api.g_miss_num
freight_cost_type_id	Number	fnd_api.g_miss_num
unit_amount	Number	fnd_api.g_miss_num
currency_code	Varchar2(15)	fnd_api.g_miss_char
conversion_date	Date	fnd_api.g_miss_date
conversion_rate	Number	fnd_api.g_miss_num
conversion_type_code	Varchar2(30)	fnd_api.g_miss_char
trip_id	Number	fnd_api.g_miss_num
trip_name	Varchar2(30)	fnd_api.g_miss_char
stop_id	Number	fnd_api.g_miss_num
stop_location_id	Number	fnd_api.g_miss_num
planned_dep_date	Date	fnd_api.g_miss_date

Attribute	Type	Default Value
delivery_id	Number	fnd_api.g_miss_num
delivery_name	Varchar2(30)	fnd_api.g_miss_char
delivery_leg_id	Number	fnd_api.g_miss_num
delivery_detail_id	Number	fnd_api.g_miss_num
attribute_category	Varchar2(150)	fnd_api.g_miss_char
attribute1	Varchar2(150)	fnd_api.g_miss_char
attribute2	Varchar2(150)	fnd_api.g_miss_char
attribute3	Varchar2(150)	fnd_api.g_miss_char
attribute4	Varchar2(150)	fnd_api.g_miss_char
attribute5	Varchar2(150)	fnd_api.g_miss_char
attribute6	Varchar2(150)	fnd_api.g_miss_char
attribute7	Varchar2(150)	fnd_api.g_miss_char
attribute8	Varchar2(150)	fnd_api.g_miss_char
attribute9	Varchar2(150)	fnd_api.g_miss_char
attribute10	Varchar2(150)	fnd_api.g_miss_char
attribute11	Varchar2(150)	fnd_api.g_miss_char
attribute12	Varchar2(150)	fnd_api.g_miss_char
attribute13	Varchar2(150)	fnd_api.g_miss_char
attribute14	Varchar2(150)	fnd_api.g_miss_char
attribute15	Varchar2(150)	fnd_api.g_miss_char

Attribute	Type	Default Value
creation_date	Date	fnd_api.g_miss_date
created_by	Number	fnd_api.g_miss_num
last_update_date	Date	fnd_api.g_miss_date
last_updated_by	Number	fnd_api.g_miss_num
last_update_login	Date	fnd_api.g_miss_date
program_application_id	Number	fnd_api.g_miss_num
program_id	Number	fnd_api.g_miss_num
program_update_date	Date	fnd_api.g_miss_date
request_id	Number	fnd_api.g_miss_num
freight_cost_type	Varchar2(30)	fnd_api.g_miss_char
action_code	Varchar2(30)	fnd_api.g_miss_char

Record Parameter Attribute Validations

freight_cost_id

Should be a valid element of wsh_freight_cost_types.freight_cost_id.

freight_cost_type_id

Should be a valid element of wsh_freight_cost_types.freight_cost_type_id.

unit_amount

Should be a non-negative number.

currency_code

Should be a valid element of fnd_currencies_vl.currency_code.

conversion_rate

Should be a non-negative number.

trip_id

Should be a valid unique element of wsh_trips.trip_id.

trip_name

Should be a valid unique element of wsh_trips.name

stop_id

Should be a valid unique element of wsh_trip_stops.stop_id.

stop_location_id

Should be a valid element of wsh_trip_stops.stop_location_id.

planned_dep_date

Should be a valid element of wsh_trip_stops.planned_departure_date.

delivery_id

Should be a valid unique element of wsh_new_deliveries.delivery_id.

delivery_name

Should be a valid unique element of wsh_new_deliveries.name

delivery_leg_id

Should be a valid element of wsh_delivery_legs.delivery_leg_id.

delivery_detail_id

Should be a valid element of wsh_delivery_details.delivery_detail_id.

freight_cost_type

Should be a valid element of wsh_freight_cost_types.name.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Pick Release Application Program Interface

The Pick Release public consists of the following procedures in package WSH_PICKING_BATCHES_PUB:

Create_Batch: Enables you to create a new pick release batch.

Release_Batch: Enables you to release a batch.

Get_Batch_Record: Takes in a batch_id or batch_name and retrieves the batch record from wsh_picking_batches.

This section describes how to use the Pick Release Public API and how it functions in Oracle Shipping Execution.

Note: When pick releasing using the

WSH_PICKING_BATCHES_PUB.CREATE_BATCH API, the default value for the attribute `append_flag` is N. To enable Append Deliveries during pick release from the API, the value of `append_flag` must be set to Y or NULL. If `append_flag` is NULL, and the `batch_id` is specified, then WSH_PICKING_BATCHES_PUB.CREATE_BATCH will get the value of `append_flag` from the picking batch.

If `append_flag` is NULL or Y, then all of the following conditions must be met:

- The `organization_id` is specified and `append_limit` of the organization in the Shipping Parameters window is not Do Not Append.
- The `autocreate_delivery_flag` is Y and `ac_delivery_criteria` is N.
- The `auto_pick_confirm_flag` is N
- The `ship_confirm_rule_id` is NULL.
- The `autopack_flag` is N.

Also, if you run this API with `p_release_mode` ONLINE, when calling WSH_PICKING_BATCHES_PUB.Release_Batch, it is not recommended that you run pick release on the same set of delivery lines with `append_flag` set to Y because may result in unpredictable results. The existing deliveries selected for appending may differ.

Create_Batch API Features

The Create_Batch API enables you to create a new pick release batch.

Functional Overview

This API creates a new pick release batch with a unique batch ID that is then used by the Release_Batch API to release the batch.

Procedure Parameter Descriptions

WSH_PICKING_BATCHES_PUB.CREATE_BATCH

The following chart describes all parameters used by the public procedure WSH_PICKING_BATCHES_PUB.CREATE_BATCH. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_PICKING_BATCHES_PUB.CREATE_BATCH Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	Out	Varchar2	-
p_rule_id	IN	Number	-
p_rule_name	IN	Varchar2	-
p_batch_rec	IN	Record	x
p_batch_prefix	IN	Varchar2	-
x_batch_id	OUT	Number	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

p_commit

Requests that the API update information for you after it completes its function. Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_rule_id

Indicates the picking rule identification.

p_rule_name

Indicates the picking rule name.

p_batch_rec

Contains all picking batch parameters.

p_batch_prefix

Used to add a prefix to the batch id.

x_batch_id

Returns the batch id that is created.

Record Parameter Descriptions

Batch_Info_Rec Type RECORD DEFINITION

To encapsulate WSH_PICKING_BATCHES table definition and Value column equivalents for ID columns in a PL/SQL record, we define Batch_Info_Rec Type and use this to pass Pick Release information to the Create_Batch routine.

Batch_Info_Rec Type RECORD DEFINITION

Attribute	Type
Allocation_Method	Varchar2(1)
Cross_Dock_Criteria_ID	Number
Backorders_Only_Flag	Varchar2(1)
Document_Set_Id	Number
Document_Set_Name	Varchar2(30)
Existing_Rsvs_Only_Flag	Varchar2(1)
Shipment_Priority_Code	Varchar2(30)
Ship_Method_Code	Varchar2(30)
Ship_Method_Name	Varchar2(80)
Customer_Id	Number
Customer_Number	Varchar2(30)
Order_Header_Id	Number
Order_Number	Number
Ship_Set_Id	Number
Ship_Set_Number	Varchar2(30)
Inventory_Item_Id	Number
Order_Type_Id	Number
From_Requested_Date	Date
To_Requested_Date	Date

Attribute	Type
From_Scheduled_Ship_Date	Date
To_Scheduled_Ship_Date	Date
Ship_To_Location_Id	Number
Ship_To_Location_code	Varchar2(30)
Ship_From_Location_Id	Number
Ship_From_Location_code	Varchar2(30)
Trip_Name	Varchar2(30)
Delivery_Id	Number
Delivery_Name	Varchar2(30)
Include_Planned_Lines	Varchar2(1)
Pick_Grouping_Rule_Id	Number
Pick_Grouping_Rule_Name	Varchar2(30)
Pick_Sequence_Rule_Id	Number
Pick_Sequence_Rule_Name	Varchar2(30)
Autocreate_Delivery_Flag	Varchar2(1)
Attribute_Category	Varchar2(30)
attribute1	Varchar2(150)
attribute2	Varchar2(150)
attribute3	Varchar2(150)
attribute4	Varchar2(150)

Attribute	Type
attribute5	Varchar2(150)
attribute6	Varchar2(150)
attribute7	Varchar2(150)
attribute8	Varchar2(150)
attribute9	Varchar2(150)
attribute10	Varchar2(150)
attribute11	Varchar2(150)
attribute12	Varchar2(150)
attribute13	Varchar2(150)
attribute14	Varchar2(150)
attribute15	Varchar2(150)
Autodetail_Pr_Flag	Varchar2(1)
Trip_Stop_Id	Number
Trip_Stop_location_id	Number
Default_Stage_Subinventory	Varchar2(10)
Default_Stage_Locator_Id	Number
Pick_From_Subinventory	Varchar2(10)
Pick_From_locator_Id	Number
Auto_Pick_Confirm_Flag	Varchar2(1)
Delivery_Detail_Id	Number

Attribute	Type
Project_Id	Number
Task_Id	Number
Organization_Id	Number
Organization_Code	Varchar2(3)
Ship_Confirm_Rule_Id	Number
Ship_Confirm_Rule_Name	Varchar2(30)
Autopack_Flag	Varchar2(1)
Autopack_Level	Number
Task_Planning_Flag	Varchar2(1)
Category_Set_ID	Number
Category_ID	Number
Ship_Set_Smc_Flag	Varchar2(1)
region_ID	Number
zone_ID	Number
ac_Delivery_Criteria	Varchar2(2)
rel_subinventory	Varchar2(10)
append_flag	Varchar2(1)
task_priority	Number

Record Parameter Attribute Validations

allocation_method

Should be a method to allocate items. Valid entries are X: Cross Dock, I: Inventory, C:

Prioritize Cross Dock, and N: Prioritize Inventory

cross_dock_criteria_ID

Should be a valid element of wms_crossdock_criteria_b.criteria_id. Translation table: wms_crossdock_criteria_tl, and it can be used to search for the criterion_id based on the criterion_name.

backorders_only_flag

Should be a valid element of wsh_picking_batches.Backorders_Only_Flag. Valid entries are I, E, and O.

document_set_id

Should be a valid element of wsh_picking_batches.Document_Set_Id.

document_set_name

Should be a valid element of wsh_report_sets.name.

existing_rsvs_only_flag

Should be a valid element of wsh_picking_batches.Existing_Rsvs_Only_Flag.

shipment_priority_code

Should be a valid element of oe_lookups.lookup_code.

ship_method_code

Should be a valid unique element of wsh_picking_batches.Ship_Method_Code.

ship_method_name

Should be a valid unique element of fnd_lookup_values_vl.meaning.

customer_id

Should be a valid unique element of wsh_picking_batches.Customer_Id.

customer_number

Should be a valid element of ra_customers.customer_number.

order_header_id

Should be a valid element of wsh_picking_batches.Order_Header_Id.

order_number

Should be a valid unique element of oe_order_headers_all.Order_Number.

ship_set_id

Should be a valid unique element of wsh_picking_batches.Ship_Set_Number

ship_set_number

Should be a valid element of oe_sets.set_name.

inventory_item_id

Should be a valid element of wsh_picking_batches.Inventory_Item_Id.

order_type_id

Should be a valid element of wsh_picking_batches.Order_Type_Id.

order_type_name

Should be a valid element of oe_transaction_types_tl.name.

from_requested_date

Starting request date to release.

to_requested_date

Ending request date to release.

from_scheduled_ship_date

Starting schedule date to release.

to_scheduled_ship_date

Ending schedule date to release.

ship_to_location_id

Should be a valid element of wsh_picking_batches.Ship_To_Location_Id.

ship_to_location_code

Should be a valid element of hr_locations_all_tl.location_code.

ship_from_location_id

Should be a valid unique element of wsh_picking_batches.Ship_From_Location_Id.

ship_from_location_code

Should be a valid unique element of hr_locations_all_tl.location_code.

trip_id

Should be a valid element of wsh_picking_batches.Trip_Id.

trip_name

Should be a valid element of wsh_trips.name.

delivery_id

Should be a valid element of wsh_picking_batches.Delivery_Id.

delivery_name

Should be a valid element of wsh_new_deliveries.name.

include_planned_lines

Should be a valid unique element of wsh_picking_batches.Include_Planned_Lines.

pick_grouping_rule_id

Should be a valid unique element of wsh_picking_batches.Pick_Grouping_Rule_Id.

pick_grouping_rule_name

Should be a valid element of wsh_pick_grouping_rules.name.

pick_sequence_rule_id

Should be a valid element of wsh_picking_batches.Pick_Sequence_Rule_Id.

pick_sequence_rule_name

Should be a valid element of wsh_pick_sequence_rules.name.

auto_create_delivery_flag

Should be a valid element of wsh_picking_batches.Autocreate_Delivery_Flag.

attribute_category

Should be a valid unique element of wsh_picking_batches.Attribute_Category.

autodetail_pr_flag

Should be a valid unique element of wsh_picking_batches.Autodetail_Pr_Flag.

trip_stop_id

Should be a valid element of wsh_picking_batches.Trip_Stop_Id.

trip_stop_location_id

Should be a valid element of wsh_trip_stops.Stop_Id.

default_stage_subinventory

Should be a valid element of wsh_picking_batches.Default_Stage_Subinventory.

default_stage_locator_id

Should be a valid element of wsh_picking_batches.Default_Stage_Locator_Id.

pick_from_subinventory

Should be a valid unique element of wsh_picking_batches.Pick_From_Subinventory.

pick_from_locator_id

Should be a valid unique element of wsh_picking_batches.Pick_From_locator_Id.

auto_pick_confirm_flag

Should be a valid element of wsh_picking_batches.Auto_Pick_Confirm_Flag. Valid entries are Y or N.

delivery_detail_id

Should be a valid element of wsh_picking_batches.Delivery_Detail_Id.

project_id

Should be a valid element of wsh_picking_batches.Project_Id.

task_id

Should be a valid element of wsh_picking_batches.Task_Id.

organization_id

Should be a valid unique element of wsh_picking_batches.Organization_Id.

organization_code

Should be a valid unique element of org_organization_definitions.organization_code.

ship_confirm_rule_id

Should be a valid element of wsh_picking_batches.Ship_Confirm_Rule_Id.

ship_confirm_rule_name

Should be a valid element of wsh_Ship_Confirm_rules.name.

autopack_flag

Should be a valid element of wsh_picking_batches.Autopack_Flag.

autopack_level

Should be a valid element of wsh_picking_batches.Autopack_Level. Valid entries are 0, 1, or 2.

task_planning_flag

Should be a valid unique element of wsh_picking_batches.Task_Planning_Flag.

category_set_id

Should be a valid unique element of wsh_picking_batches.Category_Set_ID.

category_id

Should be a valid element of wsh_picking_batches.Category_ID.

ship_set_smc_flag

Should be a valid element of wsh_picking_batches.Ship_Set_Smc_Flag.

region_id

Should be a valid element of wsh_picking_batches.region_id.

zone_id

Should be a valid element of wsh_picking_batches.zone_id.

ac_delivery_criteria

Should be a valid unique element of wsh_picking_batches.ac_delivery_criteria.

rel_subinventory

Should be a valid unique element of wsh_picking_batches.rel_subinventory.

append_flag

Should be a valid element of wsh_picking_batches.append_flag.

task_priority

Should be a valid element of wsh_picking_batches.task_priority.

Error Handling

Refer to parameters p_init_msg_list, x_msg_count, and x_msg_data on retrieving error messages.

Release_Batch API Features

The Release_Batch API enables you to release a newly created pick release batch.

Functional Overview

This API enables you to release a pick release batch that was generated by the Create_Batch API.

Procedure Parameter Descriptions

WSH_PICKING_BATCHES_PUB.RELEASE_BATCH

The following chart describes all parameters used by the public procedure WSH_PICKING_BATCHES_PUB.RELEASE_BATCH. All of the inbound and outbound parameters are listed. Additional information on these parameters follows.

WSH_PICKING_BATCHES_PUB.RELEASE_BATCH Parameters

Parameter	Usage	Type	Required
p_api_version_number	IN	Number	x
p_init_msg_list	IN	Varchar2	-
p_commit	IN	Varchar2	-
x_return_status	OUT	Varchar2	-
x_msg_count	OUT	Number	-
x_msg_data	OUT	Varchar2	-
p_batch_id	IN	Number	-

Parameter	Usage	Type	Required
p_batch_name	IN	Varchar2	-
p_log_level	IN	Number	-
p_release_mode	IN	Varchar2	-
x_request_id	OUT	Number	-

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

p_commit

Requests that the API update information for you after it completes its function. Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_batch_id

Indicates the picking batch identification which is used to get the batch information from wsh_picking_batches.

p_batch_name

Indicates the picking batch name which is used to get the batch information from wsh_picking_batches.

p_log_level

Controls the log message generated by the concurrent pick release process.

p_release_mode

Controls the release mode of either Concurrent or Online. Default = Concurrent.

x_request_id

Returns the request id for the concurrent pick release request.

Get_Batch_Record API Features

The WSH_PICKING_BATCHES_PUB.GET_BATCH_RECORD API consists of the following entities: API retrieves data from wsh_picking_batches table for the given batch_id or batch_name.

Functional Overview

The API takes in a Batch_id or Batch_name and retrieves the batch record from wsh_picking_batches. It considers batch_name value only when the parameter batch_id is not being passed to the API. The API validates p_batch_id and p_batch_name values. For the valid values, information is retrieved and passed back to the calling API through the OUT parameter x_batch_rec. API passes NULL values for all read only fields, such as customer name, item name, document set name, as per release sales order form.

Procedure Parameter Descriptions

The following table lists all parameters used by the public API WSH_PICKING_BATCHES_PUB.GET_BATCH_RECORD. All of the inbound and outbound parameters are listed. Additional information on these parameters follows the table.

Parameter Name	Usage	Type	Required	Derived
p_api_version	In	NUMBER	x	
p_init_msg_list	In	VARCHAR2	-	
p_commit	In	VARCHAR2	-	
x_return_status	Out	VARCHAR2	-	
x_msg_count	Out	NUMBER	-	
x_msg_data	Out	VARCHAR2	-	
p_batch_id	IN	NUMBER	-	
p_batch_name	IN	NUMBER	-	
x_batch_rec	OUT	WSH_PICKING_BATCHES_PUB.Batch_Info_Rec	-	

Important: Either of Batch Id or Batch name is required.

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and

Number respectively.

Default Value: FND_API.G_FALSE

p_commit

Requests that the API update information for you after it completes its function. Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_batch_id

Picking batch id value corresponds to batch_id column of WSH_PICKING_BATCHES table.

p_batch_name

Picking batch name value corresponds to name column of WSH_PICKING_BATCHES table.

Record Parameter Descriptions

Batch_Info_Rec Type RECORD DEFINITION

To encapsulate WSH_PICKING_BATCHES table definition and Value column equivalents for ID columns in a PL/SQL record. API retrieves all database column values present in WSH_PICKING_BATCHES table. It returns NULL for all read only (NAME/CODE) fields present in the record group (Example. Document Set Name, Customer Name etc).

Attribute	Type
Backorders_Only_Flag	Varchar2(1)

Attribute	Type
Document_Set_Id	Number
Document_Set_Name	Varchar2(30)
Existing_Rsvs_Only_Flag	Varchar2(1)
Shipment_Priority_Code	Varchar2(30)
Ship_Method_Code	Varchar2(30)
Ship_Method_Name	Varchar2(80)
Customer_Id	Number
Customer_Number	Varchar2(30)
Order_Header_Id	Number
Order_Number	Number
Ship_Set_Id	Number
Ship_Set_Number	Varchar2(30)
Inventory_Item_Id	Number
Order_Type_Id	Number
From_Requested_Date	Date
To_Requested_Date	Date
From_Scheduled_Ship_Date	Date
To_Scheduled_Ship_Date	Date
Ship_To_Location_Id	Number
Ship_To_Location_code	Varchar2(30)

Attribute	Type
Ship_From_Location_Id	Number
Ship_From_Location_code	Varchar2(30)
Trip_Name	Varchar2(30)
Delivery_Id	Number
Delivery_Name	Varchar2(30)
Include_Planned_Lines	Varchar2(1)
Pick_Grouping_Rule_Id	Number
Pick_Grouping_Rule_Name	Varchar2(30)
Pick_Sequence_Rule_Id	Number
Pick_Sequence_Rule_Name	Varchar2(30)
Autocreate_Delivery_Flag	Varchar2(1)
Attribute_Category	Varchar2(30)
attribute1	Varchar2(150)
attribute2	Varchar2(150)
attribute3	Varchar2(150)
attribute4	Varchar2(150)
attribute5	Varchar2(150)
attribute6	Varchar2(150)

Attribute	Type
attribute7	Varchar2(150)
attribute8	Varchar2(150)
attribute9	Varchar2(150)
attribute10	Varchar2(150)
attribute11	Varchar2(150)
attribute12	Varchar2(150)
attribute13	Varchar2(150)
attribute14	Varchar2(150)
attribute15	Varchar2(150)
Autodetail_Pr_Flag	Varchar2(1)
Trip_Stop_Id	Number
Trip_Stop_location_id	Number
Default_Stage_Subinventory	Varchar2(10)
Default_Stage_Locator_Id	Number
Pick_From_Subinventory	Varchar2(10)
Pick_From_locator_Id	Number

Attribute	Type
Auto_Pick_Confirm_Flag	Varchar2(1)
Delivery_Detail_Id	Number
Project_Id	Number
Task_Id	Number
Organization_Id	Number
Organization_Code	Varchar2(3)
Ship_Confirm_Rule_Id	Number
Ship_Confirm_Rule_Name	Varchar2(30)
Autopack_Flag	Varchar2(1)
Autopack_Level	Number
Task_Planning_Flag	Varchar2(1)
Category_Set_ID	Number
Category_ID	Number
Ship_Set_Smc_Flag	Varchar2(1)
region_ID	Number
zone_ID	Number

Attribute	Type
ac_Delivery_Criteria	Varchar2(2)
rel_subinventory	Varchar2(10)
append_flag	Varchar2(1)
task_priority	Number
actual_departure_date	Date
allocation_method	Varchar2(1)
crossdock_criteria_id	Number
crossdock_criteria_name	Varchar2(80)

Record Parameter Attribute Validations

backorders_only_flag

Corresponds to wsh_picking_batches.Backorders_Only_Flag.

document_set_id

Corresponds to wsh_picking_batches.Document_Set_Id.

document_set_name

Returns NULL value.

existing_rsvs_only_flag

Corresponds to wsh_picking_batches.Existing_Rsvs_Only_Flag.

shipment_priority_code

Corresponds to wsh_picking_batches.Shipment_Priority_Code.

ship_method_code

Corresponds to wsh_picking_batches.Ship_Method_Code.

ship_method_name

Returns NULL value.

customer_id

Corresponds to wsh_picking_batches.Customer_Id.

customer_number

Returns NULL value.

order_header_id

Corresponds to wsh_picking_batches.Order_Header_Id.

order_number

Returns NULL value.

ship_set_id

Returns NULL value.

ship_set_number

Corresponds to wsh_picking_batches.ship_set_number.

inventory_item_id

Corresponds to wsh_picking_batches.Inventory_Item_Id.

order_type_id

Corresponds to wsh_picking_batches.Order_Type_Id

order_type_name

Returns null value.

from_requested_date

Corresponds to wsh_picking_batches.from_requested_date.

to_requested_date

Corresponds to wsh_picking_batches.to_requested_date.

from_scheduled_ship_date

Corresponds to wsh_picking_batches.from_scheduled_ship_date.

to_scheduled_ship_date

Corresponds to wsh_picking_batches.to_scheduled_ship_date.

ship_to_location_id

Corresponds to wsh_picking_batches.Ship_To_Location_Id.

ship_to_location_code

Returns null value.

ship_from_location_id

Corresponds to wsh_picking_batches.Ship_From_Location_Id.

ship_from_location_code

Returns null value.

trip_id

Corresponds to wsh_picking_batches.Trip_Id.

trip_name

Returns null value.

delivery_id

Corresponds to wsh_picking_batches.Delivery_Id.

delivery_name

Returns null value.

include_planned_lines

Corresponds to wsh_picking_batches.Include_Planned_Lines.

pick_grouping_rule_id

Corresponds to wsh_picking_batches.Pick_Grouping_Rule_Id.

pick_grouping_rule_name

Returns null value.

pick_sequence_rule_id

Corresponds to wsh_picking_batches.Pick_Sequence_Rule_Id.

pick_sequence_rule_name

Returns null value.

auto_create_delivery_flag

Corresponds to wsh_picking_batches.Autocreate_Delivery_Flag.

attribute_category

Corresponds to wsh_picking_batches.Attribute_Category.

autodetail_pr_flag

Corresponds to wsh_picking_batches.Autodetail_Pr_Flag.

trip_stop_id

Corresponds to wsh_picking_batches.Trip_Stop_Id.

trip_stop_location_id

Should be a valid element of wsh_trip_stops.Stop_Id.

default_stage_subinventory

Corresponds to wsh_picking_batches.Default_Stage_Subinventory.

default_stage_locator_id

Corresponds to wsh_picking_batches.Default_Stage_Locator_Id.

pick_from_subinventory

Corresponds to wsh_picking_batches.Pick_From_Subinventory.

pick_from_locator_id

Corresponds to wsh_picking_batches.Pick_From_locator_Id.

auto_pick_confirm_flag

Corresponds to wsh_picking_batches.Auto_Pick_Confirm_Flag.

delivery_detail_id

Corresponds to wsh_picking_batches.Delivery_Detail_Id.

project_id

Should be a valid element of wsh_picking_batches.Project_Id.

task_id

Corresponds to wsh_picking_batches.Task_Id.

organization_id

Corresponds to wsh_picking_batches.Organization_Id.

organization_code

Returns null value.

ship_confirm_rule_id

Corresponds to wsh_picking_batches.Ship_Confirm_Rule_Id.

ship_confirm_rule_name

Returns null value.

autopack_flag

Corresponds to wsh_picking_batches.Autopack_Flag.

autopack_level

Corresponds to wsh_picking_batches.Autopack_Level.

task_planning_flag

Corresponds to wsh_picking_batches.Task_Planning_Flag.

category_set_id

Corresponds to wsh_picking_batches.Category_Set_ID.

category_id

Corresponds to wsh_picking_batches.Category_ID.

ship_set_smc_flag

Corresponds to wsh_picking_batches.Ship_Set_Smc_Flag.

region_id

Corresponds to wsh_picking_batches.region_id.

zone_id

Corresponds to wsh_picking_batches.zone_id.

ac_delivery_criteria

Corresponds to wsh_picking_batches.ac_delivery_criteria.

rel_subinventory

Corresponds to wsh_picking_batches.rel_subinventory.

append_flag

Corresponds to wsh_picking_batches.append_flag.

task_priority

Corresponds to wsh_picking_batches.task_priority.

actual_departure_date

Corresponds to wsh_picking_batches.actual_departure_date.

allocation_method

Corresponds to wsh_picking_batches.allocation_method.

crossdock_criteria_id

Corresponds to wsh_picking_batches.crossdock_criteria_id.

crossdock_criteria_name

Returns NULL value.

Standard Validation

Oracle Shipping Execution validates all required columns in the WSH_PICKING_BATCHES_PUB.GET_BATCH_RECORD API. For specific information on the data referenced by these columns, see *Oracle Electronic Technical Reference Manual (eTRM)* .

It validates batch_id and batch_name values passed to the API. Either batch_id or Batch_name or both can be passed to API. If both are passed without any value, then API raises the error. API validates p_batch_id and p_batch_name parameter values against wsh_picking_batches table. If no record is found then API returns error status.

Error Handling

If any validation fails, the API will return the error status to the calling module. The WSH_PICKING_BATCHES_PUB.GET_BATCH_RECORD API processes the rows and reports the following values for every record:

Condition	Message Return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

Document Public Application Program Interface

The WSH_DOCUMENT_PUB.GET_DOCUMENT API consists of the following entities:

Retrives the information from wsh_document_instances table for the given document type and for the given sales order line id. Document type can be 'PACK_TYPE' or 'BOL'. The API does not return any value either **JB- open issue-Lynn** if document corresponds to the sales order line passed is CLOSED or parameter p_document_type contains any value other than 'PACK_TYPE' /'BOL'.

Functional Overview

Returns as a record containing all attributes of the currently open document of the specified type, for example Packing slip or Bill of lading, for a sales order line. There should be only one open document of a specific type for a delivery. If more than one open doc is available, then the API returns the first available one.

Procedure Parameter Descriptions

The following table lists all parameters used by the public API GET_DOCUMENT. All of the inbound and outbound parameters are listed. Additional information on these parameters follows the table.

Parameter Name	Usage	Type	Required	Derived
p_api_version	In	NUMBER	x	
p_init_msg_list	In	VARCHAR2	-	

Parameter Name	Usage	Type	Required	Derived
p_commit	In	VARCHAR2	-	
p_validation_level	In	VARCHAR2	-	
x_return_status	Out	VARCHAR2	-	
x_msg_count	Out	NUMBER	-	
x_msg_data	Out	VARCHAR2	-	
p_order_line_id	IN	NUMBER	x	
p_document_type	IN	VARCHAR2	x	
x_document_rec	OUT	WSH_Document_ PUB. document_rectype	-	

p_api_version_number

Compares the incoming API call's version number with the current version number. An error is returned if the version numbers are incompatible.

p_init_msg_list

Requests that the API initialize the message list on your behalf. If the x_msg_count is greater than 1, then the list of messages must be retrieved using the call FND_MSG_PUB.GET. The values are:

- p_msg_index => I
- p_encoded => F
- p_data => 1_message
- p_msg_index_out => 1_msg_index_out

where 1_message and 1_msg_index_out are local variables of types Varchar2(2000) and Number respectively.

Default Value: FND_API.G_FALSE

p_commit

Requests that the API update information for you after it completes its function.

Default Value: FND_API.G_FALSE

x_return_status

Requests that the API return the status of the data after it completes its function. Valid values include:

- Success: FND_API.G_RET_STS_SUCCESS
- Error: FND_API.G_RET_STS_ERROR
- Unexpected Error: FND_API.G_RET_STS_UNEXP_ERROR

x_msg_count

Indicates number of error messages API has encountered.

x_msg_data

Returns error message text. If the x_msg_count is equal to 1, then this contains the actual message.

p_order_line_id

Valid sales order line id which is associated to one or more delivery detail lines.

p_document_type

Document type represents the type of the document for which information needs to be retrieved. Possible values for the this parameter are 'PACK_TYPE' (for Packing slip report) and 'BOL' (for Bill of lading report).

Record Parameter Descriptions

document_rectype Type RECORD DEFINITION

To encapsulate WSH_DOCUMENT_INSTANCES table definition and Value column equivalents for ID columns in a PL/SQL record. API retrieves all database column values present in WSH_DOCUMENT_INSTANCES table. It returns NULL for all read only (NAME/CODE) fields present in the record group (Example. Document Set Name, Customer Name etc).

Attribute	Type
document_instance_id	Number

Attribute	Type
document_type	Varchar2(30)
entity_name	Varchar2(50)
entity_id	Number
doc_sequence_category_id	Number
sequence_number	Varchar2(50)
status	Varchar2(30)
final_print_date	Date
created_by	Number
creation_date	Date
last_updated_by	Number
last_update_date	Date
last_update_login	Number
program_application_id	Number
program_id	Number
program_update_date	Date
request_id	Number
attribute_category	Varchar2(150)
attribute1	Varchar2(150)
Attribute2	Varchar2(150)
Attribute3	Varchar2(150)

Attribute	Type
Attribute4	Varchar2(150)
Attribute5	Varchar2(150)
Attribute6	Varchar2(150)
Attribute7	Varchar2(150)
Attribute8	Varchar2(150)
Attribute9	Varchar2(150)
attribute10	Varchar2(150)
attribute11	Varchar2(150)
attribute12	Varchar2(150)
attribute13	Varchar2(150)
attribute14	Varchar2(150)
attribute15	Varchar2(150)

Record Parameter Attribute Descriptions

document_instance_id

Corresponds to wsh_document_instances.document_instance_id.

document_type

Corresponds to wsh_document_instances.document_type.

entity_name

Corresponds to wsh_document_instances.entity_name.

entity_id

Corresponds to wsh_document_instances.entity_id.

doc_sequence_category_id

Corresponds to wsh_document_instances.doc_sequence_category_id.

sequence_number

Corresponds to wssh_document_instances.

sequence_number. status

Corresponds to wsh_document_instances.status.

final_print_date

Corresponds to wsh_document_instances.

final_print_date created_by

Corresponds to wsh_document_instances.

created_by creation_date

Corresponds to wsh_document_instances.creation_date

last_updated_by

Corresponds to wsh_document_instances.last_updated_by

last_update_date

Corresponds to wsh_document_instances.last_update_date

last_update_login

Corresponds to wsh_document_instances.last_update_login

program_application_id

Corresponds to wsh_document_instances.program_application_id

program_id

Corresponds to wsh_document_instances.program_id

program_update_date

Corresponds to wsh_document_instances.program_update_date

request_id

Corresponds to wsh_document_instances.request_id

attribute_category

Corresponds to wsh_document_instances.attribute_category

attribute1 to attribute15

Corresponds to wsh_document_instances.attribute1 to
wsh_document_instances.attribute15.

Standard Validation

Oracle Shipping Execution validates all required columns in the WSH_DOCUMENT_PUB.GET_DOCUMENT API. For specific information on the data referenced by these columns, see *Oracle Electronic Technical Reference Manual (eTRM)*.

API checks the p_document_type value passed to the API. If parameter

p_document_type contains any value other than 'PACK_TYPE' /'BOL' then API do not return any document instance information for the sales order line id passed to the API.

The API returns the following return statuses to the calling module:

Condition	Message return	Description
Success	S	Process succeeded.
Failure	E	Expected error.
Failure	U	Unexpected error.

Custom Public Application Program Interface

The Custom Application Program Interface (API) is a public API that consists of the following two procedures in the package WSH_CUSTOM_PUB:

- **UI_Location_Code** : Enables the user to tailor how the location (ui_location_code) information is displayed in the Shipping forms.
- **Calculate_TP_Dates** : Enables the user to select the delivery window dates based on different parameters from order management. Customers can customize their calculation of the delivery window dates (Earliest/Latest Ship Dates and Earliest/Latest Delivery Dates). These will then be used for population at the delivery detail level and will get propagated to the container or delivery levels at action points such as assign/pack.

UI_Location_Code API Features

The UI_Location_Code API enables you to tailor how the location information is displayed in the Shipping forms and reports based on the attributes passed to API. The API is called from the package, WSH_LOCATIONS_PKG. The tables involved are WSH_LOCATIONS, HZ_LOCATIONS ,HZ_PARTY_SITES, HZ_PARTIES, HR_LOCATIONS_ALL ,HR_LOCATIONS_ALL_TL and HR_ORGANIZATION_UNITS. The external location information is retrieved from the HZ tables and the internal locations is retrieved from the HR tables before inserting or updating the record in WSH_LOCATIONS table.

Functional Overview

This API is designed for you to tailor the display of the location (ui_location_code) information displayed in the Shipping forms and reports. The API is called when a new location is created or an existing location is updated. To use the API for the location, the

value of variable x_use_custom_ui_location in WSH_CUSTOM_PUB.ui_location_code must be set to 'Y'.

Procedure Parameter Descriptions

The following table lists all of the parameters used by the public WSH_CUSTOM_PUB.UI_Location_Code. Additional information on these parameters follows in the table.

Parameter Name	Usage	Type	Required	Derived
p_location_type	IN	VARCHAR2	x	No
p_location_idTbl	IN	WSH_LOCATIO NS_PKG.ID_Tbl _Type	-	No
p_address_1Tbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	No
p_address_2Tbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	No
p_countryTbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	
p_stateTbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	
p_provinceTbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	
p_countyTbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	
p_cityTbl	IN	WSH_LOCATIO NS_PKG.Addres s_Tbl_Type	-	

Parameter Name	Usage	Type	Required	Derived
p_postal_codeTbl	IN	WSH_LOCATION NS_PKG.Addresses_Tbl_Type	-	
p_party_site_numberTbl	IN	WSH_LOCATION NS_PKG.LocationCode_Tbl_Type	-	
p_location_codeTbl	IN	WSH_LOCATION NS_PKG.LocationCode_Tbl_Type	-	
x_use_custom_ui_location	OUT	VARCHAR2	x	No
x_custom_ui_location_codeTbl	OUT	WSH_LOCATION NS_PKG.LocationCode_Tbl_Type	-	Yes

p_location_type

The parameter can have value 'HR' for internal location and 'HZ' for external location.

p_location_idTbl

The parameter corresponds to source location id in wsh_locations table.

p_address_1Tbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_address_2Tbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_countryTbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_StateTbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_provinceTbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_countyTbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_cityTbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_postal_codeTbl

Should be valid element of wsh_locations, hz_locations and hr_locations_all table.

p_party_site_numberTbl

Should be valid element of wsh_locations, hz_party_sites table.

p_location_codeTbl

Should be valid element of wsh_locations or hr_locations_all table.

x_use_custom_ui_location

Should have a valid value 'Y' or 'N'

p_custom_ui_loc_codeTbl

Should be a valid value and not more than 500 characters.

```

---Sample code start--
IF p_location_type = 'HZ' THEN
  FOR i IN p_location_idTbl.FIRST..p_location_idTbl.LAST
    LOOP
      x_custom_ui_loc_codeTbl(i) := substrb((p_party_site_numberTbl(i)||
:
'|p_address_1Tbl(i)||'-'||p_address_2Tbl(i)||'-'||p_cityTbl(i)||'-'||nv
l(p_stateTbl(i),p_provinceTbl(i))||'-'||
p_postal_codeTbl(i)||'-'||p_countryTbl(i)),1,500);
    END LOOP;
ELSIF p_location_type = 'HR' THEN

  FOR i IN p_location_idTbl.FIRST..p_location_idTbl.LAST
    LOOP
      x_custom_ui_loc_codeTbl(i) := substrb((p_location_codeTbl(i)||'
:
'|p_address_1Tbl(i)||'-'||p_address_2Tbl(i)||'-'||p_cityTbl(i)||'-'||p_
stateTbl(i)||'-'|| p_postal_codeTbl(i)||'-'||p_countryTbl(i)),1,500);
    END LOOP;
  END IF;
EXCEPTION
WHEN others THEN
  l_sqlcode := SQLCODE;
  l_sqlerr := SQLERRM;
  WSH_UTIL_CORE.printmsg('In the Others Exception of
WSH_CUSTOM_PUB.ui_location_code');
  WSH_UTIL_CORE.printmsg(l_sqlcode);
  WSH_UTIL_CORE.printmsg(l_sqlerr);
---Sample code end --

```

Standard Validation

There is no validation done for parameters passed as it is a custom procedure. For specific information on the data referenced by these columns, see

Oracle Electronic Technical Reference Manual (eTRM).

This API cannot be used alone. The API is always called from the package WSH_LOCATIONS_PKG. All validations are done in package WSH_LOCATIONS_PKG.

Calculate_TP_Dates API Features

The Calculate_TP_Dates API consists of the following entities:

This API enables to set the delivery window dates based on different parameters from Oracle Order Management. Customers can customize their calculation of the delivery window dates (Earliest/Latest Ship Dates and Earliest/Latest Delivery Dates). These dates will be then used for the population of values at the delivery detail level and will get propagated to the container or delivery levels at action points such as assign/pack etc.

Functional Overview

This API enables the user to set the delivery window dates based on different parameters from order management. Customers can customize their calculation of the delivery window dates (Earliest/Latest Ship Dates and Earliest/Latest Delivery Dates). These will be then used for population of values at the delivery detail level and will get propagated to the container or delivery levels at action points such as assign/pack etc.

Procedure Parameter Descriptions

The following table lists all parameters used by the public WSH_CUSTOM_PUB.Calculate_TP_Dates. All of the inbound and outbound parameters are listed. Additional information on these parameters follows in the table.

Parameter Name	Usage	Type	Required	Derived
p_source_line_id	IN	Number	-	No
p_source_code	IN	VARCHAR2	-	No
x_earliest_pickup_date	OUT	DATE	-	Yes
x_latest_pickup_date	OUT	DATE	-	Yes
x_earliest_dropoff_date	OUT	DATE	-	Yes

Parameter Name	Usage	Type	Required	Derived
x_latest_dropoff_date	OUT	DATE	-	Yes
x_modified	OUT	VARCHAR2	x	No

p_source_line_id

Should be valid element of wsh_delivery_details.source_line_id

p_source_code

Should be valid element of wsh_delivery_details.source_code.

x_earliest_pickup_date

Should be a valid date.

x_latest_pickup_date

Should be a valid date.

x_earliest_dropoff_date

Should be a valid date.

x_latest_dropoff_date

Should be a valid date.

x_modified

Should be a valid value 'N' or 'Y'. The default value is 'N'.

Standard Validation

There is no validation done for parameters passed as it is a custom procedure. For specific information on the data referenced by these columns, see *Oracle Electronic Technical Reference Manual (eTRM)*.

Example of API Packages and Procedures

The following examples demonstrate 5 procedures that use the Shipping public APIs. For more information about using the package, see the Script section following the Procedures section.

Package

Create or replace package WSH_SHIP_API as

```
procedure SHIP_CONFIRM_EXAMPLE1 (
```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2);

```
procedure SHIP_CONFIRM_EXAMPLE2 (
```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2);

```
procedure SHIP_CONFIRM_EXAMPLE3 (
```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2);

```
procedure UPDATE_SERIAL_EXAMPLE (
```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2);

```
procedure MANUAL_PACK_EXAMPLE (
```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2);

```
END WSH_SHIP_API;
/
-- show errors package WSH_SHIP_API;
```

Procedures

Create or replace package body WSH_SHIP_API as

/*

Example 1:

This procedure is used as an example of how to ship confirm a delivery consisting of delivery details that have already been pick released and assigned to a delivery:

Call WSH_DELIVERY_DETAILS_PUB.Update_Shipping_Attributes API to update the corresponding delivery details to ship all quantities in the first delivery detail; to back order all in the second; to stage all in the third delivery detail.

Call WSH_FREIGHT_COSTS_PUB.Create_Update_Freight_Costs API to create freight costs for the delivery that the details have been assigned to.

Call WSH_CONTAINER_PUB.Auto_Pack to pack the delivery lines into a container.

Call WSH_DELIVERIES_PUB.Delivery_Action API to ship confirm the delivery.

*/

procedure SHIP_CONFIRM_EXAMPLE1(

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2) IS

/*

--Standard Parameters.

p_api_version_number	NUMBER;
init_msg_list	VARCHAR2(30);
x_msg_details	VARCHAR2(3000);
x_msg_summary	VARCHAR2(3000);
p_validation_level	NUMBER;
p_commit	VARCHAR2(30);

--Parameters for WSH_DELIVERY_DETAILS_PUB.Update_Shipping_Attributes.

source_code	VARCHAR2 (15);
-------------	----------------

changed_attributes	WSH_DELIVERY_DETAILS_PUB.ChangedAt
tributeTabType;	

--Parameters for WSH_CONTAINER_PUB.Auto_Pack.

p_entity_tab	WSH_UTIL_CORE.id_tab_type;
--------------	----------------------------

p_entity_type	VARCHAR2 (30);
---------------	----------------

p_group_id_tab	WSH_UTIL_CORE.id_tab_type;
----------------	----------------------------

p_pack_cont_flag	varchar2 (30);
------------------	----------------

x_cont_inst_tab	WSH_UTIL_CORE.id_tab_type;
-----------------	----------------------------

--Parameters for WSH_FREIGHT_COSTS_PUB.Create_Update_Freight_Costs.

action_code	VARCHAR2 (15);
-------------	----------------

pub_freight_costs	WSH_FREIGHT_COSTS_PUB.PubFreightCo
stRecType;	

freight_cost_id	NUMBER;
-----------------	---------

--Parameters for WSH_DELIVERIES_PUB.Delivery_Action.

p_action_code	VARCHAR2 (15);
---------------	----------------

p_delivery_id	NUMBER;
---------------	---------

p_delivery_name	VARCHAR2 (30);
-----------------	----------------

p_asg_trip_id	NUMBER;
---------------	---------

p_asg_trip_name	VARCHAR2 (30);
-----------------	----------------

p_asg_pickup_stop_id	NUMBER;
p_asg_pickup_loc_id	NUMBER;
p_asg_pickup_loc_code	VARCHAR2 (30) ;
p_asg_pickup_arr_date	DATE;
p_asg_pickup_dep_date	DATE;
p_asg_dropoff_stop_id	NUMBER;
p_asg_dropoff_loc_id	NUMBER;
p_asg_dropoff_loc_code	VARCHAR2 (30) ;
p_asg_dropoff_arr_date	DATE;
p_asg_dropoff_dep_date	DATE;
p_sc_action_flag	VARCHAR2 (10) ;
p_sc_close_trip_flag	VARCHAR2 (10) ;
p_sc_create_bol_flag	VARCHAR2 (10) ;
p_sc_stage_del_flag	VARCHAR2 (10) ;
p_sc_trip_ship_method	VARCHAR2 (30) ;
p_sc_actual_dep_date	VARCHAR2 (30) ;
p_sc_report_set_id	NUMBER;
p_sc_report_set_name	VARCHAR2 (60) ;
p_wv_override_flag	VARCHAR2 (10) ;
x_trip_id	VARCHAR2 (30) ;
x_trip_name	VARCHAR2 (30) ;

```

/*Handle exceptions*/
fail_api EXCEPTION;

BEGIN

/* Initialize return status*/
x_return_status := WSH_UTIL_CORE.G_RET_STS_SUCCESS;

Call this procedure to initialize applications parameters. To determine
parameter values, refer to the Application Parameter Initialization
section of this chapter.*/
FND_GLOBAL.APPS_INITIALIZE(user_id => 1001594

                                ,resp_id =>52892

                                ,resp_appl_id =>660);

/* Values for updating delivery details to ship all quantities in the
first line, stage everything in the second line, and back order all in
the third. It is assumed that the user knows the quantities in each
line.
*/

    source_code := 'OE'; -- The only source code that should be used by the
API
    changed_attributes(1).delivery_detail_id := 13431; -- Ship All
quantities in this detail.
    changed_attributes(1).shipped_quantity := 1;
    changed_attributes(2).source_line_id := 13432; -- Back Order All in
this -- detail.

    changed_attributes(2).shipped_quantity := 0;
    changed_attributes(2).cycle_count_quantity := 2;
    changed_attributes(3).source_line_id := 13433; -- Stage All in this
detail.
    changed_attributes(3).shipped_quantity := 0;
    changed_attributes(3).cycle_count_quantity := 0;

--Call to WSH_DELIVERY_DETAILS_PUB.Update_Shipping_Attributes.
WSH_DELIVERY_DETAILS_PUB.Update_Shipping_Attributes(


```

p_api_version_number	=> 1.0,
p_init_msg_list	=> init_msg_list,
p_commit	=> p_commit,
x_return_status	=> x_return_status,
x_msg_count	=> x_msg_count,
x_msg_data	=> x_msg_data,
p_changed_attributes	=> changed_attributes,

p_source_code	=> source_code);
---------------	------------------

```
if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;
```

```
/* Values for creating freight costs for the delivery created for the
above delivery details. The delivery can be queried for the respective
delivery detail through wsh_delivery_assignments.
*/
```

pub_freight_costs.freight_cost_typ	:= 1;
e_id	

pub_freight_costs.unit_amount	:= 20;
-------------------------------	--------

pub_freight_costs.currency_code	:= 'USD';
---------------------------------	-----------

pub_freight_costs.delivery_id	:= 5341;
-------------------------------	----------

```
--Call to WSH_FREIGHT_COSTS_PUB.Create_Update_Freight_Costs.
WSH_FREIGHT_COSTS_PUB.Create_Update_Freight_Costs(
```

p_api_version_number	=> 1.0,
----------------------	---------

p_init_msg_list	=> init_msg_list,
-----------------	-------------------

p_commit	=> p_commit,
----------	--------------

x_return_status	=> x_return_status,
-----------------	---------------------

x_msg_count	=> x_msg_count,
-------------	-----------------

x_msg_data	=> x_msg_data,
------------	----------------

p_pub_freight_costs	=> pub_freight_costs,
---------------------	-----------------------

p_action_code	=> 'CREATE',
---------------	--------------

x_freight_cost_id	=> freight_cost_id);
-------------------	----------------------

```

if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;
/* Values for autopacking the delivery details to a container.
*/
p_entity_tab(1) := 13431;
p_entity_tab(2) := 13432;
p_entity_tab(3) := 13434;

--Call to WSH_CONTAINER_PUB.Auto_Pack
WSH_CONTAINER_PUB.Auto_Pack(

```

p_api_version	=> 1.0,
p_init_msg_list	=> init_msg_list,
p_commit	=> p_commit,
p_validation_level	=> p_validation_level,
x_return_status	=> x_return_status,
x_msg_count	=> x_msg_count,
x_msg_data	=> x_msg_data,
p_entity_tab	=> p_entity_tab,
p_entity_type	=> 'L',
p_group_id_tab	=> p_group_id_tab,
p_pack_cont_flag	=> p_pack_cont_flag,
x_cont_inst_tab	=> x_cont_inst_tab);

```

if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;

/* Values for Ship Confirming the delivery.
*/

```

p_action_code	:= 'CONFIRM'; -- The action code for ship confirm
p_delivery_id	:= 5341; -- The delivery that needs to be confirmed

p_delivery_name	:= '5341'; -- The delivery name,
p_sc_action_flag	:= 'S'; -- Ship entered quantity.
p_sc_close_trip_flag	:= 'Y'; -- Close the trip after ship confirm
p_sc_trip_ship_method	:= 'GROUND'; -- The ship method code

```
-- Call to WSH_DELIVERIES_PUB.Delivery_Action.
WSH_DELIVERIES_PUB.Delivery_Action(
```

p_api_version_number	=> 1.0,
p_init_msg_list	=> init_msg_list,
x_return_status	=> x_return_status,
x_msg_count	=> x_msg_count,
x_msg_data	=> x_msg_data,
p_action_code	=> p_action_code,
p_delivery_id	=> p_delivery_id,
p_delivery_name	=> p_delivery_name,
p_asg_trip_id	=> p_asg_trip_id,
p_asg_trip_name	=> p_asg_trip_name,
p_asg_pickup_stop_id	=> p_asg_pickup_stop_id,
p_asg_pickup_loc_id	=> p_asg_pickup_loc_id,
p_asg_pickup_loc_code	=> p_asg_pickup_loc_code,
p_asg_pickup_arr_date	=> p_asg_pickup_arr_date,
p_asg_pickup_dep_date	=> p_asg_pickup_dep_date,
p_asg_dropoff_stop_id	=> p_asg_dropoff_stop_id,

p_asg_dropoff_loc_id	=> p_asg_dropoff_loc_id,
p_asg_dropoff_loc_code	=> p_asg_dropoff_loc_code,
p_asg_dropoff_arr_date	=> p_asg_dropoff_arr_date,
p_asg_dropoff_dep_date	=> p_asg_dropoff_dep_date,
p_sc_action_flag	=> p_sc_action_flag,
p_sc_close_trip_flag	=> p_sc_close_trip_flag,
p_sc_create_bol_flag	=> p_sc_create_bol_flag,
p_sc_stage_del_flag	=> p_sc_stage_del_flag,
p_sc_trip_ship_method	=> p_sc_trip_ship_method,
p_sc_actual_dep_date	=> p_sc_actual_dep_date,
p_sc_report_set_id	=> p_sc_report_set_id,
p_sc_report_set_name	=> p_sc_report_set_name,
p_wv_override_flag	=> p_wv_override_flag,
x_trip_id	=> x_trip_id,
x_trip_name	=> x_trip_name);

```

if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;

exception

when fail_api then
    WSH_UTIL_CORE.get_messages('Y', x_msg_summary, x_msg_details,
    x_msg_count);
    if x_msg_count > 1 then
        x_msg_data := x_msg_summary || x_msg_details;
    else
        x_msg_data := x_msg_summary;
    end if;
END SHIP_CONFIRM_EXAMPLE1;

/**Example 2
This procedure is used as an example to ship confirm delivery details
that were not assigned to a delivery.

Call WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY to create a new delivery.
Call WSH_DELIVERY_DETAILS_PUB.Detail_to_Delivery to assign the delivery
details to the new delivery.
Call WSH_DELIVERIES_PUB.Delivery_Action to ship confirm.
**/
procedure SHIP_CONFIRM_EXAMPLE2(

```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2) IS

-- Standard Parameters.

p_api_version_number	NUMBER;
init_msg_list	VARCHAR2(30);
x_msg_details	VARCHAR2(3000);
x_msg_summary	VARCHAR2(3000);
p_validation_level	NUMBER;
commit	VARCHAR2(30);

-- Parameters for WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY

action_code	VARCHAR2 (15) ;
delivery_id	NUMBER;
delivery_info	WSH_DELIVERIES_PUB.Delivery_Pub_Re c_Type;
name	VARCHAR2 (30) ;

-- Parameters for WSH_DELIVERY_DETAILS_PUB.Detail_to_Delivery

p_delivery_id	NUMBER;
delivery_name	VARCHAR2 (30) ;
p_TabOfDelDet	WSH_DELIVERY_DETAILS_PUB.id_tab_ty pe;
p_action	VARCHAR2 (30) ;

-- Parameters for WSH_DELIVERIES_PUB.Delivery_Action.

p_action_code	VARCHAR2 (15) ;
p_delivery_id	NUMBER;
p_delivery_name	VARCHAR2 (30) ;
p_asg_trip_id	NUMBER;
p_asg_trip_name	VARCHAR2 (30) ;
p_asg_pickup_stop_id	NUMBER;
p_asg_pickup_loc_id	NUMBER;
p_asg_pickup_loc_code	VARCHAR2 (30) ;
p_asg_pickup_arr_date	DATE;
p_asg_pickup_dep_date	DATE;
p_asg_dropoff_stop_id	NUMBER;

p_asg_dropoff_loc_id	NUMBER;
p_asg_dropoff_loc_code	VARCHAR2(30);
p_asg_dropoff_arr_date	DATE;
p_asg_dropoff_dep_date	DATE;
p_sc_action_flag	VARCHAR2(10);
p_sc_close_trip_flag	VARCHAR2(10);
p_sc_create_bol_flag	VARCHAR2(10);
p_sc_stage_del_flag	VARCHAR2(10);
p_sc_trip_ship_method	VARCHAR2(30);
p_sc_actual_dep_date	VARCHAR2(30);
p_sc_report_set_id	NUMBER;
p_sc_report_set_name	VARCHAR2(60);
p_wv_override_flag	VARCHAR2(10);
x_trip_id	VARCHAR2(30);
x_trip_name	VARCHAR2(30);

/*Handle exceptions*/

fail_api	EXCEPTION;
----------	------------

```

BEGIN
  -- Initialize return status
  x_return_status := WSH_UTIL_CORE.G_RET_STS_SUCCESS;
  -- Values for WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY
  -- Create a new delivery for the following

```

```

delivery_info.initial_pickup_locat := 204;
ion_id

```

```
delivery_info.ultimate_dropoff_location_id := 840;
```

```
delivery_info.gross_weight := 10;
```

```
delivery_info.ship_method_code := 'UPS';
```

```
p_action_code := 'CREATE';
```

```
-- Call to WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY  
WSH_DELIVERIES_PUB.CREATE_UPDATE_DELIVERY(
```

```
p_api_version_number => 1.0,
```

```
p_init_msg_list => init_msg_list,
```

```
x_return_status => return_status,
```

```
x_msg_count => msg_count,
```

```
x_msg_data => msg_data,
```

```
p_action_code => p_action_code,
```

```
p_delivery_info => delivery_info,
```

```
p_delivery_name => delivery_name,
```

```
x_delivery_id => delivery_id,
```

```
x_name => name );
```

```
if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then  
  raise fail_api;  
end if;
```

```
p_delivery_id := delivery_id;
```

```
pub_freight_costs.delivery_id := delivery_id;
```

```
-- Values for WSH_DELIVERY_DETAILS_PUB.Detail_to_Delivery  
-- Call Detail_to_Delivery with an action code of ASSIGN to  
assign details to a
```

-- delivery.

p_TabOfDelDets(1)	:= 13463;
p_TabOfDelDets(2)	:= 13464;
p_action	:= 'ASSIGN';

-- Call to WSH_DELIVERY_DETAILS_PUB.Detail_to_Delivery.
WSH_DELIVERY_DETAILS_PUB.Detail_to_Delivery(

p_api_version	=> 1.0,
p_init_msg_list	=> init_msg_list,
p_commit	=> commit,
p_validation_level	=> p_validation_level,
x_return_status	=> return_status,
x_msg_count	=> msg_count,
x_msg_data	=> msg_data,
p_TabOfDelDets	=> p_TabOfDelDets,
p_action	=> p_action,
p_delivery_id	=> p_delivery_id,
p_delivery_name	=> delivery_name);

if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
 raise fail_api;
end if;

-- Values for Ship Confirming the delivery.

p_action_code	:= 'CONFIRM'; -- The action code for ship confirm
p_delivery_id	:= 5341; -- The delivery that needs to be confirmed

p_delivery_name	:= '5341'; -- The delivery name,
p_sc_action_flag	:= 'S'; -- Ship entered quantity.
p_sc_close_trip_flag	:= 'Y'; -- Close the trip after ship confirm
p_sc_trip_ship_method	:= 'UPS'; -- The ship method code

```
-- Call to WSH_DELIVERIES_PUB.Delivery_Action.
WSH_DELIVERIES_PUB.Delivery_Action(
```

p_api_version_number	=> 1.0,
p_init_msg_list	=> init_msg_list,
x_return_status	=> x_return_status,
x_msg_count	=> x_msg_count,
p_action_code	=> p_action_code,
p_delivery_id	=> p_delivery_id,
p_delivery_name	=> p_delivery_name,
p_asg_trip_id	=> p_asg_trip_id,
p_asg_trip_name	=> p_asg_trip_name,
p_asg_pickup_stop_id	=> p_asg_pickup_stop_id,
p_asg_pickup_loc_id	=> p_asg_pickup_loc_id,
p_asg_pickup_loc_code	=> p_asg_pickup_loc_code,
p_asg_pickup_arr_date	=> p_asg_pickup_arr_date,
p_asg_pickup_dep_date	=> p_asg_pickup_dep_date,
p_asg_dropoff_stop_id	=> p_asg_dropoff_stop_id,
p_asg_dropoff_loc_id	=> p_asg_dropoff_loc_id,

p_asg_dropoff_loc_code	=> p_asg_dropoff_loc_code,
p_asg_dropoff_arr_date	=> p_asg_dropoff_arr_date,
p_asg_dropoff_dep_date	=> p_asg_dropoff_dep_date,
p_sc_action_flag	=> p_sc_action_flag,
p_sc_close_trip_flag	=> p_sc_close_trip_flag,
p_sc_create_bol_flag	=> p_sc_create_bol_flag,
p_sc_stage_del_flag	=> p_sc_stage_del_flag,
p_sc_trip_ship_method	=> p_sc_trip_ship_method,
p_sc_actual_dep_date	=> p_sc_actual_dep_date,
p_sc_report_set_id	=> p_sc_report_set_id,
p_sc_report_set_name	=> p_sc_report_set_name,
p_wv_override_flag	=> p_wv_override_flag,
x_trip_id	=> x_trip_id,
x_trip_name	=> x_trip_name);

```

if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;

exception

when fail_api then
    WSH_UTIL_CORE.get_messages('Y', x_msg_summary, x_msg_details,
x_msg_count);
if x_msg_count > 1 then
    x_msg_data := x_msg_summary || x_msg_details;
else
    x_msg_data := x_msg_summary;
end if;

END SHIP_CONFIRM_EXAMPLE2;

/** Example 3:

This procedure can be used as an example on how to pick and ship confirm
a delivery in a single process.

Call WSH_DELIVERIES_PUB.DELIVERY_ACTION for the delivery to pick and
ship confirm the delivery in a single process.
**/
procedure SHIP_CONFIRM_EXAMPLE3(

```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2) IS

```
-- Standard Parameters.
```

p_init_msg_list	VARCHAR2(30) ;
x_msg_details	VARCHAR2(3000);
x_msg_summary	VARCHAR2(3000);

```
-- Parameters for WSH_DELIVERIES_PUB.DELIVERY_ACTION
```

p_action_code	VARCHAR2(2000);
p_delivery_id	NUMBER;
p_delivery_name	VARCHAR2(2000) ;

p_asg_trip_id	NUMBER ;
p_asg_trip_name	VARCHAR2(2000) ;
p_asg_pickup_stop_id	NUMBER ;
p_asg_pickup_loc_id	NUMBER ;
p_asg_pickup_stop_seq	NUMBER ;
p_asg_pickup_loc_code	VARCHAR2(2000) ;
p_asg_pickup_arr_date	DATE ;
p_asg_pickup_dep_date	DATE ;
p_asg_dropoff_stop_id	NUMBER ;
p_asg_dropoff_loc_id	NUMBER ;
p_asg_dropoff_stop_seq	NUMBER ;
p_asg_dropoff_loc_code	VARCHAR2(2000) ;
p_asg_dropoff_arr_date	DATE ;
p_asg_dropoff_dep_date	DATE ;
p_sc_action_flag	VARCHAR2(2000) ;
p_sc_intransit_flag	VARCHAR2(2000) ;
p_sc_close_trip_flag	VARCHAR2(2000) ;
p_sc_create_bol_flag	VARCHAR2(2000) ;
p_sc_stage_del_flag	VARCHAR2(2000) ;
p_sc_trip_ship_method	VARCHAR2(2000) ;
p_sc_actual_dep_date	DATE ;
p_sc_report_set_id	NUMBER ;
p_sc_report_set_name	VARCHAR2(2000) ;

p_sc_defer_interface_flag	VARCHAR2(2000) ;
p_sc_send_945_flag	VARCHAR2(2000) ;
p_sc_rule_id	NUMBER ;
p_sc_rule_name	VARCHAR2(2000) ;
p_wv_override_flag	VARCHAR2(2000) ;
x_trip_id	VARCHAR2(2000) ;
x_trip_name	VARCHAR2(2000) ;

```

/*Handle exceptions*/
fail_api EXCEPTION;

```

```

Begin

```

```

-- Initialize return status
x_return_status := WSH_UTIL_CORE.G_RET_STS_SUCCESS;

```

```

/* Call this procedure to initialize applications parameters. To
determine parameter values, refer to the Application Paramater
Initialization section of this chapter. */
FND_GLOBAL.APPS_INITIALIZE(user_id => 1001594
,resp_id =>52892
,resp_appl_id =>660);

```

```

-- Values for WSH_DELIVERIES_PUB.DELIVERY_ACTION
-- Pick and Ship the delivery in a single step

```

```

p_action_code := 'PICK-SHIP';
p_delivery_id := 176120;

```

```

-- Call to WSH_DELIVERIES_PUB.DELIVERY_ACTION
WSH_DELIVERIES_PUB.Delivery_Action(

```

p_api_version_number	=> 1.0
,p_init_msg_list	=> p_init_msg_list
,x_return_status	=> x_return_status
,x_msg_count	=> x_msg_count
,x_msg_data	=> x_msg_data
,p_action_code	=> p_action_code

,p_delivery_id	=> p_delivery_id
,p_delivery_name	=> p_delivery_name
,p_asg_trip_id	=> p_asg_trip_id
,p_asg_trip_name	=> p_asg_trip_name
,p_asg_pickup_stop_id	=> p_asg_pickup_stop_id
,p_asg_pickup_loc_id	=> p_asg_pickup_loc_id
,p_asg_pickup_stop_seq	=> p_asg_pickup_stop_seq
,p_asg_pickup_loc_code	=> p_asg_pickup_loc_code
,p_asg_pickup_arr_date	=> p_asg_pickup_arr_date
,p_asg_pickup_dep_date	=> p_asg_pickup_dep_date
,p_asg_dropoff_stop_id	=> p_asg_dropoff_stop_id
,p_asg_dropoff_loc_id	=> p_asg_dropoff_loc_id
,p_asg_dropoff_stop_seq	=> p_asg_dropoff_stop_seq
,p_asg_dropoff_loc_code	=> p_asg_dropoff_loc_code
,p_asg_dropoff_arr_date	=> p_asg_dropoff_arr_date
,p_asg_dropoff_dep_date	=> p_asg_dropoff_dep_date
,p_sc_action_flag	=> p_sc_action_flag
,p_sc_intransit_flag	=> p_sc_intransit_flag
,p_sc_close_trip_flag	=> p_sc_close_trip_flag
,p_sc_create_bol_flag	=> p_sc_create_bol_flag
,p_sc_stage_del_flag	=> p_sc_stage_del_flag
,p_sc_trip_ship_method	=> p_sc_trip_ship_method
,p_sc_actual_dep_date	=> p_sc_actual_dep_date

,p_sc_report_set_id	=> p_sc_report_set_id
,p_sc_report_set_name	=> p_sc_report_set_name
,p_sc_defer_interface_flag	=> p_sc_defer_interface_flag
,p_sc_send_945_flag	=> p_sc_send_945_flag
,p_sc_rule_id	=> p_sc_rule_id
,p_sc_rule_name	=> p_sc_rule_name
,p_wv_override_flag	=> p_wv_override_flag
,x_trip_id	=> x_trip_id
,x_trip_name	=> x_trip_name

);

```
if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;
```

exception

```
when fail_api then
    WSH_UTIL_CORE.get_messages('Y', x_msg_summary, x_msg_details,
    x_msg_count);
if x_msg_count > 1 then
    x_msg_data := x_msg_summary || x_msg_details;
else
    x_msg_data := x_msg_summary;
end if;
End SHIP_CONFIRM_EXAMPLE3;
```

```
/** Example 4:
This procedure can be used as an example on how to update serial numbers
for a delivery detail.
Call WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES to update
serial numbers for a delivery detail.
**/
procedure UPDATE_SERIAL_EXAMPLE(
```

x_return_status	OUT	VARCHAR2,
x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2) IS

```

-- Standard Parameters.


```

p_init_msg_list	VARCHAR2(30) ;
x_msg_details	VARCHAR2(3000);
x_msg_summary	VARCHAR2(3000);
p_commit	VARCHAR2(30);

```

-- Parameters for WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES
(Overloaded)


```

p_changed_attributes	WSH_DELIVERY_DETAILS_PUB.CHANGEDAT TRIBUTETABTYPE ;
p_source_code	VARCHAR2(2000);
p_container_flag	VARCHAR2(2000) ;
p_serial_range_tab	WSH_GLBL_VAR_STRCT_GRP.ddSerialRan geTabType;

```

/*Handle exceptions*/
fail_api EXCEPTION;

Begin

-- Initialize return status
  x_return_status := WSH_UTIL_CORE.G_RET_STS_SUCCESS;

/* Call this procedure to initialize applications parameters. To
determine parameter values, refer to the Application Paramater
Initialization section of this chapter. */

FND_GLOBAL.APPS_INITIALIZE(user_id => 1001594
,resp_id =>52892
,resp_appl_id =>660);

-- Values for WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES
  p_source_code := 'OE';
  p_changed_attributes(1).delivery_detail_id := 320484;
  p_changed_attributes(1).shipped_quantity := 10;
  p_serial_range_tab(1).delivery_detail_id := 320484 ;
  p_serial_range_tab(1).from_serial_number := 'A01';
  p_serial_range_tab(1).to_serial_number := 'A10';
  p_serial_range_tab(1).quantity := 10;

-- Call to WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES
WSH_DELIVERY_DETAILS_PUB.UPDATE_SHIPPING_ATTRIBUTES(

```

p_api_version_number	=> 1.0
,p_init_msg_list	=> p_init_msg_list
,p_commit	=> p_commit
,x_return_status	=> x_return_status
,x_msg_count	=> x_msg_count
,x_msg_data	=> x_msg_data
,p_changed_attributes	=> p_changed_attributes
,p_source_code	=> p_source_code
,p_container_flag	=> p_container_flag
,p_serial_range_tab	=> p_serial_range_tab

);

```
if (l_x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
  raise fail_api;
end if;
```

exception

```
when fail_api then
  WSH_UTIL_CORE.get_messages('Y', x_msg_summary, x_msg_details,
x_msg_count);
if x_msg_count > 1 then
  x_msg_data := x_msg_summary || x_msg_details;
else
  x_msg_data := x_msg_summary;
end if;
End UPDATE_SERIAL_EXAMPLE;
```

/** Example 5:

This procedure can be used as an example on how to create a container and then manually pack a delivery detail.

Call WSH_CONTAINER_PUB.CREATE_CONTAINERS API to create container(s)
Call WSH_CONTAINER_PUB.CONTAINER_ACTIONS to pack the delivery lines into the container.

*/

procedure MANUAL_PACK_EXAMPLE (

x_return_status	OUT	VARCHAR2,
-----------------	-----	-----------

x_msg_count	OUT	NUMBER,
x_msg_data	OUT	VARCHAR2) IS

```
-- Standard Parameters.
```

p_init_msg_list	VARCHAR2(30) ;
x_msg_details	VARCHAR2(3000);
x_msg_summary	VARCHAR2(3000);
p_validation_level	NUMBER;
p_commit	VARCHAR2(30);

```
-- Parameters for WSH_CONTAINER_PUB.CREATE_CONTAINERS
```

p_container_item_id	NUMBER;
p_container_item_name	VARCHAR2(2000) ;
p_container_item_seg	FND_FLEX_EXT.SEGMENTARRAY ;
p_organization_id	NUMBER;
p_organization_code	VARCHAR2(2000) ;
p_name_prefix	VARCHAR2(2000) ;
p_name_suffix	VARCHAR2(2000) ;
p_base_number	NUMBER ;
p_num_digits	NUMBER ;
p_quantity	NUMBER ;
p_container_name	VARCHAR2(2000) ;
x_container_ids	WSH_UTIL_CORE.ID_TAB_TYPE ;

```
-- Parameters for WSH_CONTAINER_PUB.CONTAINER_ACTIONS
```

p_detail_tab	WSH_UTIL_CORE.ID_TAB_TYPE ;
p_container_name	VARCHAR2(2000);
p_cont_instance_id	NUMBER ;
p_container_flag	VARCHAR2(2000) ;
p_delivery_flag	VARCHAR2(2000) ;
p_delivery_id	NUMBER ;
p_delivery_name	VARCHAR2(2000) ;
p_action_code	VARCHAR2(2000);

```
/*Handle exceptions*/
fail_api EXCEPTION;
```

```
Begin
```

```
-- Initialize return status
x_return_status := WSH_UTIL_CORE.G_RET_STS_SUCCESS;
/* Call this procedure to initialize applications parameters. To
determine parameter values, refer to the Application Parameter
Initialization section of this chapter. */
```

```
FND_GLOBAL.APPS_INITIALIZE(user_id => 1001594
,resp_id =>52892
,resp_appl_id =>660);
```

```
-- Values for WSH_CONTAINER_PUB.CREATE_CONTAINERS
p_container_item_id := 4556;
p_organization_id := 207;
p_quantity number := 1
```

```
-- Call to WSH_CONTAINER_PUB.CREATE_CONTAINERS
WSH_CONTAINER_PUB.CREATE_CONTAINERS(
```

p_api_version	=> 1.0
,p_init_msg_list	=> p_init_msg_list
,p_commit	=> p_commit
,p_validation_level	=> p_validation_level
,x_return_status	=> x_return_status

,x_msg_count	=> x_msg_count
,x_msg_data	=> x_msg_data
,p_container_item_id	=> p_container_item_id
,p_container_item_name	=> p_container_item_name
,p_container_item_seg	=> p_container_item_seg
,p_organization_id	=> p_organization_id
,p_organization_code	=> p_organization_code
,p_name_prefix	=> p_name_prefix
,p_name_suffix	=> p_name_suffix
,p_base_number	=> p_base_number
,p_num_digits	=> p_num_digits
,p_quantity	=> p_quantity
,p_container_name	=> p_container_name
,x_container_ids	=> x_container_ids

```
);

if (x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;
/*Get the container id (X_CONTAINER_IDS) generated by CREATE_CONTAINER
API then run CONTAINER_ACTIONS to Pack it in the generated container.*/

-- Values for WSH_CONTAINER_PUB.CONTAINER_ACTIONS
p_container_name := X_CONTAINER_IDS(1); --Container_name
p_action_code := 'PACK'
p_detail_tab(1) := 320496; --Delivery detail to be packed

-- Call to WSH_CONTAINER_PUB.CONTAINER_ACTIONS
WSH_CONTAINER_PUB.CONTAINER_ACTIONS(
```

p_api_version	=> 1.0
,p_init_msg_list	=> p_init_msg_list

,p_commit	=> p_commit
,p_validation_level	=> p_validation_level
,x_return_status	=> x_return_status
,x_msg_count	=> x_msg_count
,x_msg_data	=> x_msg_data
,p_detail_tab	=> p_detail_tab
,p_container_name	=> p_container_name
,p_cont_instance_id	=> p_cont_instance_id
,p_container_flag	=> p_container_flag
,p_delivery_flag	=> p_delivery_flag
,p_delivery_id	=> p_delivery_id
,p_delivery_name	=> p_delivery_name
,p_action_code	=> p_action_code

```

);

if (l_x_return_status <> WSH_UTIL_CORE.G_RET_STS_SUCCESS) then
    raise fail_api;
end if;

exception

when fail_api then
    WSH_UTIL_CORE.get_messages('Y', x_msg_summary, x_msg_details,
x_msg_count);
if x_msg_count > 1 then
    x_msg_data := x_msg_summary || x_msg_details;
else
    x_msg_data := x_msg_summary;
end if;
End MANUAL_PACK_EXAMPLE;
END WSH_SHIP_API;
/
--show errors package body WSH_SHIP_API;
--COMMIT;

```

Script

```
set serveroutput on
```

```
Declare
```

```
x_return_status          VARCHAR2(15);  
  
x_msg_count              NUMBER;  
  
x_msg_data               VARCHAR2(3000);
```

```
Begin
```

```
WSH_SHIP_API.SHIP_CONFIRM_EXAMPLE1(
```

```
x_return_status          => x_return_status;  
  
x_msg_count              => x_msg_count;  
  
x_msg_data               => x_msg_data);
```

```
dbms_output.put_line('The return status: ' || x_return_status);  
dbms_output.put_line(x_msg_data);
```

```
End;
```

Oracle Shipping Execution XML Transactions

This chapter covers the following topics:

- XML Overview
- Oracle XML Gateway Setup
- Using XML Messaging with Carrier Manifesting
- Using XML Messaging with Third Party Warehousing
- Using XML Messaging with International Trade Management

XML Overview

XML messaging is used to transact data between Oracle Shipping Execution and trading partners, including carrier manifesting systems and third party warehouses. In order to use XML messaging for Oracle Shipping Execution transactions, the following are required:

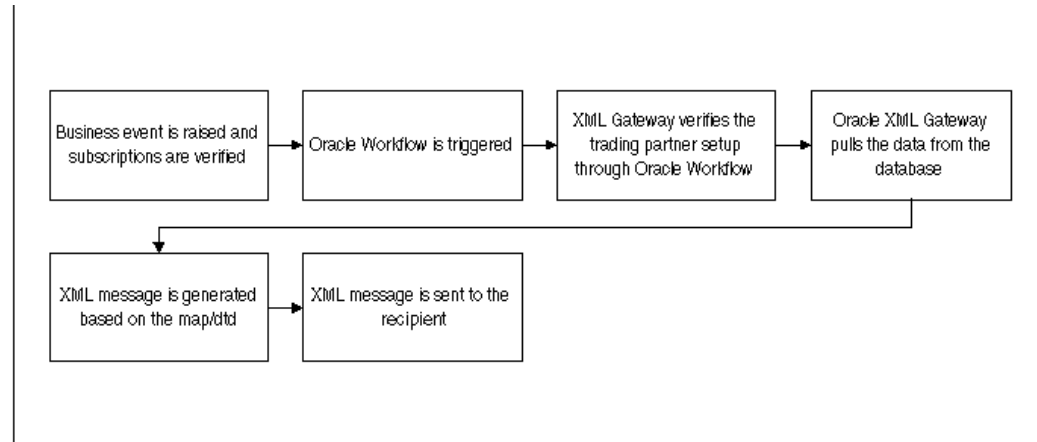
- Oracle XML Gateway setup, including Advanced Queuing, is required .
(See: *Oracle XML Gateway User's Guide*.)
- Oracle Workflow must be used to move the XML data from and to Oracle Shipping Execution. Workflows are seeded as part of Oracle Applications.
(See: *Oracle Workflow User's Guide*.)
- Business events must be subscribed to by each application using Oracle Workflow in order to move the XML data. Business events are seeded as part of Oracle Applications.
(See: *Oracle Workflow User's Guide*.)

Each Oracle Shipping Execution function that uses XML transactions requires

additional setup in Oracle XML Gateway, where trading partners must be defined.

The following graphic represents the Oracle XML transaction flow, including Oracle Workflow and Oracle XML Gateway.

XML Messaging Process



Oracle XML Gateway Setup

Oracle XML Gateway is a set of services that enables integration with Oracle Applications to support XML messaging. Open Applications Group (OAG) Data Type Definitions (DTDs) are used to map the XML messages. These DTDs are modified to meet Oracle-specific requirements and are seeded within the Oracle XML Gateway. You select the DTDs (or Maps in the Trading Partner Setup window) and assign them to each transaction during the trading partner setup.

Oracle XML Gateway processes events raised by the Oracle Applications and subscribes to inbound events for processing. Oracle XML Gateway uses Oracle Advanced Queuing to integrate with the Oracle Transport Agent to deliver messages to and receive messages from other applications.

Any application that receives XML messages from Oracle Shipping Execution or sends XML messages to Oracle Shipping Execution must be defined as a trading partner within Oracle XML Gateway.

You select XML transactions and enable them for each trading partner during the trading partner setup.

Code conversions are also seeded and defined in Oracle XML Gateway. Code conversions enable you to effectively communicate with trading partners by converting your codes (such as TON) into your trading partners' codes (such as TONS) by cross-referencing the codes.

As part of XML Gateway setup, Advanced Queuing is defined for the following:

- For outbound messages: A Queue is defined to hold complete XML messages created by Oracle XML Gateway. An XML message is picked up by the middleware, and then transmitted.
- For inbound messages: A Queue is defined to hold inbound XML messages that have no errors.
- For inbound messages: A Queue is defined to hold inbound XML error information.

Trading Partners

You must define trading partners in Oracle XML Gateway before their XML transactions can be processed. A trading partner is an entity (such as a customer, supplier, carrier, or internal organization) with which you exchange XML messages. Because a given entity may have several locations, you must define one trading partner for each entity location (that is, one trading partner for each customer address, supplier site, or carrier location).

Trading Partner data is used to:

- Link a particular address location in Oracle Applications to the trading partner definition in the Oracle XML Gateway.
- Provide a means of telling the Execution Engine which trading partner message map to use.

Note: A message map represents the relationship between the DTD and the XML message.

- Enable specific transactions for trading partners
- Determine how to deliver the message

Related Topics

Oracle XML Gateway User's Guide

Code Conversions

The Oracle XML Gateway code conversion function provides a method to cross-reference the codes defined in Oracle Applications, for example Unit of Measure, to codes used by trading partners, the XML standard, or other standard codes in the transactions.

For example, assume that Business World transmits a manifest to the XYZ Corporation and one of the values used is EACH. The XYZ Corporation processes the incoming data using its own values and instead of using EACH, XYZ uses EA. However, XYZ is

required to return the information to Business World using Business World's code values. Business World receives the response transactions with their own values based on the code conversion in Oracle XML Gateway. The value EA is converted to EACH when the data is received by Business World.

Oracle Applications provides seeded code conversion categories. The following seeded code conversion categories are provided to support the seeded code conversions:

- INVENTORY_ITEM
- UOM
- CUSTOMER_ITEM
- FOB
- SHIPMENT_PRIORITY
- CURRENCY_CODE
- CHARGE_TYPE
- WSH_SUBINVENTORY
- CARRIER_CODE
- CONTAINER_TYPE
- CUSTOMER_LOCATION_CODE
- FREIGHT_TERMS
- SERVICE_LEVEL
- WAREHOUSE_LOCATION_CODE
- WAREHOUSE_ORG
- WSH_TRANS_METHOD

Related Topics

Oracle XML Gateway User's Guide

Using XML Messaging with Carrier Manifesting

Oracle Shipping Execution enables you to send XML messages to carrier manifesting systems. These transactions contain delivery details to determine freight costs and to

print the required labels and paperwork. You can manually or automatically extract the delivery details and send them to a manifesting system. Once the carrier manifesting system responds, the delivery is automatically confirmed in Oracle Shipping Execution.

Before using XML messages for carrier manifesting, you must:

- Define your carriers (See: *Oracle Order Management Implementation Guide*)
- Enable manifesting for your carriers
- Enable manifesting for your organization
- Define your carrier manifesting system as a trading partner
- Define your organization as a trading partner

The following business events are used for carrier manifesting XML transactions:

- oracle.apps.wsh.sup.ssro: Business event to send the Shipment Request Outbound
- oracle.apps.wsh.sup.ssai: Business event to receive the Shipment Advice Inbound

The XML transactions used for carrier manifesting are:

- ShowShipment Request for outbound messages
- ShowShipment Advice for inbound messages

ShowShipment Request Transaction for Carrier Manifesting

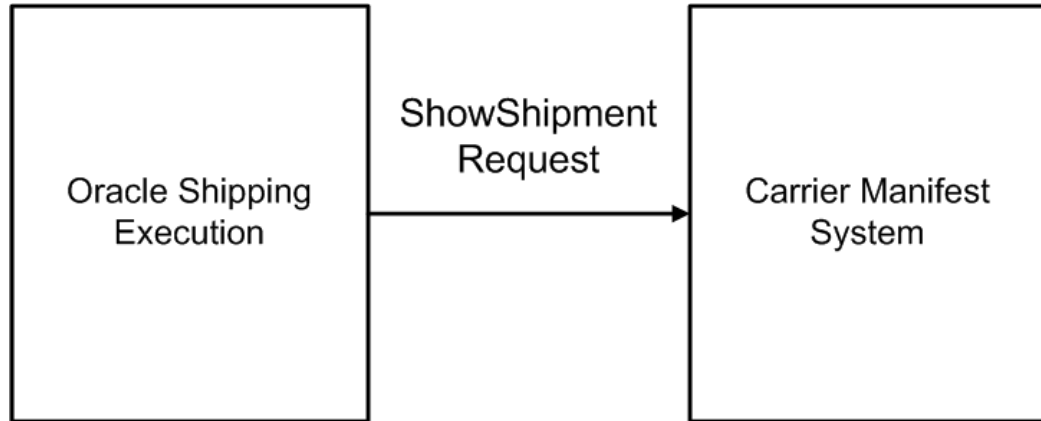
Show Shipment Request transaction (the XML equivalent of the ASC X12 940 transaction) is a modified version of the Open Applications Group (OAG) document type definition (DTD) show_shipment_005. This DTD is used to send shipment information from Oracle Shipping Execution to a carrier manifest system.

These transactions contain information such as:

- Delivery Name
- Customer
- Ship To Address
- Ship To Contact
- Requested Pick Up Date and Time
- Requested Delivery Date and Time
- Weight

- Carrier
- Service Level

ShowShipment Request for Carrier Manifesting

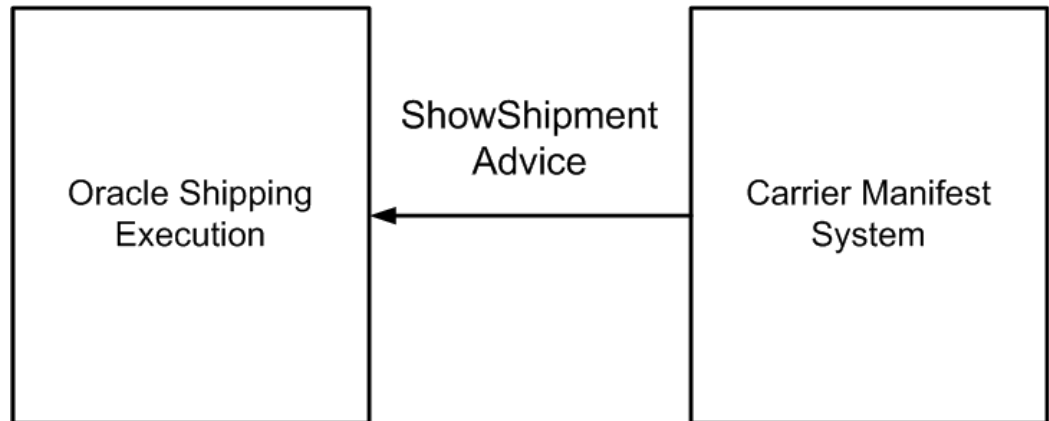


ShowShipment Advice Transaction for Carrier Manifesting

Show Shipment Advice transaction (the XML equivalent of the ASC X12 945 transaction) is a modified version of the OAG DTD `show_shipment_005`. This DTD is used to send shipment information from a carrier manifest system to Oracle Shipping Execution.

The ShowShipment Advice transaction supplies the actual carrier and service level used for the shipment, the tracking numbers assigned to the shipment or packages on the shipment, and the total weight, among other information.

ShowShipment Advice (show_shipment_005) for Carrier Manifesting



Defining Trading Partners for Carrier Manifesting

All organizations that send and receive XML transactions as part of carrier manifesting within Oracle Shipping Execution, must be defined as trading partners.

Trading partners are defined in Oracle XML Gateway.

To define organizations as trading partners to generate and send outbound messages:

1. Log into the Oracle applications using the XML Gateway responsibility.
2. Navigate to the Trading Partner Setup window.
3. Select Internal from the list of values in the Trading Partner Type field.
4. Select the organization's location code using the list of values in the Trading Partner Name field.
5. Select the concatenated address of the organization's location from the list of values in the Trading Partner Site field.
6. Enter the email address in the Company Admin Email field.
7. In the Trading Partner Details region, use the list of values in the Transaction Type field to choose FTE with the Transaction Sub Type of SSNO.
8. Use the list of values to select WSH_SSNO_OAG721_OUT in the Map field.
9. In the Connection/Hub field, select Direct or select a Hub through which to communicate with the carrier manifesting application.

Note: A DIRECT connect and a hub are the methods by which the

message can be communicated. The XML message can be sent directly to a trading partner, or sent to a trading partner using a hub. The hub then communicate the message to the trading partner. Select DIRECT to conduct business directly with a trading partner, or select a hub from the list of values. If a hub is selected, then select a trading partner from that hub.

See: *Oracle XML User's Guide*

If you choose to connect to a carrier manifesting application using Direct, the XML Gateway will require that you provide information in the following fields:

- Protocol Type
- Username
- Password
- Protocol address

If you choose to connect to a carrier manifesting application using a hub, Oracle XML Gateway will require that you select only a Username from the list of values. The rest of the information will default based on the information entered during the creation of the hub.

See: *Oracle XML Gateway User's Guide* for detailed information on defining a hub.

The Connection/Hub field is only required when the direction of the XML is OUT.

10. Enter a Source Trading Partner Location Code.

Note: If you are using a predefined hub, the Source Trading Partner Location Code is automatically populated.

11. Save your work.

12. Click Code Conversion to define new code conversions.

See: *Oracle XML Gateway User's Guide* for detailed information on defining code conversions.

Note: Code conversion categories are seeded. These include OAG and Oracle standard code conversions.

To define organizations as trading partners to receive inbound messages:

1. Log into the Oracle applications using the XML Gateway responsibility.

2. Navigate to the Trading Partner Setup window.
3. Select Internal from the list of values in the Trading Partner Type field.
4. Select the organization's location using the list of values in the Trading Partner Name field.
5. Select the organization location's address from the list of values in the Trading Partner Site field.
6. Enter the email address in the Company Admin Email field.
7. In the Trading Partner Details region, use the list of values in the Transaction Type field to choose FTE with the Transaction Sub Type of SSNI.
8. Use the list of values to select WSH_SSNI_OAG721_IN in the Map field.
9. Enter a Source Trading Partner Location Code.
10. Save your work.
11. Click Code Conversion to define new code conversions.

See: *Oracle XML Gateway User's Guide* for detailed information on defining code conversions.

Note: Code conversion categories are seeded. These include OAG and Oracle standard code conversions.

To define Transaction Type for Confirm Business Object Document (CBOD)

1. Start a new Trading Partner Detail line.
2. Use the list of values in the Transaction Type field to select ECX with External Transaction Type of BOD, Transaction Subtype as Confirm, and the Direction IN.
3. Use the list of values to select ECX_CBODI_OAG72_IN_CONFIRM in the Map field.
4. Enter source partner location code.

The source partner location code is identical to the source partner location code specified in the setup for outbound transactions.
5. Save your work.

Overview of XML and DTD for Carrier Manifesting

The following contains a sample DTD and a sample XML used for carrier manifesting in Oracle Shipping Execution . Also, a table is available that displays all required element/attributes, a description of each, and the Oracle table and column for each element/attribute, where applicable.

Document Type Definition (DTD) for Carrier Manifesting

The following is the DTD used for carrier manifesting in Oracle Shipping Execution.

Carrier Manifesting DTD

```
<! - ===== - >
<!ENTITY % RESOURCES SYSTEM "oagis_resources.dtd">
%RESOURCES;
<! - ===== - >
<!ELEMENT SHOW_SHIPMENT_005 (CNTROLAREA, DATAAREA+)>
<!ATTLIST VERB
  value CDATA #FIXED "SHOW">
<!ATTLIST NOUN
  value CDATA #FIXED "SHIPMENT">
<!ATTLIST REVISION
  value CDATA #FIXED "005">
<!ELEMENT DATAAREA (SHOW_SHIPMENT)>
<!ELEMENT SHOW_SHIPMENT (SHIPMENT, SHIPUNIT*)>
<!ELEMENT SHIPMENT ((%DATETIME.DOCUMENT;)?, (%AMOUNT.DECLAREVAL;)?,
  (%AMOUNT.DOCUMENT.T;)?, (%AMOUNT.ESTFREIGHT;)?, (%AMOUNT.FREIGHT.T;)?,
  (%AMOUNT.ITEM.T;)?, (%DATETIME.CREATION;)?, (%DATETIME.DELIVACT;)?,
  (%DATETIME.DELIVSCHED;)?, (%DATETIME.EARLSTSHIP;)?,
  (%DATETIME.LOADING;)?, (%DATETIME.NEEDDELV;)?, (%DATETIME.PROMDELV;)?,
  (%DATETIME.PROMSHIP;)?, (%DATETIME.SHIP;)?, (%DATETIME.SHIPSCHED;)?,
  (%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?,
  (%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
  (%QUANTITY.SHIPUNIT;)?, (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?,
  (%TEMPRATURE.ACTUAL;)?, (%TEMPRATURE.DELIVERY;)?,
  (%TEMPRATURE.LOADING;)?, DOCUMENTID, SHIPPERID, SYNCID, CARRIER?,
  CARRSRVLVL?, COUNTRYDST?, COUNTRYORG?, DESCRIPTN?, DISTCENTER?, DOCKID?,
  DOCTYPE?, DOCUMENTTRV?, DROPSHIP?, EXPORTLIC?, FRGHTCLS?, FRGHTTERMS?,
  HAZRDMATL?, HDRSTATUS?, IMPORTLIC?, LOADPOINT?, NOTES*, PARTIALSHP?,
  PRIORITY?, ROUTEID?, SHIPNOTES?, SHIPPOINT?, SHIPPRIOR?, SHIPSRVLVL?,
  SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, WAREHOUSE?,
  USERAREA?, PARTNER+, SHIPITEM*, ATTCHREF*, CHARGE*, DOCUMNTREF*)>
<!ELEMENT SHIPITEM ((%QUANTITY.ITEM;), (%AMOUNT.EXTENDED.T;)?,
  (%AMOUNT.ITEM.T;)?, (%QUANTITY.BACKORDERD;)?, (%QUANTITY.OPEN;)?,
  (%QUANTITY.ORDERED;)?, (%QUANTITY.SHIPPED;)?, (%QUANTITY.VOLUME;)?,
  (%QUANTITY.WEIGHT;)?, ITEM, COUNTRYORG?, DISPOSITN?, EXPORTLIC?,
  FRGHTCLS?, FRGHTITEM?, HAZRDMATL?, IMPORTLIC?, ITEMDESC?, ITEMRV?,
  ITEMVRX?, ITEMX?, LOTLEVEL*, NOTES*, OWNRSHPCDE?, PACKING?, PARTIALSHP?,
  SHIPNOTES?, SHIPPRIOR?, UPC?, USERAREA?, DOCUMNTREF*, INVDETAIL*,
  CHARGE*)
>
<!ELEMENT INVDETAIL ((%QUANTITY.ITEM;)?, DESCRIPTN?, DISPOSITN?,
  LOTLEVEL*, NOTES*, SERIALNUM?, USERAREA?)
>
<!ELEMENT SHIPUNIT ((%AMOUNT.DECLAREVAL;)?, (%AMOUNT.FREIGHT.T;)?,
  (%AMOUNT.ITEM.T;)?, (%AMOUNT.TOTAL.T;)?, (%DATETIME.DELIVACT;)?,
  (%DATETIME.DELIVSCHED;)?, (%DATETIME.LOADING;)?, (%DATETIME.SHIP;)?,
  (%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?, (%QUANTITY.HEIGHT;)?,
  (%QUANTITY.LENGTH;)?, (%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
  (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?, (%QUANTITY.WIDTH;)?,
  (%TEMPRATURE.ACTUAL;)?, (%TEMPRATURE.DELIVERY;)?,
  (%TEMPRATURE.LOADING;)?, CARRIER, SHIPPERID, TRACKINGID, CARRSRVLVL?,
  CONTAINRID?, CONTNRSEAL?, CONTNRTYPE?, COUNTRYDST?, COUNTRYORG?,
  DESCRIPTN?, EXPORTLIC?, FRGHTCLS?, FRGHTITEM?, FRGHTTERMS?, HAZRDMATL?,
  IMPORTLIC?, LOADPOINT?, NOTES*, SHIPMATLID?, SHIPNOTES?, SHIPPOINT?,
  SHIPSRVLVL?, SHPMTLDESC?, SHPUNITSEQ?, SHPUNITSTS?, SHPUNITTOT?,
  SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, USERAREA?, PARTNER+,
  ATTCHREF*, CHARGE*, CONTAINER*, DOCUMNTREF*, INVITEM*)
>
<!ELEMENT CONTAINER (CONTAINRID?, CONTNRSEAL?, CONTNRTYPE?, DESCRIPTN?,
```

```

NOTES*, PACKING?, PACKNGDESC?, PARCNTNRID?, SHIPMATLID?, SHPMTLDESC?,
USERAREA?, DOCUMNTREF*, INVITEM*, CHARGE*)
>
<!ELEMENT INVITEM ((%QUANTITY.ITEM;), (%AMOUNT.EXTENDED.T;)?,
(%AMOUNT.ITEM.T;)?, (%QUANTITY.VOLUME;)?, (%QUANTITY.WEIGHT;)?, ITEM,
COUNTRYORG?, FRGHTCLS?, FRGHTITEM?, HAZRDMATL?, ITEMDESC?, ITEMRV?,
ITEMRVX?, ITEMX?, LOTLEVEL*, NOTES*, OWNRSHPCODE?, SHIPNOTES?, UPC?,
USERAREA?, DOCUMNTREF*, INVDETAIL*)>
@
115.1.11510.1
log
@'mkbranch'
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ ->
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@
115.1.1159.1
log
@mkbranch
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ ->
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@
115.1.1158.1
log
@mkbranch
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ ->
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@
115.0
log
@initial version
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.0 2002/02/15 05:27:39
rvishnuv noship $ ->
d6 2

```

```

a7 2
    $Revision: 7.2.1 $
    $Date: 31 October 2001 $
d11 1
a11 1
    $Name: 161_show_shipment_005.dtd $
d22 1
a22 1
    SHIPMENT ( PARTNER+, SHIPITEM+, ATTCHREF*, CHARGE*, DOCUMNTREF*,)
d59 1
a59 1
<!ELEMENT SHIPMENT ((%DATETIME.DOCUMENT;)?, (%AMOUNT.DECLAREVAL;)?,
(%AMOUNT.DOCUMENT.T;)?, (%AMOUNT.ESTFREIGHT;)?, (%AMOUNT.FREIGHT.T;)?,
(%AMOUNT.ITEM.T;)?, (%DATETIME.CREATION;)?, (%DATETIME.DELIVACT;)?,
(%DATETIME.DELIVSCHED;)?, (%DATETIME.EARLSTSHIP;)?,
(%DATETIME.LOADING;)?, (%DATETIME.NEEDDELV;)?, (%DATETIME.PROMDELV;)?,
(%DATETIME.PROMSHIP;)?, (%DATETIME.SHIP;)?, (%DATETIME.SHIPSCHED;)?,
(%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?,
(%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
(%QUANTITY.SHIPUNIT;)?, (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?,
(%TEMPRATURE.ACTUAL;)?, (%TEMPRATURE.DELIVERY;)?,
(%TEMPRATURE.LOADING;)?, DOCUMENTID, SHIPPERID, SYNCID, CARRIER?,
CARRSRVLVL?, COUNTRYDST?, COUNTRYORG?, DESCRIPTN?, DISTCENTER?, DOCKID?,
DOCTYPE?, DOCUMENTRV?, DROPSHIP?, EXPORTLIC?, FRGHTCLS?, FRGHTTERMS?,
HAZRDMATL?, HDRSTATUS?, IMPORTLIC?, LOADPOINT?, NOTES*, PARTIALSHP?,
PRIORITY?, ROUTEID?, SHIPNOTES?, SHIPPOINT?, SHIPPRIOR?, SHIPSRVLVL?,
SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, WAREHOUSE?,
USERAREA?, PARTNER+, SHIPITEM+, ATTCHREF*, CHARGE*, DOCUMNTREF*)
@

```

Sample XML Transaction for Carrier Manifesting

Carrier Manifesting Sample XML

The following table lists each of the ShowShipment elements/attributes, whether they are required or optional, and whether they are used for inbound, outbound, or both. A sample OUTBOUND ShowShipment Request XML transaction for carrier manifesting follows the table and a sample INBOUND ShowShipment Request XML transaction for carrier manifesting follows the OUTBOUND sample.

The Required? column indicates whether the element/attribute is required or optional for INBOUND (I), OUTBOUND (O), or BOTH (B).

Carrier Manifesting XML Field Information

Field Name	Required?	Transaction Table.Column	Description
CNTROL AREA	Required (B)		The fields included in this area provide information about the XML document

Field Name	Required?	Transaction Table.Column	Description
BSR	Required (B)		Shows the Business Service Request name per OAGI
VERB	Required (B)		Value is SHOW
NOUN	Required (B)		Value is SHIPMENT
REVISION	Required (B)		Value is 006
SENDER	Required (B)		Provides information on the system that sends the document
LOGICALID	Required (B)		Sender system identifier
COMPONENT	Required (B)		Sender application name. Value is FTE
TASK	Required		Event or Action
REFERENCEID	Required (B)		Unique reference ID for this document
CONFIRMATION	Required (B)		Confirmation when document is received
LANGUAGE	Required (B)		Language in which the text fields are transmitted
CODEPAGE	Required (B)		Character set used in this XML document
AUTHID	Required (B)		System ID of sender. Value is APPS
DATETIME (CREATION)(T)	Required (B)		Creation date and time of the XML document

Field Name	Required?	Transaction Table.Column	Description
DATAAREA.SHOW_SHIPMENT.SHIPMENT	Required (B)		Contains the general document information for goods being transported to a particular partner destination
SHIPMENT.DATETIME	Required (B)	Sys date time	Contains date and time stamp of the latest update to the XML document
SHIPMENT.DOCUMENTID (Shipment)	Required (B)	WSH_TRANSACTION_HISTORY.DOCUMENT_ID	Shipment document ID
SHIPMENT.SHIPPERID	Required (B)	LOCATION_CODE from HR_LOCATIONS_ALL_TL based on the WSH_NEW_DELIVERIES.INITIAL_PICKUP_LOCATION_ID	Ship From ID
SHIPMENT.SYNCID	Required (B)		XML: "A" = Add, "D" = Delete "C" = Change EDI: "N" for Add, "F" for Delete EDI code conversions: "N" for Add, "R" for Change, "F" for Cancel
SHIPMENT.AMOUNT (DECLAREVAL)	Not used	Not used	Not used
SHIPMENT.AMOUNT (DOCUMENT) (T)	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.AMOUNT (ESTFREIGHT)	Not used	Not used	Not used
SHIPMENT.AMOUNT (FREIGHT) (T)	Optional (B)	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	Total freight charge
SHIPMENT.AMOUNT (ITEM) (T)	Not used	Not used	Not used
SHIPMENT.CARRIER	Required (B)	PARTY_NAME FROM HZ_PARTIES based on the column WSH_NEW_DELIVERIES.CARRIER_ID	Carrier ID
SHIPMENT.CARRIER_SRLVL	Optional (B)	WSH_NEW_DELIVERIES.SERVICE_LEVEL	Carrier service level
SHIPMENT.COUNTRYDST	Required (O)	WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID,	Destination country
SHIPMENT.COUNTRYORG	Not Used	Not Used	Not Used
SHIPMENT.DATETIME (CREATION)	Required (O)	WSH_NEW_DELIVERIES.CREATION_DATE	Creation date and time
SHIPMENT.DATETIME (DELIVACT)	Optional (I)	WSH_TRIP_STOPS.ACTUAL_ARRIVAL_DATE	Actual delivery date
SHIPMENT.DATETIME (DELIVSCHED)	Optional (B)	WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_DATE	Delivery scheduled date and time

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.DATETIME (EARLSTSHIP)	Not used	Not used	Not used
SHIPMENT.DATETIME (LOADING)	Not used	Not used	Not used
SHIPMENT.DATETIME (NEEDELV)	Optional (O)	WSH_NEW_DELIVERIES. .ULTIMATE_DROPOFF_DATE	Date and time the delivery is needed
SHIPMENT.DATETIME (PROMDELV)	Not used	Not used	Not used
SHIPMENT.DATETIME (PROMSHIP)	Not used	Not used	Not used
SHIPMENT.DATETIME (SHIP)	Optional (I)	WSH_TRIP_STOPS.ACTUAL_DEPARTURE_DATE	Actual ship date/time
SHIPMENT.DATETIME (SHIPSCHED)	Optional (O)	WSH_NEW_DELIVERIES.INITIAL_PICKUP_DATE	Shipment scheduled date and time
SHIPMENT.DESRIPTION	Optional (B)	WSH_NEW_DELIVERIES.DESCRPTION	Shipment description
SHIPMENT.DISTCENTER	Not used	Not used	Not used
SHIPMENT.DOCKID	Not used	Not used	Not used
SHIPMENT.DOCTYPE	Required (B)	'SR' (SHIPMENT REQUEST FOR OUTBOUND) AND 'SA' (SHIMENT ADVICE FOR INBOUND) at the Supplier Instance.	The OAG document name SR (SHIPMENT REQUEST) outbound transactions and SA (SHIPMENT ADVICE) for inbound transactions

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.DOCUMENTRV	Optional (B)		Revision number of this document '001' for 'Add' and '002' for 'Delete'
SHIPMENT.DROPSHIP	Not used	Not used	Not used
SHIPMENT.EXPORTLIC	Not used	Not used	Not used
SHIPMENT.FRIGHTCLS	Not used	Not used	Not used
SHIPMENT.FRIGHTTERMS	Optional (B)	WSH_NEW_DELIVERIES.FREIGHT_TERMINUS_CODE	Describes how the freight cost for a shipment of goods should be billed
SHIPMENT.HAZRDMATL	Not used	Not used	Not used
SHIPMENT.HDRSTATUS	Not used	Not used	Not used
SHIPMENT.IMPORTLIC	Not used	Not used	Not used
SHIPMENT.LOADPOINT	Not used	Not used	Not used
SHIPMENT.NOTES1 – NOTES9	Not used	Not used	Not used
SHIPMENT.PARTIALSHIP	Not used	Not used	Not used
SHIPMENT.PRIORITY	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.QUANTITY (ESTWEIGHT)	Optional (B)	WSH_NEW_DELIVERIES.GROSS_WEIGHT WSH_NEW_DELIVERIES.WEIGHT_UOM_CODE	The estimated weight of an item or container
SHIPMENT.QUANTITY (LOADINGWT)	Optional (B)	WSH_NEW_DELIVERIES.GROSS_WEIGHT – WSH_NEW_DELIVERIES.NET WEIGHT	The net weight at the loading time of the container in which the materials are being shipped
SHIPMENT.QUANTITY (NETWEIGHT)	Optional (B)	WSH_NEW_DELIVERIES.NET_WEIGHT WSH_NEW_DELIVERIES.WEIGHT_UOM_CODE	The net weight of the goods or materials being shipped
SHIPMENT.QUANTITY (SHIPUNIT)	Optional (B)	WSH_NEW_DELIVERIES.NUMBER_OF_LPN	This segment contains the quantity of units shipped. This represents the container(s), not the product shipped. An example of this is "4 truck loads" or "2 wooden crates".
SHIPMENT.QUANTITY (TOTWEIGHT)	Optional (B)	WSH_NEW_DELIVERIES.GROSS_WEIGHT WSH_NEW_DELIVERIES.WEIGHT_UOM_CODE	This segment quantifies the total weight of an item, or shipment
SHIPMENT.QUANTITY (VOLUME)	Optional (B)	WSH_NEW_DELIVERIES.VOLUME WSH_NEW_DELIVERIES.VOLUME_UOM_CODE	This segment quantifies the volume, or amount of space, that is occupied by an item, container or shipment

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.ROUTEID	Not used	Not used	Not used
SHIPMENT.RELEASENUM	Not used	Not used	Not used
SHIPMENT.RELEASENUMTOT	Not used	Not used	Not used
SHIPMENT.ROUTEITYPE	Not used	Not used	Not used
SHIPMENT.SHIPNOTES	Optional (B)	WSH_NEW_DELIVERIES.SHIPPING_MARKS	This is a free form area that contains special instructions
SHIPMENT.SHIPPOINT	Optional (B)	LOCATION FROM HZ_CUST_SITE_USES_ALL based on the WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID converted to code	Indicates the ship to location
SHIPMENT.SHIPPRIOR	Not used	Not used	Not used
SHIPMENT.SHIPSRVLVL	Optional (B)	WSH_NEW_DELIVERIES.SHIP_SERVICE_LEVEL	Shipment service level
SHIPMENT.SPECIALHND	Not used	Not used	Not used
SHIPMENT.STAGEPOINT	Not used	Not used	Not used
SHIPMENT.TEMPRACTURE (ACTUAL)	Not used	Not used	Not used
SHIPMENT.TEMPRACTURE (DELIVERY)	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.TEMPRA TURE (LOADING)	Not used	Not used	Not used
SHIPMENT.TRANS METHD	Optional (B)	WSH_NEW_DELIVE RIES.SHIP_MODE_O F_TRANSPORT	Identifies the general type of carrier transportation used to deliver goods
SHIPMENT.TRANST ERMS	Optional (B)	WSH_NEW_DELIVE RIES.FOB_CODE	Ownership transfer terms
SHIPMENT.WAREH OUSE	Optional (B)	WSH_NEW_DELIVE RIES.ORGANIZATIO N_ID converted to code	The storage facility for inventory

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.USERAREA	Optional (B)	WSH_NEW_DELIVERIES.DELIVERY_NAME (required)	The USERAREA is defined by embedding elements and attributes for each new Field Identifier or Segment needed within the USERAREA. When a new Field Identifier or a new Segment is determined to be necessary, and it is not included in the OAGIS specification, new tags can be developed by the project team for the specific integration project. These new tags can then be used to describe the fields and segments within the USERAREA. The USERAREA may contain multiple fields or segments coded in this way. These are all embedded within the one special field called USERAREA.
ORACLE.DELIVERY_NAME		WSH_NEW_DELIVERIES.LOADING_SEQUENCE	
ORACLE.LOADING_SEQUENCE		WSH_TRIPS.VEHICLE_NUM_PREFIX	
ORACLE.VEHICLE_NUM_PREFIX		WSH_TRIPS.VEHICLE_NUMBER	
ORACLE.VEHICLE_NUMBER		WSH_TRIPS.ROUTE_ID	
ORACLE.ROUTE_ID		WSH_TRIPS.ROUTING_INSTRUCTIONS	
ORACLE.ROUTING_INSTRUCTIONS		WSH_TRIP_STOPS.DEPARTURE_SEAL_CODE	
ORACLE.DEPARTURE_SEAL_CODE		WSH_TRANSACTION_HISTORY.ORIG_DOCUMENT_NUMBER storing the parent message's	
ORACLE.ORIG_DOC_ID		DOCUMENTID (Required for Inbound)	
ORACLE.CUSTOMER_NAME		HZ_PARTIES.PARTY_NAME based on the	
ORACLE.CUSTOMER_NUMBER		WSH_NEW_DELIVERIES.CUSTOMER_ID	
ORACLE.WAYBILL		HZ_CUST_ACCOUNTS.ACCOUNT_NUMBER based on the	
ORACLE.INTMED_SHIP_TO_LOCATION		WSH_NEW_DELIVERIES.customer_id	
ORACLE.MANUAL_WV		WSH_NEW_DELIVERIES.WAYBILL	
ORACLE.SHIPMENT_DIRECTION		LOCATION FROM HZ_CUST_SITE_USE	
ORACLE.OPERATOR			
ORACLE.DEScriptive_FLEXFIELDS			

Field Name	Required?	Transaction Table.Column	Description
		S_ALL based on the WSH_NEW_DELIVE RIES.INTMED_SHIP_ TO_LOCATION_ID WSH_NEW_DELIVE RIES.WV_FROZEN_F LAGWSH_NEW_DE LIVERIES.SHIPMEN T_DIRECTIONWSH_ TRIPS.OPERATOR WSH_NEW_DELIVE RY descriptive flex fields	
PARTNER.	Required (O)		This area is for the partner information

Field Name	Required?	Transaction Table.Column	Description
PARTNER.NAME	Required (O)	<p>If this partner segment is the Ship To segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID</p> <p>If this partner segment is the Ship from segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ORGANIZATION_ID</p> <p>If this partner segment is the Customer segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.CUSTOMER_ID</p> <p>If this partner segment is the Supplier segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ORGANIZATION_ID</p>	Corporate name of a business entity in the shipment address
PARTNER.ONETIME	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
PARTNER.PARTNRID	Required (O)	LOCATION FROM HZ_CUST_SITE_USES_ALL based on WSH_NEW_DELIVERIES.ULTIMATE_DROFF_LOCATION_ID converted to external value, or LOCATION_CODE FROM HR_LOCATIONS_ALL_TL based on WSH_NEW_DELIVERIES.INITIAL_PICKUP_LOCATION_ID AND ORGANIZATION_ID converted to external value.	The partner ID
PARTNER.PARTNRTYPE	Required (O)	WSH_LOOKUPS.MEANING based on lookup type = 'WSH_PARTNER_TYPE' (Values are 'ShipFrom' or 'ShipTo')	Indicates the type of partner entity
PARTNER.ACTIVE	Not used	Not used	Not used
PARTNER.DESCRPTION	Not used	Not used	Not used
PARTNER.DUNSNUMBER	Not Used	Not Used	Not Used
PARTNER.GLENTITYS	Not Used	Not Used	Not Used
PARTNER.NAME2-NAME9	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
PARTNER.PARENTID	Not used	Not used	Not used
PARTNER.PARTNRATG	Not used	Not used	Not used
PARTNER.PARTNRROLE	Not used	Not used	Not used
PARTNER.PAYMETHOD	Not used	Not used	Not used
PARTNER.TAXEXEMPT	Not used	Not used	Not used
PARTNER.TAXID	Not used	Not used	Not used
PARTNER.TERMID	Not used	Not used	Not used
PARTNER – ADDRESS.	Required (O)		This area is for the partner address information
PARTNER–ADDRESS.ADDRLINE1-ADDRLINE9	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner address
PARTNER–ADDRESS.ADDRTYPE	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
PARTNER- ADDRESS.CITY	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner city
PARTNER- ADDRESS.COUNTRY	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner country
PARTNER- ADDRESS.COUNTY	Optional (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner county
PARTNER- ADDRESS.DESCRPTION	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
PARTNER- ADDRESS.FAX1-FAX 9	Optional (O)	HZ_CONTACT_POI NTS.RAW_PHONE_ NUMBER for the PARTY_ID, and CONTACT_POINT_T YPE = "FAX" for "Ship To" and HR_LOCATIONS_A LL.TELEPHONE_NU MBER_2 for "Ship From"	Partner fax numbers
PARTNER- ADDRESS.POSTALC ODE	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVE RIES table.	Partner postal code
PARTNER- ADDRESS.REGION	Optional (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVE RIES table.	Partner region
PARTNER- ADDRESS.STATEPR OVN	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVE RIES table.	Partner state or province

Field Name	Required?	Transaction Table.Column	Description
PARTNER- ADDRESS.TAXJRS DCTN	Not used	Not used	Not used
PARTNER- ADDRESS.TELEPHO NE1-TELEPHONE9	Required (O)	HZ_CONTACT_POI NTS.RAW_PHONE_ NUMBER for the PARTY_ID, and CONTACT_POINT_T YPE = "GEN" for " Ship To" and HZ_LOCATIONS_A LL.TELEPHONE_NU MBER_1 for "Ship From".	Partner telephone numbers
PARTNER- ADDRESS.URL	Optional (O)	HZ_CONTACT_POI NTS.URL for the PARTY_ID and CONTACT_POINT_T YPE = "GEN" for " Ship To"	Partner web site URL
PARTNER- ADDRESS.USERARE A	Not used	Not used	Not used
PARTNER – CONTACT.	Optional (O)		Partner contact information
PARTNER- CONTACT.NAME1	Optional (O)	HZ_PARTIES.PERSO N_FIRST_NAME + PERSON_MIDDLE_ NAME + PERSON_LAST_NA ME based on customer and ship to location on WSH_NEW_DELIVE RIES and based on SHIP_TO_CONTACT _ID on WSH_DELIVERY_DE TAILS.	Partner contact name

Field Name	Required?	Transaction Table.Column	Description
PARTNER-CONTACT.CONTCTTYPE	Not used	Not used	Not used
PARTNER-CONTACT.DESCRPTION	Not used	Not used	Not used
PARTNER-CONTACT.EMAIL	Not used	Not used	Not used
PARTNER-CONTACT.FAX1 - FAX9	Not used	Not used	Not used
PARTNER-CONTACT.NAME2 - NAME9	Not used	Not used	Not used
PARTNER-CONTACT.TELEPHONE1 -TELEPHONE9	Required (O)	HZ_CONTACT_POINTS.RAW_PHONE_NUMBER based on customer and ship to location on WSH_NEW_DELIVERIES and based on SHIP_TO_CONTACT_ID on WSH_DELIVERY_DETAILS.	Partner contact phone numbers
PARTNER-CONTACT.USERAREA	Not used	Not used	Not used
DOCUMNTREF.	Required (O)		This area contains the details of the document type

Field Name	Required?	Transaction Table.Column	Description
DOCUMNTREF.DOC TYPE	Required (O)	'SR' (SHIPMENT REQUEST FOR OUTBOUND) AND 'SA' (SHIMENT ADVICE FOR INBOUND) at the Supplier Instance.	The document type. SHIPMENT REQUEST
DOCUMNTREF.DOC UMENTID	Required (O)	WSH_TRANSACTION_ HISTORY.ORIG_ DOCUMENT_NUMB ER	The document ID
DOCUMNTREF.PAR TNRID	Required (O)	ORGANIZATION_C ODE based on WSH_NEW_DELIVE RIES.ORGANIZATIO N_ID	The partner ID
DOCUMNTREF.PAR TNRTYPE	Not used	Not used	Not used
DOCUMNTREF.DES CRIPTN	Not used	Not used	Not used
DOCUMNTREF.DOC UMENTRV	Not used	Not used	Not used
DOCUMNTREF.LIN ENUM	Not used	Not used	Not used
DOCUMNTREF.NOT ES1 – NOTES9	Not used	Not used	Not used
DOCUMNTREF.SCH LINENUM	Not used	Not used	Not used
DOCUMNTREF.SUB LINENUM	Not used	Not used	Not used
DOCUMNTREF.USE RAREA	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.	Required (O)		Represents the request to ship specific quantity of goods
SHIPITEM.AMOUNT (EXTENDED)	Not used	Not used	Not used
SHIPITEM.AMOUNT (ITEM)	Not used	Not used	Not used
SHIPITEM.COUNTRYORG	Required (O)	WSH_DELIVERY_DE TAILS.COUNTRY_OF_ORIGIN	Country of origin
SHIPITEM.DISPOSITION	Not used	Not used	Not used
SHIPITEM.EXPORTLIC	Not used	Not used	Not used
SHIPITEM.FRGHTCLASS	Optional (O)	MTL_CATEGORY_SETS_VL.CATEGORY_SET_NAME based on WSH_DELIVERY_DE TAILS.FREIGHT_CLASS_CAT_ID	Qualifies goods for purposes of freight cost calculation
SHIPITEM.FRGHTITEM	Not used	Not used	Not used
SHIPITEM.HAZRDMATL	Optional (O)	PO_HAZARD_CLASSES_VL . HAZARD_CLASS based on WSH_DELIVERY_DE TAILS.HAZARD_CLASS_ID	Contains a hazardous material code or description that can be a free form area or an agreed upon set of codes for the applications
SHIPITEM.IMPORTLIC	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.LINENUM	Not used	Not used	Not used
SHIPITEM.NOTES1 – NOTES9	Optional (O)	WSH_DELIVERY_DE TAILS.PACKING_IN STRUCTIONS	Item notes
SHIPITEM.OWNRSH PCDE	Optional (O)	WSH_DELIVERY_DE TAILS.FOB_CODE	Used to identify who or which organization owns the item or part. The content of this Field ID is user defined based on a specific Customer or Supplier
SHIPITEM.PACKING	Not used	Not used	Not used
SHIPITEM.PACKING DESC	Not used	Not used	Not used
SHIPITEM.QUANTITY (BACKORDERD)	Optional (I)	WSH_DELIVERY_DE TAILS.BACKORDERED_QUANTITY	Quantity of items backordered
SHIPITEM.QUANTITY (OPEN)	Not used	Not used	Not used
SHIPITEM.QUANTITY (SHIPPED)	Required (B)	WSH_DELIVERY_DE TAILS.SHIPPED_QUANTITY	This represents the actual quantity shipped
SHIPITEM.QUANTITY (VOLUME)	Optional (B)	WSH_DELIVERY_DE TAILS.VOLUME	The volume, or amount of space, that is occupied by an item, container or shipment
SHIPITEM.QUANTITY (WEIGHT)	Optional (B)	WSH_DELIVERY_DE TAILS.NET_WEIGHT	The weight of an item, container, or shipment

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.SHIPMAT LID	Not used	Not used	Not used
SHIPITEM.SHIPNOTES	Optional (B)	WSH_DELIVERY_DE TAILS.SHIPPING_IN STRUCTIONS	Notes area used for any additional information
SHIPITEM.SHIPPRIOR	Not used	Not used	Not used
SHIPITEM.SHIPMTL DESC	Not used	Not used	Not used
SHIPITEM.UPC	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.USERAREA	Optional (I)	WSH_DELIVERY_DE TAILS.DELIVERY_D ETAIL_ID	Ship item user area
ORACLE.DELIVERY _DETAIL_NUMBER		WSH_DELIVERY_AS SIGNMENTS.PAREN T_DELIVERY_DETAI L_ID	
ORACLE.PARENT_D ELIVERY_DETAIL_N UMBER		WSH_DELIVERY_DE TAILS.SHIP_TOLER ANCE_ABOVE	
ORACLE.SHIP_TOLE RANCE_ABOVE		WSH_DELIVERY_DE TAILS.SHIP_TOLER ANCE_BELOW	
ORACLE.SHIP_TOLE RANCE_BELOW		WSH_DELIVERY_DE TAILS.LOAD_SEQU ENCE_NUMBER	
ORACLE.LOAD_SEQ UENCE_NUMBER		WSH_DELIVERY_DE TAILS.SUBINVENTO RY	
ORACLE.SHIP_SET_ NUMBER		WSH_DELIVERY_DE TAILS descriptive flexfields	
ORACLE.SUBINVEN TORY		WSH_DELIVERY_DE TAILS.SHIP_SET_ID	
ORACLE.TOP_MOD EL_LINE_NUMBER		WSH_DELIVERY_DE TAILS.SUBINVENTO RY	
ORACLE.SHIP_MOD EL_COMPLETE_FLA G		WSH_DELIVERY_DE TAILS.TOP_MODEL _LINE_ID	
ORACLE.COMMODI TY_CODE_CATEGO RY		WSH_DELIVERY_DE TAILS.SHIP_MODEL _COMPLETE_FLAG	
ORACLE.SHIPMENT _PRIORITY_CODE		WSH_DELIVERY_DE TAILS.COMMODITY _CODE_CAT_ID converted to code	
ORACLE.DELIVER_ TO_LOCATION		WSH_DELIVERY_DE TAILS.SHIPMENT_P RORITY_CODE converted to "01", "02"	
ORACLE.CUSTOME R_NAME			
ORACLE.CUSTOME R_NUMBER			
ORACLE.TXN_SRC_ LINE_NUMBERQUA			

Field Name	Required?	Transaction Table.Column	Description
NTITY (TOTWEIGHT)		, or "03" for XML,	
ORACLE.MANUAL_ WV		WSH_DELIVERY_DE TAILS.DELIVER_TO _LOCATION_ID	
ORACLE.LINE_DIRE CTION		CONTACT segment sends CONTACT NAME and PHONE NUMBER of the contact based on the ship_to_contact_id on wsh_delivery_details.	
ORACLE.REQUEST_ DATE_TYPE_CODE			
DATETIME (ORACLE.EARLSTPI CK)			
DATETIME (ORACLE.LATESTPI CK)			
SHIPITEM.USERARE A (continued)			
DATETIME (ORACLE.EARLSTD ROP)			
DATETIME (ORACLE.LATESTD ROP)			
DATETIME (ORACLE.SCHEDUL ED)			
DATETIME (ORACLE.REQUEST ED)CONTACT Segment			
CHARGE.	Optional (B)		Represents additional freight or transportation charges associated with the SHPITEM

Field Name	Required?	Transaction Table.Column	Description
CHARGE.CHARGEID	Required (B)	WSH_FREIGHT_COST_TYPES.NAME based on WSH_FREIGHT_COST_TYPES.FREIGHT_COST_TYPE_ID converted to external value	Charge ID.
CHARGE.OPERAMT (EXTENDED) (T)	Required (B)	WSH_FREIGHT_COST_TYPES.UNIT_AMOUNT WSH_FREIGHT_COST_TYPES.CURRENCY_CODE	Extended operating cost amount
CHARGE.AMOUNT (EXTENDED) (T)	Not used	Not used	Not used
CHARGE.CHARGETYPE	Optional (B)	WSH_FREIGHT_COST_TYPES.NAME based on WSH_FREIGHT_COST_TYPES.FREIGHT_COST_TYPE_ID	Charge type
CHARGE.CHGLINE NUM	Not used	Not used	Not used
CHARGE.DESCRPTION	Not used	Not used	Not used
CHARGE.USERAREA	Optional (B)	WSH_FREIGHT_COST_TYPES.descriptive flexfields	User area
INVDETAIL.	Optional (B)		Inventory item attribute detail associated with the SHIPITEM
INVDETAIL.DISPOSITION	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
INVDETAIL.NOTES1 - NOTES9	Not used	Not used	Not used
DISTRIBUTION.	Not used	Not used	Not used
DISTRIBUTION.BUS NAREA	Not used	Not used	Not used
DISTRIBUTION.COSTCENTER	Not used	Not used	Not used
DISTRIBUTION.DEPARTMENT	Not used	Not used	Not used
DISTRIBUTION.DIVISION	Not used	Not used	Not used
DISTRIBUTION.ELEMENT1 - ELEMENT999	Not used	Not used	Not used
DISTRIBUTION.FUND	Not used	Not used	Not used
DISTRIBUTION.GEOGRAPHY	Not used	Not used	Not used
DISTRIBUTION.GLNTITYS	Not used	Not used	Not used
DISTRIBUTION.GLNOMACCT	Not used	Not used	Not used
DISTRIBUTION.OPERAMT (EXTENDED) (T)	Not used	Not used	Not used
DISTRIBUTION.PROFITCTR	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
DISTRIBUTION.PROJECT	Not used	Not used	Not used
DISTRIBUTION.UNIT	Not used	Not used	Not used
DISTRIBUTION.WAREHOUSE	Not used	Not used	Not used
DISTRIBUTION.USERAREA	Not used	Not used	Not used
SHIPUNIT.	Required (B)		Represents a single trackable shipping unit. A shipping unit is a uniquely identifiable assembly or container that holds one or more items to be shipped
SHIPUNIT.CARRIER	Required (B)	WSH_DELIVERY_DE TAILS.CARRIER_ID converted to external value	Carrier of the ship unit
SHIPUNIT.SHIPPERID	Required (I)	WSH_DELIVERY_DE TAILS.SHIP_FROM_ LOCATION_ID converted to external value.	The identifier of the partner who is responsible for packaging inventory goods for shipment
SHIPUNIT.TRACKINGID	Optional (B)	WSH_DELIVERY_DE TAILS.TRACKING_ NUMBER	A unique identifier for the purpose of tracking an individual package or shipment
SHIPUNIT.AMOUNT (DECLAREVAL)	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.AMOUNT (FREIGHT) (T)	Not used	Not used	Not used
SHIPUNIT.AMOUNT (ITEM) (T)	Not used	Not used	Not used
SHIPUNIT.AMOUNT (TOTAL) (T)	Not used	Not used	Not used
SHIPUNIT.CARRSERVICELEVEL	Optional (B)	WSH_NEW_DELIVERIES.SERVICE_LEVEL	Service level for the ship unit
SHIPUNIT.CONTAINERSEAL	Optional (B)	WSH_DELIVERY_DETAILS.SEAL_CODE	Seal number on the ship unit
SHIPUNIT.COUNTRYDST	Not used	Not used	Not used
SHIPUNIT.COUNTRYORG	Optional (B)	WSH_DELIVERY_DETAILS.COUNTRY_ORF_ORIGIN	Container country organization
SHIPUNIT.DATETIME (DELIVACT)	Optional (B)	WSH_TRIP_STOPS.ACTUAL_ARRIVAL_DATE	Actual delivery date and time
SHIPUNIT.DATETIME (DELIVSCHED)	Optional (I)	WSH_DELIVERY_DETAILS.DATE_SCHEDULED	Delivery schedule date and time
SHIPUNIT.DATETIME (LOADING)	Not used	Not used	Not used
SHIPUNIT.DATETIME (SHIP)	Not used	Not used	Not used
SHIPUNIT.DESCRPTION	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.EXPORTLIC	Not used	Not used	Not used
SHIPUNIT.FRGLTCLS	Optional (B)	MTL_CATEGORY_SETS_VL.CATEGORY_SET_NAME based on WSH_DELIVERY_DETAILS.FREIGHT_CLASS_ASS_CAT_ID converted to external value	Freight class of the ship unit
SHIPUNIT.FRGHTTERMS	Optional (B)	WSH_DELIVERY_DETAILS.FREIGHT_TERMS_CODE converted to external value	Ship unit freight terms
SHIPUNIT.HAZRDMATL	Optional (B)	WSH_DELIVERY_DETAILS.HAZARD_CLASS_ASS_ID converted to external value	Contains a hazardous material code or description that can be a free form area or an agreed upon set of codes for the applications involved in an integration scenario
SHIPUNIT.IMPORTLIC	Not used	Not used	Not used
SHIPUNIT.LOADPOINT	Not used	Not used	Not used
SHIPUNIT.NOTES1 - NOTES9	Not used	Not used	Not used
SHIPUNIT.OPERAMENT (FREIGHT) (T)	Not used	Not used	Not used
SHIPUNIT.PACKING	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.PACKINGDESC	Not used	Not used	Not used
SHIPUNIT.QUANTITY (ESTWEIGHT)	Not used	Not used	Not used
SHIPUNIT.QUANTITY (HEIGHT)	Optional (B)	Populate from MTL_SYSTEM_ITEMS	Height of the ship unit
SHIPUNIT.QUANTITY (LENGTH)	Optional (B)	Populate from MTL_SYSTEM_ITEMS	Length of the ship unit
SHIPUNIT.QUANTITY (LOADINGWT)	Optional (B)	Calculate WSH_DELIVERY_DETAILED.GROSS WEIGHT - WSH_DELIVERY_DETAILED.NET WEIGHT	Loading weight of the ship unit
SHIPUNIT.QUANTITY (NETWEIGHT)	Optional (B)	WSH_DELIVERY_DETAILED.NET_WEIGHT + WSH_DELIVERY_DETAILED.WEIGHT_UOM_CODE for delivery lines = container item	Net weight of the ship unit
SHIPUNIT.QUANTITY (TOTWEIGHT)	Required (B)	WSH_DELIVERY_DETAILED.GROSS_WEIGHT + WSH_DELIVERY_DETAILED.WEIGHT_UOM_CODE	Total weight of the ship unit

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.QUANTITY (VOLUME)	Optional (B)	WSH_DELIVERY_DE TAILS.VOLUME + WSH_DELIVERY_DE TAILS.VOLUME_UO M_CODE for delivery lines = container item	Volume of the ship unit
SHIPUNIT.QUANTITY (WIDTH)	Not used	Not used	Not used
SHIPUNIT.SHIPMAT LID	Optional (B)	WSH_DELIVERY_DE TAILS.INVENTORY_ ITEM_ID converted to Item Number by joining to mtl_system_items	The delivery detail id of the container record
SHIPUNIT.SHIPNOTES	Optional (B)	WSH_DELIVERY_DE TAILS.SHIPPING_IN STRUCTIONS	The shipping instructions or notes
SHIPUNIT.SHIPPOINT	Optional (B)	WSH_DELIVERY_DE TAILS.DELIVER_TO _LOCATION_ID	Identifies the location the goods are to be shipped
SHIPUNIT.SHIPSRV LVL	Optional (B)	WSH_DELIVERY_DE TAILS.SHIP_METHO D_CODE	A generic code that specifies a particular type or quality of carrier delivery service to be used for transporting goods
SHIPUNIT.SHPMTL DESC	Optional (B)	WSH_DELIVERY_DE TAILS.ITEM_DESCRI PTION	Describes the material or container a product or item is shipped in
SHIPUNIT.SHPUNIT SEQ	Not used	Not used	Not used
SHIPUNIT.SHPUNIT STS	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.SHPUNIT TOT	Not used	Not used	Not used
SHIPUNIT.SPECIAL HND	Not used	Not used	Not used
SHIPUNIT.STAGEPO INT	Not used	Not used	Not used
SHIPUNIT.TEMPRA TURE (ACTUAL)	Not used	Not used	Not used
SHIPUNIT.TEMPRA TURE (DELIVERY)	Not used	Not used	Not used
SHIPUNIT.TEMPRA TURE (LOADING)	Not used	Not used	Not used
SHIPUNIT.TRACKIN GIDTYPE	Not used	Not used	Not used
SHIPUNIT.TRANSM ETHD	Optional (B)	WSH_DELIVERY_DE TAILS.MODE_OF_T RANSPORT	Transportation method
SHIPUNIT.TRANSTE RMS	Optional (B)	WSH_NEW_DELIVE RIES.FOB_CODE	A contractual code used to describe the point where delivery occurs (goods ownership is transferred) in a contract of sale

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.USERARE A	Optional (I)	WSH_DELIVERY_DE TAILS.DELIVERY_D ETAIL_ID (To be called as DELIVERY_DETAIL_ NUMBER) WSH_DELIVERY_DE TAILS.PACKING_IN STRUCTIONS WSH_DELIVERY_DE TAILS.DESCRPTIVE _FLEXFIELDS WSH_DELIVERY_DE TAILS.LOT_NUMBE R WSH_DELIVERY_DE TAILS.SUBLOT_NU MBER WSH_DELIVERY_DE TAILS.REVISION WSH_DELIVERY_DE TAILS.SERIAL NUMBER WSH_DELIVERY_DE TAILS.TO_SERIAL NUMBER	User area for the ship unit
CONTAINER.	Optional (I)		Represents information about an intermediate packaging level within the shipping unit. A CONTAINER may or may not have INVITEM inventory associated with it

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.CONTAINRID	Optional (B)	WSH_DELIVERY_DE TAILS.CONTAINER_ NAME	Container ID
CONTAINER.CONTNRSEAL	Optional (B)	WSH_DELIVERY_DE TAILS.SEAL_CODE	Identifies the tamper-proof seal placed on a shipping container to prevent pilfering of the contents
CONTAINER.CONTRTYPE	Optional (B)	WSH_DELIVERY_DE TAILS.CONTAINER_ TYPE_CODE	A classification of containers
CONTAINER.DESCRIPTN	Not used	Not used	Not used
CONTAINER.NOTES 1 - NOTES 9	Not used	Not used	Not used
CONTAINER.PACKING	Not used	Not used	Not used
CONTAINER.PACKINGDESC	Not used	Not used	Not used
CONTAINER.PARTNTRID	Not used	Not used	Not used
CONTAINER.SHIPMENT ATLID	Optional (B)	WSH_DELIVERY_DE TAILS.INVENTORY_ ITEM_ID converted to ITEM NUMBER by joining to mtl_system_items	Container part number
CONTAINER.SHIPMENT TLDESC	Optional (B)	WSH_DELIVERY_DE TAILS.ITEM_DESCRIP TION	Container description – used most often for returnable containers

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.USER AREA	Optional (B)	WSH_DELIVERY_AS SIGNMENTS.DELIV ERY_DETAIL_ID (To be called as DELIVERY_DETAIL_ NUMBER) WSH_DELIVERY_AS SIGNMENTS.PAREN T_DELIVERY_DETAI L_ID (To be called as PARENT_DELIVERY _DETAIL_NUMBER) HEIGHT – From mtl_system_items LENGTH - From mtl_system_items WEIGHT- WSH_DELIVERY_DE TAILS. GROSS_WEIGHT WSH_DELIVERY_DE TAILS. NET_WEIGHT VOLUME – WSH_DELIVERY_DE TAILS. VOLUME WIDTH- From mtl_system_items LOT- WSH_DELIVERY_DE TAILS. LOT_NUMBER SUBLOT-WSH_DELI VERY_DETAILS. SUBLOT_NUMBER WSH_DELIVERY_DE	User area for additional information on the container

Field Name	Required?	Transaction Table.Column	Description
		TAILS. REVISION	
		WSH_DELIVERY_DE TAILS. SERIAL NUMBER	
		WSH_DELIVERY_DE TAILS. TO_SERIAL NUMBER	

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.USERA REA (continued)		WSH_DELIVERY_DE TAILS. SHIP_FROM_LOCAT ION_ID converted to code (To be called as SHIP_FROM_LOCAT ION LOCATION) WSH_DELIVERY_DE TAILS. DELIVER_TO_LOCA TION_ID (To be called as DELIVER_TO_LOCA TION) WSH_DELIVERY_DE TAILS. TRACKING_NUMBE R WSH_DELIVERY_DE TAILS. COUNTRY_OF_ORI GIN WSH_TRIP_STOPS.A CTUAL_ ARRIVAL_DATE WSH_DELIVERY_DE TAILS. DATE_SCHEDULED WSH_TRIP_STOPS.A CTUAL_ DEPARTURE_DATE WSH_DELIVERY_DE TAILS. SHIP_METHOD_CO DE WSH_DELIVERY_DE TAILS. FREIGHT_CLASS_C	

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.USERA REA (continued)		AT_ID converted to external value	
		(To be called as FREIGHT_CLASS_C ATEGORY)	
		WSH_DELIVERY_DE TAILS. FREIGHT_TERMS_C ODE converted to external value	
		WSH_DELIVERY_DE TAILS. HAZARD_CLASS_ID converted to external value	
		(To be called as HAZARD_CLASS or as HAZARD_CLASS_C ODE)	
		WSH_DELIVERY_DE TAILS. FOB_CODE	
		WSH_DELIVERY_DE TAILS. SHIPPING_INSTRUC TIONS	
		WSH_DELIVERY_DE TAILS. PACKING_INSTRUC TIONS	
		WSH_DELIVERY_DE TAILS. Descriptive Flexfields	

Field Name	Required?	Transaction Table.Column	Description
CHARGE.	Required (I)		This CHARGE section is for a CONTAINER. Represents additional freight or transportation charges associated with the shipment
CHARGE.CHARGEID	Required (I)	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	Charge ID
CHARGE.OPERAMT (EXTENDED) (T)	Required (I)	WSH_FREIGHT_COSTS.TOTAL_AMOUNT WSH_FREIGHT_COSTS.CURRENCY_CODE	Operating amount
CHARGE.AMOUNT (EXTENDED) (T)	Required (I)		Extended charge amount
CHARGE.CHARGETYPE	Required (I)	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	Charge type
CHARGE.CHGLINE NUM	Required (I)		Line number of the charge
CHARGE.DESCRPTION	Required (I)		Description of the charge
CHARGE.USERARE A	Optional (I)	WSH_FREIGHT_COSTS descriptive flexfields	

Field Name	Required?	Transaction Table.Column	Description
INVITEM.	Required (B)		Represents the occurrence of a specific quantity of goods inventory packed within the SHIPUNIT or CONTAINER. The inventory item occurrence typically refers to a line item or shipping schedule line of a sales order, purchase order or other business document
INVITEM.ITEM	Required (B)	MTL_SYSTEM_ITEM S_KFV. CONCATENATED_S EGMENTS	Item number
INVITEM.QUANTIT Y (ITEM)	Not used	Not used	Not used
INVITEM.AMOUNT (EXTENDED) (T)	Not used	Not used	Not used
INVITEM.AMOUNT (ITEM) (T)	Not used	Not used	Not used
INVITEM.FRGTCL S	Not used	Not used	Not used
INVITEM.FRGTITE M	Not used	Not used	Not used
INVITEM.HAZRDM ATL	Not used	Not used	Not used
INVITEM.ITEMDESC	Not used	Not used	Not used
INVITEM.ITEMRV	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
INVITEM.ITEMRVX	Not used	Not used	Not used
INVITEM.LOTLEVEL 1 - LOTLEVEL2	Not used	Not used	Not used
INVITEM.NOTES 1 - NOTES 9	Not used	Not used	Not used
INVITEM.OWNRSH PCDE	Not used	Not used	Not used
INVITEM.QUANTIT Y (VOLUME)	Not used	Not used	Not used
INVITEM.QUANTIT Y (WEIGHT)	Not used	Not used	Not used
INVITEM.SHIPNOTE S	Not used	Not used	Not used
INVITEM.UPC	Not used	Not used	Not used
INVITEM.USERARE A	Not used	Not used	Not used
CHARGE.	Required (I)	This CHARGE section is for SHIPUNIT.	Represents additional freight or transportation charges associated with the shipment
CHARGE.CHARGEI D	Required (I)	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID	Charge ID
CHARGE.OPERAMT (EXTENDED) (T)	Required (I)	WSH_FREIGHT_COSTS.TOTAL_AMOUNT WSH_FREIGHT_COSTS.CURRENCY_CODE	Charge total amount

Field Name	Required?	Transaction Table.Column	Description
CHARGE.AMOUNT (EXTENDED) (T)	Required (I)		Extended charge amount
CHARGE.CHARGET YPE	Required (I)	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	Charge type
CHARGE.CHGLINE NUM	Required (I)		Charge line number
CHARGE.DESCRIP T	Required (I)		Description of the charge

The following represents the SHOW_SHIPMENT REQUEST XML data for carrier manifesting in Oracle Shipping Execution.

```

<?xml version = '1.0' encoding = 'UTF-8' standalone = 'no'?>
<!-- Oracle eXtensible Markup Language Gateway Server -->
<!DOCTYPE SHOW_SHIPMENT_005 SYSTEM "161_show_shipment_005.dtd">
<SHOW_SHIPMENT_005>
  <CNTROLAREA>
    <BSR>
      <VERB value="SHOW">SHOW</VERB>
      <NOUN value="SHIPMENT">SHIPMENT</NOUN>
      <REVISION value="005">005</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>WSH</COMPONENT>
      <TASK>SSNO</TASK>

    <REFERENCEID>P11510C2.US.ORACLE.COM:oracle.apps.wsh.sup.ssro:TPW346</REFERENCEID>
      <CONFIRMATION>0</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2005</YEAR>
      <MONTH>04</MONTH>
      <DAY>15</DAY>
      <HOUR>09</HOUR>
      <MINUTE>44</MINUTE>
      <SECOND>41</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SHOW_SHIPMENT>
      <SHIPMENT>
        <DATETIME qualifier="DOCUMENT">
          <YEAR>2005</YEAR>
          <MONTH>04</MONTH>
          <DAY>15</DAY>
          <HOUR>09</HOUR>
          <MINUTE>44</MINUTE>
          <SECOND>41</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+0000</TIMEZONE>
        </DATETIME>
        <AMOUNT qualifier="DECLAREVAL">
          <VALUE/>
          <NUMOFDEC/>
          <SIGN/>
          <CURRENCY/>
          <DRCR/>
        </AMOUNT>
        <AMOUNT qualifier="DOCUMENT" type="T">
          <VALUE/>
          <NUMOFDEC/>
          <SIGN/>
          <CURRENCY/>
          <DRCR/>
        </AMOUNT>
        <AMOUNT qualifier="ESTFREIGHT">

```

```

<VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<DATETIME qualifier="CREATION">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>09</HOUR>
    <MINUTE>43</MINUTE>
    <SECOND>57</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="DELIVACT">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="DELIVSCHED">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>23</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>00</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="EARLSTSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="LOADING">

```

```

<YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="NEEDELV">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>23</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>00</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="PROMDELV">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="PROMSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIPSCHEDE">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>23</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>00</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<OPERAMT qualifier="FREIGHT" type="T">
    <VALUE/>

```

```

<NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <UOMVALUE/>
    <UOMNUMDEC/>
    <UOM/>
</OPERAMT>
<QUANTITY qualifier="ESTWEIGHT">
    <VALUE>180</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="LOADINGWT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="NETWEIGHT">
    <VALUE>180</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPUNIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="TOTWEIGHT">
    <VALUE>180</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
    <VALUE>12</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>FT3</UOM>
</QUANTITY>
<TEMPRATURE qualifier="ACTUAL" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="DELIVERY" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="LOADING" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<DOCUMENTID>340</DOCUMENTID>
<SHIPPERID>V1- New York City</SHIPPERID>
<SYNCID>A</SYNCID>

```



```

<CARRIER/>
  <CARRSRVLVL/>
  <COUNTRYDST>US</COUNTRYDST>
  <COUNTRYORG/>
  <DESCRIPTN/>
  <DISTCENTER/>
  <DOCKID/>
  <DOCTYPE>SR</DOCTYPE>
  <DOCUMENTRV/>
  <DROPSHIP/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTTERMS/>
  <HAZRDMATL/>
  <HDRSTATUS/>
  <IMPORTLIC/>
  <LOADPOINT/>
  <NOTES index="1"/>
  <PARTIALSHP/>
  <PRIORITY/>
  <ROUTEID/>
  <SHIPNOTES/>
  <SHIPPOINT>Chattanooga (OPS)</SHIPPOINT>
  <SHIPPRIOR/>
  <SHIPSRLVLVL/>
  <SPECIALHND/>
  <STAGEPOINT/>
  <TRANSMETHD/>
  <TRANSTERMS/>
  <WAREHOUSE>V1</WAREHOUSE>
  <USERAREA/>
  <PARTNER>
    <NAME index="1">Phoenix Manufacturing</NAME>
    <ONETIME/>
    <PARTNRID>V1- New York City</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER>0</DUNSNUMBER>
    <GLENTITYS/>
    <NAME index="2"/>
    <PARENTID/>
    <PARTNRIDX/>
    <PARTNRRATG/>
    <PARTNRROLE/>
    <PAYMETHOD/>
    <TAXEXEMPT/>
    <TAXID/>
    <TERMID/>
    <USERAREA/>
    <ADDRESS>
      <ADDRLINE index="1">3200 East Camelback</ADDRLINE>
      <ADDRLINE index="2">Suite 255</ADDRLINE>
      <ADDRLINE index="3"/>
      <ADDRLINE index="4"/>
      <ADDRTYPE/>
      <CITY>Phoenix</CITY>
      <COUNTRY>US</COUNTRY>
      <COUNTY/>
    
```

```

<DESCRIPTN/>
    <FAX index="1">602-224-6250</FAX>
    <POSTALCODE>85018</POSTALCODE>
    <REGION/>
    <STATEPROVN>AZ</STATEPROVN>
    <TAXJRSDCTN/>
    <TELEPHONE index="1">602-224-6200</TELEPHONE>
    <URL/>
    <USERAREA/>
</ADDRESS>
<ATTCHREF>
    <DATETIME qualifier="CREATION">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <CMPSRNTYPE/>
    <CMPSRNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
</ATTCHREF>
<CONTACT>
    <NAME index="1"/>
    <CONTCTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>
</CONTACT>
</PARTNER>
<PARTNER>
    <NAME index="1">Computer Service and Rentals</NAME>
    <ONETIME/>
    <PARTNRID/>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER/>
    <GLENTITYS/>
    <NAME index="2"/>
    <PARENTID/>
    <PARTNRIDX/>

```

```

<PARTNRRATG/>
  <PARTNRROLE/>
  <PAYMETHOD/>
  <TAXEXEMPT/>
  <TAXID/>
  <TERMINID/>
  <USERAREA/>
  <ADDRESS>
    <ADDRLINE index="1">301 Summit Hill Drive</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRTYPE/>
    <CITY>Chattanooga</CITY>
    <COUNTRY>US</COUNTRY>
    <COUNTY>Hamilton</COUNTY>
    <DESCRIPTN/>
    <FAX index="1"/>
    <POSTALCODE>37401</POSTALCODE>
    <REGION/>
    <STATEPROVN>TN</STATEPROVN>
    <TAXJRS DCTN/>
    <TELEPHONE index="1">615-522-3600</TELEPHONE>
    <URL/>
    <USERAREA/>
  </ADDRESS>
  <ATTCHREF>
    <DATETIME qualifier="CREATION">
      <YEAR/>
      <MONTH/>
      <DAY/>
      <HOUR/>
      <MINUTE/>
      <SECOND/>
      <SUBSECOND/>
      <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <CMPSNTYPE/>
    <CMPSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
  </ATTCHREF>
  <CONTACT>
    <NAME index="1">Gerry Brown</NAME>
    <CONTACTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1">+1 (703) 8441212</TELEPHONE>
    <USERAREA/>
  </CONTACT>

```

```

</CONTACT>
  </PARTNER>
  <SHIPITEM>
    <QUANTITY qualifier="ITEM">
      <VALUE>1</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>Ea</UOM>
    </QUANTITY>
    <AMOUNT qualifier="EXTENDED" type="T">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <CURRENCY/>
      <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="ITEM" type="T">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <CURRENCY/>
      <DRCR/>
    </AMOUNT>
    <QUANTITY qualifier="BACKORDERD">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="OPEN">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="ORDERED">
      <VALUE>1</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="SHIPPED">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="VOLUME">
      <VALUE>2</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>FT3</UOM>
    </QUANTITY>
    <QUANTITY qualifier="WEIGHT">
      <VALUE>30</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>Lbs</UOM>
    </QUANTITY>
    <ITEM>AS54888</ITEM>
    <COUNTRYORG/>
  </SHIPITEM>
</CONTACT>

```

```

<DISPOSITN/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTITEM/>
  <HAZRDMATL/>
  <IMPORTLIC/>
  <ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
  <ITEMRV/>
  <ITEMRVX/>
  <ITEMX/>
  <LOTLEVEL index="1"/>
  <NOTES index="1"/>
  <OWNRSHPCDE>Destination</OWNRSHPCDE>
  <PACKING/>
  <PARTIALSHP>0</PARTIALSHP>
  <SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>

```

```

</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <CURRENCY/>
  <DRCR/>
</AMOUNT>
<QUANTITY qualifier="BACKORDERD">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="OPEN">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="ORDERED">
  <VALUE>1</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Ea</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPPED">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>2</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
  <VALUE>30</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>
<PARTIALSHP>0</PARTIALSHP>

```

```

<SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <QUANTITY qualifier="BACKORDERD">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="OPEN">

```

```

<VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="ORDERED">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPPED">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
    <VALUE>2</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
    <VALUE>30</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>
<PARTIALSHP>0</PARTIALSHP>
<SHIPNOTES/>
<SHIPPRIOR/>
<UPC/>
<USERAREA/>
<DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>

```



```

<USERAREA/>
</DOCUMNTREF>
<INVDETAIL>
  <QUANTITY qualifier="ITEM">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
  <DISPOSITN/>
  <LOTLEVEL index="1"/>
  <LOTLEVEL index="2"/>
  <NOTES index="1"/>
  <SERIALNUM/>
  <USERAREA/>
</INVDETAIL>
</SHIPITEM>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <QUANTITY qualifier="BACKORDERD">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="OPEN">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="ORDERED">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <QUANTITY qualifier="SHIPPED">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>

```

<UOM/>

```
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>2</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
  <VALUE>30</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>
<PARTIALSHP>0</PARTIALSHP>
<SHIPNOTES/>
<SHIPPRIOR/>
<UPC/>
<USERAREA/>
<DOCUMNTREF>
  <DOCTYPE>PURCHASEORDER</DOCTYPE>
  <DOCUMENTID/>
  <PARTNRID>Chattanooga (OPS)</PARTNRID>
  <PARTNRTYPE>ShipTo</PARTNRTYPE>
  <DESCRIPTN/>
  <DOCUMENTRV/>
  <LINENUM/>
  <NOTES index="1"/>
  <SCHLINENUM/>
  <SUBLINENUM/>
  <USERAREA/>
</DOCUMNTREF>
<INVDDETAIL>
  <QUANTITY qualifier="ITEM">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
  <DISPOSITN/>
  <LOTLEVEL index="1"/>
  <LOTLEVEL index="2"/>
  <NOTES index="1"/>
  <SERIALNUM/>
```

```

<USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <QUANTITY qualifier="BACKORDERD">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="OPEN">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="ORDERED">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <QUANTITY qualifier="SHIPPED">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="VOLUME">
    <VALUE>2</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>FT3</UOM>
  </QUANTITY>
  <QUANTITY qualifier="WEIGHT">
    <VALUE>30</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
  </QUANTITY>
</ITEM>AS54888</ITEM>

```

```

<COUNTRYORG/>
  <DISPOSITN/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTITEM/>
  <HAZRDMATL/>
  <IMPORTLIC/>
  <ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
  <ITEMRV/>
  <ITEMRVX/>
  <ITEMX/>
  <LOTLEVEL index="1"/>
  <NOTES index="1"/>
  <OWNRSHPCDE>Destination</OWNRSHPCDE>
  <PACKING/>
  <PARTIALSHP>0</PARTIALSHP>
  <SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</COUNTRYORG>

```

```

</INVDETAIL>
  </SHIPITEM>
  <SHIPITEM>
    <QUANTITY qualifier="ITEM">
      <VALUE>1</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>Ea</UOM>
    </QUANTITY>
    <AMOUNT qualifier="EXTENDED" type="T">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <CURRENCY/>
      <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="ITEM" type="T">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <CURRENCY/>
      <DRCR/>
    </AMOUNT>
    <QUANTITY qualifier="BACKORDERD">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="OPEN">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="ORDERED">
      <VALUE>1</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="SHIPPED">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="VOLUME">
      <VALUE>2</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>FT3</UOM>
    </QUANTITY>
    <QUANTITY qualifier="WEIGHT">
      <VALUE>30</VALUE>
      <NUMOFDEC/>
      <SIGN>+</SIGN>
      <UOM>Lbs</UOM>
    </QUANTITY>
    <ITEM>AS54888</ITEM>
    <COUNTRYORG/>
  </SHIPITEM>

```

```

<DISPOSITN/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTITEM/>
  <HAZRDMATL/>
  <IMPORTLIC/>
  <ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
  <ITEMRV/>
  <ITEMRVX/>
  <ITEMX/>
  <LOTLEVEL index="1"/>
  <NOTES index="1"/>
  <OWNRSHPCDE>Destination</OWNRSHPCDE>
  <PACKING/>
  <PARTIALSHP>0</PARTIALSHP>
  <SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<ATTCHREF>
  <DATETIME qualifier="CREATION">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
  </DATETIME>
  <QUANTITY qualifier="FILESIZE">
    <VALUE/>

```

```

<NUMOFDEC/>
    <SIGN/>
    <UOM/>
    </QUANTITY>
    <CMPRSNTYPE/>
    <CMPRSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
</ATTCHREF>
<DOCUMNTREF>
    <DOCTYPE>SR</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>V1</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
</DOCUMNTREF>
</SHIPMENT>
</SHOW_SHIPMENT>
</DATAAREA>
</SHOW_SHIPMENT_005>

```

Carrier Manifesting INBOUND Sample XML

The following represents the SHOW_SHIPMENT ADVICE XML data for carrier manifesting in Oracle Shipping Execution.

```

<?xml version = '1.0' encoding = 'UTF-8' standalone = 'no'?>
<!-- Oracle eXtensible Markup Language Gateway Server -->
<!DOCTYPE SHOW_SHIPMENT_005 SYSTEM "161_show_shipment_005.dtd">
<SHOW_SHIPMENT_005>
  <CNTRLAREA>
    <BSR>
      <VERB value="SHOW">SHOW</VERB>
      <NOUN value="SHIPMENT">SHIPMENT</NOUN>
      <REVISION value="005">005</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID/>
      <COMPONENT>WSH</COMPONENT>
      <TASK>SSNO</TASK>

    <REFERENCEID>HM000A:oracle.apps.wsh.sup.ssro:CMS437</REFERENCEID>
      <CONFIRMATION>0</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2003</YEAR>
      <MONTH>01</MONTH>
      <DAY>15</DAY>
      <HOUR>15</HOUR>
      <MINUTE>18</MINUTE>
      <SECOND>41</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATETIME>
  </CNTRLAREA>
  <DATAAREA>
    <SHOW_SHIPMENT>
      <SHIPMENT>
        <DATETIME qualifier="DOCUMENT">
          <YEAR>2003</YEAR>
          <MONTH>01</MONTH>
          <DAY>15</DAY>
          <HOUR>15</HOUR>
          <MINUTE>18</MINUTE>
          <SECOND>40</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+0000</TIMEZONE>
        </DATETIME>
        <AMOUNT qualifier="DECLAREVAL">
          <VALUE/>
          <NUMOFDEC/>
          <SIGN/>
          <CURRENCY/>
          <DRCR/>
        </AMOUNT>
        <AMOUNT qualifier="DOCUMENT" type="T">
          <VALUE/>
          <NUMOFDEC/>
          <SIGN/>
          <CURRENCY/>
          <DRCR/>
        </AMOUNT>
        <AMOUNT qualifier="ESTFREIGHT">
          <VALUE/>

```



```

<NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<DATETIME qualifier="CREATION">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>54</MINUTE>
    <SECOND>57</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="DELIVACT">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="DELIVSCHED">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>10</MINUTE>
    <SECOND>17</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="EARLSTSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="LOADING">
    <YEAR/>

```

```

<MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="NEEDDELV">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>10</MINUTE>
    <SECOND>17</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="PROMDELV">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="PROMSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIPSCHEDE">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>10</MINUTE>
    <SECOND>17</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<OPERAMT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>

```

```

<SIGN/>
    <CURRENCY/>
    <UOMVALUE/>
    <UOMNUMDEC/>
    <UOM/>
</OPERAMT>
<QUANTITY qualifier="ESTWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="LOADINGWT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="NETWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="SHIPUNIT">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="TOTWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<TEMPRATURE qualifier="ACTUAL" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="DELIVERY" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="LOADING" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<DOCUMENTID>367-I</DOCUMENTID>
<SHIPPERID>M3- Dallas</SHIPPERID>
<SYNCID>A</SYNCID>
<CARRIER>WSH-DHL</CARRIER>

```

```

<CARRSRVLVL>1DM</CARRSRVLVL>
  <COUNTRYDST>US</COUNTRYDST>
  <COUNTRYORG/>
  <DESCRIPTN/>
  <DISTCENTER/>
  <DOCKID/>
  <DOCTYPE>SA</DOCTYPE>
  <DOCUMENTRV/>
  <DROPSHIP/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTTERMS>Paid</FRGHTTERMS>
  <HAZRDMATL/>
  <HDRSTATUS/>
  <IMPORTLIC/>
  <LOADPOINT/>
  <NOTES index="1"/>
  <PARTIALSHP/>
  <PRIORITY/>
  <ROUTEID/>
  <SHIPNOTES/>
  <SHIPPOINT>Chattanooga (OPS)</SHIPPOINT>
  <SHIPPRIOR/>
  <SHIPSRLVL>1DM</SHIPSRLVL>
  <SPECIALHND/>
  <STAGEPOINT/>
  <TRANSMETHD>PARCEL</TRANSMETHD>
  <TRANSTERMS>Destination</TRANSTERMS>
  <WAREHOUSE>M3</WAREHOUSE>
  <USERAREA/>
  <PARTNER>
    <NAME index="1">Dallas Manufacturing</NAME>
    <ONETIME/>
    <PARTNRID>M3- Dallas</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER>0</DUNSNUMBER>
    <GLENTITYS/>
    <NAME index="2"/>
    <PARENTID/>
    <PARTNRIDX/>
    <PARTNRRATG/>
    <PARTNRROLE/>
    <PAYMETHOD/>
    <TAXEXEMPT/>
    <TAXID/>
    <TERMID/>
    <USERAREA/>
    <ADDRESS>
      <ADDRLINE index="1">222 West Las Colinas
Bldv</ADDRLINE>
      <ADDRLINE index="2"/>
      <ADDRLINE index="3"/>
      <ADDRLINE index="4"/>
      <ADDRTYPE/>
      <CITY>Irving</CITY>
      <COUNTRY>US</COUNTRY>
      <COUNTY/>

```

```

<DESCRIPTN/>
    <FAX index="1"/>
    <POSTALCODE>75039</POSTALCODE>
    <REGION/>
    <STATEPROVN/>
    <TAXJRSDCTN/>
    <TELEPHONE index="1">972-868-4457</TELEPHONE>
    <URL/>
    <USERAREA/>
</ADDRESS>
<ATTCHREF>
    <DATETIME qualifier="CREATION">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <CMPSNSTYPE/>
    <CMPSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
</ATTCHREF>
<CONTACT>
    <NAME index="1"/>
    <CONTCTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>
</CONTACT>
</PARTNER>
<PARTNER>
    <NAME index="1">Computer Service and Rentals</NAME>
    <ONETIME/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER/>
    <GENTITYS/>
    <NAME index="2"/>
    <PARENTID/>
    <PARTNRIDX/>

```

```

<PARTNRRATG/>
  <PARTNRROLE/>
  <PAYMETHOD/>
  <TAXEXEMPT/>
  <TAXID/>
  <TERMID/>
  <USERAREA/>
  <ADDRESS>
    <ADDRLINE index="1">301 Summit Hill Drive</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRTYPE/>
    <CITY>Chattanooga</CITY>
    <COUNTRY>US</COUNTRY>
    <COUNTY>Hamilton</COUNTY>
    <DESCRIPTN/>
    <FAX index="1"/>
    <POSTALCODE>37401</POSTALCODE>
    <REGION/>
    <STATEPROVN>TN</STATEPROVN>
    <TAXJRS DCTN/>
    <TELEPHONE index="1">615-522-3600</TELEPHONE>
    <URL/>
    <USERAREA/>
  </ADDRESS>
  <ATTCHREF>
    <DATETIME qualifier="CREATION">
      <YEAR/>
      <MONTH/>
      <DAY/>
      <HOUR/>
      <MINUTE/>
      <SECOND/>
      <SUBSECOND/>
      <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <CMPSNTYPE/>
    <CMPSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
  </ATTCHREF>
  <CONTACT>
    <NAME index="1"/>
    <CONTACTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>

```

```

</CONTACT>
</PARTNER>
<ATTCHREF>
  <DATETIME qualifier="CREATION">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
  </DATETIME>
  <QUANTITY qualifier="FILESIZE">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <CMPRSNTYPE/>
  <CMPRSNID/>
  <DESCRIPTN/>
  <FILETYPE/>
  <NOTES index="1"/>
  <TITLE/>
  <FILENAME/>
  <USERAREA/>
</ATTCHREF>
<CHARGE>
  <OPERAMT qualifier="EXTENDED" type="T">
    <VALUE>75</VALUE>
    <NUMOFDEC>1</NUMOFDEC>
    <SIGN>+</SIGN>
    <CURRENCY>USD</CURRENCY>
    <UOMVALUE>D</UOMVALUE>
    <UOMNUMDEC/>
    <UOM/>
  </OPERAMT>
  <CHARGEID>Handling</CHARGEID>
  <CHARGETYPE>Handling</CHARGETYPE>
  <CHGLINENUM/>
  <DESCRIPTN/>
  <USERAREA/>
  <DISTRIBUTN>
    <OPERAMT qualifier="EXTENDED" type="T">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <CURRENCY/>
      <UOMVALUE/>
      <UOMNUMDEC/>
      <UOM/>
    </OPERAMT>
    <BUSNAREA/>
    <COSTCENTER/>
    <DEPARTMENT/>
    <DIVISION/>
    <ELEMENT index="1"/>
    <FUND/>
    <GEOGRAPHY/>
    <GLENTITYS/>
  </DISTRIBUTN>
</CHARGE>

```

```

<GLNOMACCT/>
    <PROFITCTR/>
    <PROJECT/>
    <UNIT/>
    <WAREHOUSE/>
    <USERAREA/>
  </DISTRIBUTN>
</CHARGE>
<DOCUMNTREF>
  <DOCTYPE>SR</DOCTYPE>
  <DOCUMENTID/>
  <PARTNRID>M3</PARTNRID>
  <PARTNRTYPE>ShipFrom</PARTNRTYPE>
  <DESCRIPTN/>
  <DOCUMENTRV/>
  <LINENUM/>
  <NOTES index="1"/>
  <SCHLINENUM/>
  <SUBLINENUM/>
  <USERAREA/>
</DOCUMNTREF>
</SHIPMENT>
<SHIPUNIT>
  <AMOUNT qualifier="DECLAREVAL">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="TOTAL" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <DATETIME qualifier="DELIVACT">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
  </DATETIME>

```



```

<DATETIME qualifier="DELIVSCHED">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="LOADING">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<OPERAMT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <UOMVALUE/>
    <UOMNUMDEC/>
    <UOM/>
</OPERAMT>
<QUANTITY qualifier="ESTWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="HEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="LENGTH">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="LOADINGWT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>

```

<UOM/>

```
</QUANTITY>
<QUANTITY qualifier="NETWEIGHT">
  <VALUE>1</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="TOTWEIGHT">
  <VALUE>1001</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>1</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WIDTH">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<TEMPERATURE qualifier="ACTUAL" type="FAHRENHEIT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
</TEMPERATURE>
<TEMPERATURE qualifier="DELIVERY" type="FAHRENHEIT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
</TEMPERATURE>
<TEMPERATURE qualifier="LOADING" type="FAHRENHEIT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
</TEMPERATURE>
<CARRIER>WSH-DHL</CARRIER>
<SHIPPERID>M3- Dallas</SHIPPERID>
<TRACKINGID>1Z4444441500000227</TRACKINGID>
<CARRSRVLVL>1DM</CARRSRVLVL>
<CONTAINRID>118081</CONTAINRID>
<CONTNRSEAL/>
<CONTNRTYPE>ALC</CONTNRTYPE>
<COUNTRYDST/>
<COUNTRYORG/>
<DESCRIPTN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<FRGHTTERMS>Paid</FRGHTTERMS>
<HAZRDMATL/>
<IMPORTLIC/>
<LOADPOINT/>
<NOTES index="1"/>
<SHIPMATLID>A1 DOM</SHIPMATLID>
<SHIPNOTES/>
```

```

<SHIPPOINT>Chattanooga (OPS)</SHIPPOINT>
  <SHIPSRVLVL>1DM</SHIPSRVLVL>
  <SHPMTLDESC>A1 Domestic Airline Container</SHPMTLDESC>
  <SHPUNITSEQ/>
  <SHPUNITSTS/>
  <SHPUNITTOT/>
  <SPECIALHND/>
  <STAGEPOINT/>
  <TRANSMETHD>PARCEL</TRANSMETHD>
  <TRANSTERMS>Destination</TRANSTERMS>
  <USERAREA/>
  <PARTNER>
    <NAME index="1">Computer Service and Rentals</NAME>
    <ONETIME/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRRTYPE>ShipTo</PARTNRRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER/>
    <GLENTITYS/>
    <NAME index="1"/>
    <PARENTID/>
    <PARTNRIDX/>
    <PARTNRRATG/>
    <PARTNRROLE/>
    <PAYMETHOD/>
    <TAXEXEMPT/>
    <TAXID/>
    <TERMID/>
    <USERAREA/>
    <ADDRESS>
      <ADDRLINE index="1">301 Summit Hill Drive</ADDRLINE>
      <ADDRTYPE/>
      <CITY>Chattanooga</CITY>
      <COUNTRY>US</COUNTRY>
      <COUNTY/>
      <DESCRIPTN/>
      <FAX index="1"/>
      <POSTALCODE>37401</POSTALCODE>
      <REGION/>
      <STATEPROVN>TN</STATEPROVN>
      <TAXJRSRCTN/>
      <TELEPHONE index="1">615-522-3600</TELEPHONE>
      <URL/>
      <USERAREA/>
    </ADDRESS>
    <ATTCHREF>
      <DATETIME qualifier="CREATION">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
      </DATETIME>
      <QUANTITY qualifier="FILESIZE">
        <VALUE/>
      </QUANTITY>
    </ATTCHREF>
  </PARTNER>
</SHIPPOINT>

```

```

<NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <CMPRSNTYPE/>
  <CMPRSNID/>
  <DESCRIPTN/>
  <FILETYPE/>
  <NOTES index="1"/>
  <TITLE/>
  <FILENAME/>
  <USERAREA/>
</ATTCHREF>
<CONTACT>
  <NAME index="1"/>
  <CONTCTTYPE/>
  <DESCRIPTN/>
  <EMAIL/>
  <FAX index="1"/>
  <NAME index="1"/>
  <TELEPHONE index="1"/>
  <USERAREA/>
</CONTACT>
</PARTNER>
<ATTCHREF>
  <DATETIME qualifier="CREATION">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
  </DATETIME>
  <QUANTITY qualifier="FILESIZE">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <CMPRSNTYPE/>
  <CMPRSNID/>
  <DESCRIPTN/>
  <FILETYPE/>
  <NOTES index="1"/>
  <TITLE/>
  <FILENAME/>
  <USERAREA/>
</ATTCHREF>
<DOCUMNTREF>
  <DOCTYPE/>
  <DOCUMENTID/>
  <PARTNRID/>
  <PARTNRTYPE/>
  <DESCRIPTN/>
  <DOCUMENTRV/>
  <LINENUM/>
  <NOTES index="1"/>
  <SCHLINENUM/>
  <SUBLINENUM/>

```

```

<USERAREA/>
</DOCUMNTREF>
<INVITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <QUANTITY qualifier="VOLUME">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="WEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <ITEM/>
  <COUNTRYORG/>
  <FRGHTCLS/>
  <FRGHTITEM/>
  <HAZRDMATL/>
  <ITEMDESC/>
  <ITEMRV/>
  <ITEMRVX/>
  <ITEMX/>
  <LOTLEVEL index="1"/>
  <NOTES index="1"/>
  <OWNRSHPCDE/>
  <SHIPNOTES/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE/>
    <DOCUMENTID/>
    <PARTNRID/>
    <PARTNRTYPE/>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
  </DOCUMNTREF>
</INVITEM>

```

```

<USERAREA/>
    </DOCUMNTREF>
    <INVDDETAIL>
        <QUANTITY qualifier="ITEM">
            <VALUE/>
            <NUMOFDEC/>
            <SIGN/>
            <UOM/>
        </QUANTITY>
        <DESCRIPTN/>
        <DISPOSITN/>
        <LOTLEVEL index="1"/>
        <NOTES index="1"/>
        <SERIALNUM/>
        <USERAREA/>
    </INVDDETAIL>
</INVITEM>
</SHIPUNIT>
</SHOW_SHIPMENT>
</DATAAREA>
</SHOW_SHIPMENT_005>

```

Using XML Messaging with Third Party Warehousing

Third party warehouse transactions support communications between you and a separate distribution warehouse. Oracle Shipping Execution enables you to send an outbound order to the separate warehouse indicating what the warehouse needs to ship inventory on your behalf. After the third party warehouse has completed the shipment, the details of the shipment are sent back to Oracle Shipping Execution.

These transactions contain shipment details to determine what to ship, where to ship, and to print the required labels and paperwork. Once the third party warehouse responds, the delivery is automatically confirmed in Oracle Shipping Execution.

Before using XML messages for third party warehousing, you must:

- Define a Distributed organization
- Define your third party warehouses as trading partners
- Define your organization as a trading partner

The following business events are used for Third Party Warehouse XML transactions:

- oracle.apps.wsh.sup.ssro: Business event to send the Shipment Request Outbound
- oracle.apps.wsh.sup.ssai: Business event to receive the Shipment Advice Inbound
- oracle.apps.fte.ssno.confirm: Business event to receive the Confirm Business Object Document (CBOD) (acceptance or rejection)
- oracle.apps.wsh.tpw.ssri: Business event to receive the Shipment Request Inbound

- oracle.apps.wsh.tpw.ssao: Business event to send the Shipment Advice Outbound
- oracle.apps.wsh.tpw.spwf: Business event to bring the Shipment Request workflow instance to a closure when the cancellation is accepted at the third party instance
- oracle.apps.wsh.tpw.scbod: Business event to send the CBOD Outbound (acceptance or rejection)

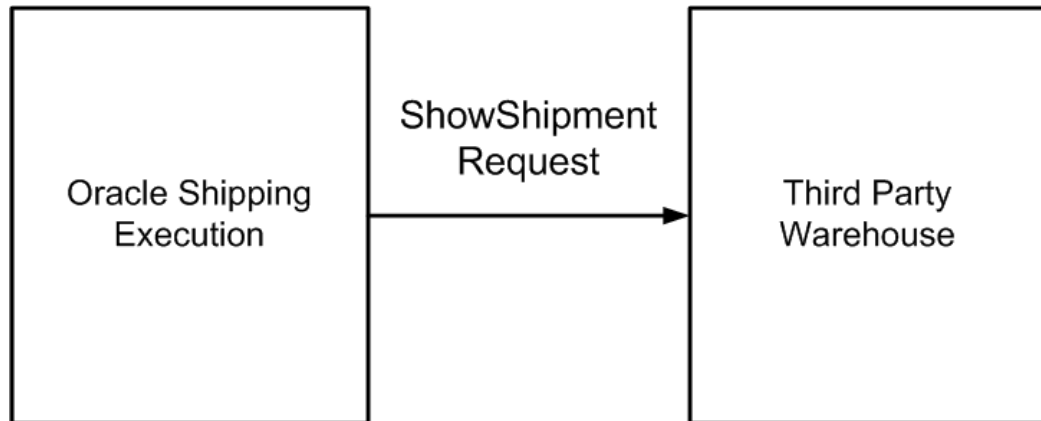
ShowShipment Request Transaction for Third Party Warehousing

Show Shipment Request transaction (the XML equivalent of the ASC X12 940 transaction) is a modified version of the OAG DTD show_shipment_005. This DTD is used to send shipment information from Oracle Shipping Execution to a third party warehouse system.

These transactions contain information such as:

- Delivery Name
- Customer
- Ship To Address
- Ship To Contact
- Requested Pick Up Date and Time
- Requested Delivery Date and Time
- Weight

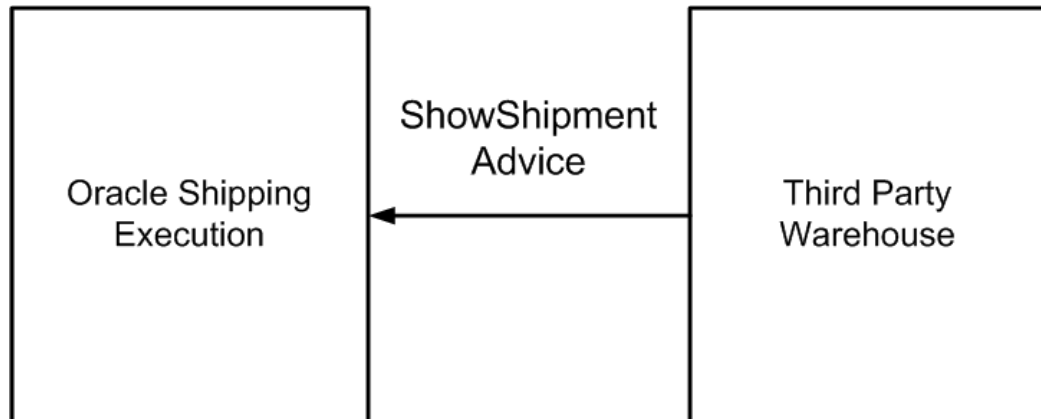
ShowShipment Request (show_shipment_005) for Third Party Warehousing



ShowShipment Advice Transaction for Third Party Warehousing

Show Shipment Advice transaction (the XML equivalent of the ASC X12 945 transaction) is a modified version of the OAG DTD. This DTD is used to send shipment information from a third party warehouse to Oracle Shipping Execution.

ShowShipment Advice (show_shipment_005) for Third Party Warehousing



Defining Trading Partners for Third Party Warehousing

All third party warehouses that receive XML transactions must be defined as trading partners. If there are multiple locations for the third party warehouse, each location must be defined as a trading partner.

All organizations that send and receive XML transactions must be defined as trading partners in Oracle XML Gateway.

To define a warehouse as a trading partner for third party warehousing:

1. Log into the Oracle applications using the XML Gateway responsibility.
2. Navigate to the Trading Partner Setup window.
3. Select Internal from the list of values in the Trading Partner Type field.
4. Select the warehouse name using the list of values in the Trading Partner Name field.
5. Select the warehouse address using the list of values in the Trading Partner Site field.
6. Enter the email address in the Company Admin Email field.
7. In the Trading Partner Details region, use the list of values in the Transaction Type field to choose WSH with the Transaction Sub Type of SSNO.
8. Use the list of values to select WSH_SSNI_OAG721_IN in the Map field.
9. Enter a Source Trading Partner Location Code.
10. Begin a new Trading Partner Detail line.
11. Use the list of values in the Transaction Type field to choose WSH with the External Transaction Type of SHIPMENT and the Direction OUT.
12. Use the list of values to select WSH_SSNO_OAG721_OUT in the Map field.
13. In the Connection/Hub field, select Direct or select a hub through which to communicate with your third party warehouse.

Note: DIRECT enables you to communicate with your third party warehouse directly. A hub is an integration point within your network (either your intranet or the Internet).

If you choose to connect to your third party warehouse using Direct, the XML Gateway will require that you provide information in the following fields:

- Protocol Type
- Username
- Password
- Protocol Address

If you choose to connect to your third party warehouse using a hub, Oracle XML Gateway will require that you select a Username from the list of values only. The

rest of the information will default based on the information entered during the creation of the hub.

See: *Oracle XML Gateway User's Guide* for detailed information on defining a hub.

The Connection/Hub field is only required when the direction of the XML is OUT.

14. Enter a Source Trading Partner Location Code.
15. Begin a new Trading Partner Detail line.
16. Use the list of values in the Transaction Type field to choose ECX with the External Transaction Type of BOD, Transaction Subtype as CONFIRM and the Direction OUT.
17. Use the list of values to select ECX_CBODO_OAG721_OUT_CONFIRM in the Map field.
18. In the Connection/Hub field, select Direct or select a hub through which to communicate with your third party warehouse.

Note: DIRECT enables you to communicate with your third party warehouse directly. A hub is an integration point within your network (either your intranet or the Internet).

If you choose to connect to your third party warehouse using Direct, Oracle XML Gateway will require that you provide information in the following fields:

- Protocol Type
- Username
- Password
- Protocol Address

If you choose to connect to your third party warehouse using a hub, Oracle XML Gateway will require that you select a Username from the list of values only. The rest of the information will default based on the information entered during the creation of the hub.

See: *Oracle XML Gateway User's Guide* for detailed information on defining a hub.

The Connection/Hub field is only required when the direction of the XML is OUT.

19. Enter a Source Trading Partner Location Code.
20. Save your work.

21. Click Code Conversion to define new code conversions.
22. See: *Oracle XML Gateway User's Guide* for detailed information on defining code conversions.

Note: Code conversion categories are seeded. These include OAG and Oracle standard code conversions.

To define organizations as trading partners for third party warehousing:

1. Log into the Oracle applications using the XML Gateway responsibility.
2. Navigate to the Trading Partner Setup window.
3. Select Internal from the list of values in the Trading Partner Type field.
4. Select the organization's location using the list of values in the Trading Partner Name field.
5. Select the organization's location address using the list of values in the Trading Partner Site field.
6. Enter the email address in the Company Admin Email field.
7. In the Trading Partner Details region, use the list of values in the Transaction Type field to choose FTE with the Transaction Sub Type of SSNI.
8. Use the list of values to select WSH_SSNI_OAG721_IN in the Map field.
9. Enter a Source Trading Partner Location Code.
10. Begin a new Trading Partner Detail line.
11. Use the list of values in the Transaction Type field to choose FTE with the Transaction Sub Type of SSNO.
12. Use the list of values to select WSH_SSNO_OAG721_OUT in the Map field.
13. Within the Connection/Hub field, select Direct or select a hub through which to communicate with your carrier.

Note: DIRECT enables you to communicate with your carrier directly. A hub is an integration point within your network (either your intranet or the Internet).

If you choose to connect to your organization using Direct, Oracle XML Gateway will require that you provide information in the following fields:

- Protocol Type
- Username
- Password
- Protocol Address

If you choose to connect to your organization using a hub, Oracle XML Gateway will require that you select only a Username from the list of values. The rest of the information will default based on the information entered during the creation of the hub.

See: *Oracle XML Gateway User's Guide* for detailed information on defining a hub.

The Connection/Hub field is only required when the direction of the XML is OUT.

14. Enter a Source Trading Partner Location Code.
15. Enter 2 in document confirmation field.
16. Begin a new Trading Partner Detail line.
17. Use the list of values in the Transaction Type field to choose ECX with the External Transaction Type of BOD, Transaction Subtype as CONFIRM, and the Direction IN.
18. Use the list of values to select ECX_CBODO_OAG721_IN_CONFIRM in the Map field.
19. Within the Connection/Hub field, select Direct or select a hub through which to communicate with your third party warehouse.

Note: DIRECT enables you to communicate with your third party warehouse directly. A hub is an integration point within your network (either your intranet or the Internet).

If you choose to connect to your organization using Direct, Oracle XML Gateway will require that you provide information in the following fields:

- Protocol Type
- Username
- Password
- Protocol Address

If you choose to connect to your organization using a hub, Oracle XML Gateway will require that you select only a Username from the list of values. The rest of the

information will default based on the information entered during the creation of the hub.

See: *Oracle XML Gateway User's Guide* for detailed information on defining a hub.

The Connection/Hub field is only required when the direction of the XML is OUT.

20. Enter a Source Trading Partner Location Code.

21. Save your work.

22. Click Code Conversion to define new code conversions.

See: *Oracle XML Gateway User's Guide* for detailed information on defining code conversions.

Note: Code conversion categories are seeded. These include OAG and Oracle standard code conversions.

Overview of XML and DTD for Third Party Warehousing

The following is a sample DTD and a sample XML used for third party warehousing in Oracle Shipping Execution. Also, a table is available that displays all required element/attributes, a description of each, and the Oracle table and column for each element/attribute, where applicable.

Sample Document Type Definitions (DTD) for Third Party Warehousing

The following section includes a sample DTD used for third party warehousing in Oracle Shipping Execution.

Third Party Warehousing Sample DTD

```
<! - ===== - >
<!ENTITY % RESOURCES SYSTEM "oagis_resources.dtd">
%RESOURCES;
<! - ===== - >
<!ELEMENT SHOW_SHIPMENT_005 (CNTROLAREA, DATAAREA+)>
<!ATTLIST VERB
  value CDATA #FIXED "SHOW"
>
<!ATTLIST NOUN
  value CDATA #FIXED "SHIPMENT"
>
<!ATTLIST REVISION
  value CDATA #FIXED "005"
>
<!ELEMENT DATAAREA (SHOW_SHIPMENT)
>
<!ELEMENT SHOW_SHIPMENT (SHIPMENT, SHIPUNIT*)
>
<!ELEMENT SHIPMENT ((%DATETIME.DOCUMENT;), (%AMOUNT.DECLAREVAL;)?,
  (%AMOUNT.DOCUMENT.T;)?, (%AMOUNT.ESTFREIGHT;)?, (%AMOUNT.FREIGHT.T;)?,
  (%AMOUNT.ITEM.T;)?, (%DATETIME.CREATION;)?, (%DATETIME.DELIVACT;)?,
  (%DATETIME.DELIVSCHED;)?, (%DATETIME.EARLSTSHIP;)?,
  (%DATETIME.LOADING;)?, (%DATETIME.NEEDDELV;)?, (%DATETIME.PROMDELV;)?,
  (%DATETIME.PROMSHIP;)?, (%DATETIME.SHIP;)?, (%DATETIME.SHIPSCHED;)?,
  (%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?,
  (%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
  (%QUANTITY.SHIPUNIT;)?, (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?,
  (%TEMPRATURE.ACTUAL;)?, (%TEMPRATURE.DELIVERY;)?,
  (%TEMPRATURE.LOADING;)?, DOCUMENTID, SHIPPERID, SYNCID, CARRIER?,
  CARRSRVLVL?, COUNTRYDST?, COUNTRYORG?, DESCRIPTN?, DISTCENTER?, DOCKID?,
  DOCTYPE?, DOCUMENTTRV?, DROPSHIP?, EXPORTLIC?, FRGHTCLS?, FRGHTTERMS?,
  HAZRDMATL?, HDRSTATUS?, IMPORTLIC?, LOADPOINT?, NOTES*, PARTIALSHP?,
  PRIORITY?, ROUTEID?, SHIPNOTES?, SHIPPOINT?, SHIPPRIOR?, SHIPSRVLVL?,
  SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, WAREHOUSE?,
  USERAREA?, PARTNER+, SHIPITEM*, ATTCHREF*, CHARGE*, DOCUMNTREF*)
>
<!ELEMENT SHIPITEM ((%QUANTITY.ITEM;), (%AMOUNT.EXTENDED.T;)?,
  (%AMOUNT.ITEM.T;)?, (%QUANTITY.BACKORDERD;)?, (%QUANTITY.OPEN;)?,
  (%QUANTITY.ORDERED;)?, (%QUANTITY.SHIPPED;)?, (%QUANTITY.VOLUME;)?,
  (%QUANTITY.WEIGHT;)?, ITEM, COUNTRYORG?, DISPOSITN?, EXPORTLIC?,
  FRGHTCLS?, FRGHTITEM?, HAZRDMATL?, IMPORTLIC?, ITEMDESC?, ITEMRV?,
  ITEMRVX?, ITEMX?, LOTLEVEL*, NOTES*, OWNRSHPCDE?, PACKING?, PARTIALSHP?,
  SHIPNOTES?, SHIPPRIOR?, UPC?, USERAREA?, DOCUMNTREF*, INVDETAIL*,
  CHARGE*)
>
<!ELEMENT INVDETAIL ((%QUANTITY.ITEM;)?, DESCRIPTN?, DISPOSITN?,
  LOTLEVEL*, NOTES*, SERIALNUM?, USERAREA?)
>
<!ELEMENT SHIPUNIT ((%AMOUNT.DECLAREVAL;)?, (%AMOUNT.FREIGHT.T;)?,
  (%AMOUNT.ITEM.T;)?, (%AMOUNT.TOTAL.T;)?, (%DATETIME.DELIVACT;)?,
  (%DATETIME.DELIVSCHED;)?, (%DATETIME.LOADING;)?, (%DATETIME.SHIP;)?,
  (%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?, (%QUANTITY.HEIGHT;)?,
  (%QUANTITY.LENGTH;)?, (%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
  (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?, (%QUANTITY.WIDTH;)?,
  (%TEMPRATURE.ACTUAL;)?, (%TEMPRATURE.DELIVERY;)?,
  (%TEMPRATURE.LOADING;)?, CARRIER, SHIPPERID, TRACKINGID, CARRSRVLVL?,
  CONTAINERID?, CONTNRSEAL?, CONTNRATYPE?, COUNTRYDST?, COUNTRYORG?,
  DESCRIPTN?, EXPORTLIC?, FRGHTCLS?, FRGHTITEM?, FRGHTTERMS?, HAZRDMATL?,
```

```

IMPORTLIC?, LOADPOINT?, NOTES*, SHIPMATLID?, SHIPNOTES?, SHIPPOINT?,
SHIPSRLVL?, SHPMTLDESC?, SHPUNITSEQ?, SHPUNITSTS?, SHPUNITTOT?,
SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, USERAREA?, PARTNER+,
ATTCHREF*, CHARGE*, CONTAINER*, DOCUMNTREF*, INVITEM*)
>
<!ELEMENT CONTAINER (CONTAINRID?, CONTRSEAL?, CONTRTYPE?, DESCRIPTN?,
NOTES*, PACKING?, PACKNGDESC?, PARCNTNRID?, SHIPMATLID?, SHPMTLDESC?,
USERAREA?, DOCUMNTREF*, INVITEM*, CHARGE*)
>
<!ELEMENT INVITEM ((%QUANTITY.ITEM;), (%AMOUNT.EXTENDED.T;)?,
(%AMOUNT.ITEM.T;)?, (%QUANTITY.VOLUME;)?, (%QUANTITY.WEIGHT;)?, ITEM,
COUNTRYORG?, FRGHTCLS?, FRGHTITEM?, HAZRDMATL?, ITEMDESC?, ITEMRV?,
ITEMRVX?, ITEMX?, LOTLEVEL*, NOTES*, OWNRSHPCDE?, SHIPNOTES?, UPC?,
USERAREA?, DOCUMNTREF*, INVDETAIL*)
>
@

```

```

115.1.11510.1
log
@'mkbranch'
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ - >
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@

```

```

115.1.1159.1
log
@mkbranch
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ - >
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@

```

```

115.1.1158.1
log
@mkbranch
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ - >
d6 2
a7 2

```

```

$Revision: 115.1 $
$Date: 2002/03/28 22:47:30 $
@

115.0
log
@initial version
@
text
@d1 1
a1 1
<!-- $Header: 161_show_shipment_005.dtd 115.0 2002/02/15 05:27:39
rvishnuv noship $ - >
d6 2
a7 2
$Revision: 7.2.1 $
$Date: 31 October 2001 $
d11 1
a11 1
$Name: 161_show_shipment_005.dtd $
d22 1
a22 1
SHIPMENT ( PARTNER+, SHIPITEM+, ATTCHREF*, CHARGE*, DOCUMNTREF*,)
d59 1
a59 1
<!ELEMENT SHIPMENT ((%DATETIME.DOCUMENT;), (%AMOUNT.DECLAREVAL;)?,
(%AMOUNT.DOCUMENT.T;)?, (%AMOUNT.ESTFREIGHT;)?, (%AMOUNT.FREIGHT.T;)?,
(%AMOUNT.ITEM.T;)?, (%DATETIME.CREATION;)?, (%DATETIME.DELIVACT;)?,
(%DATETIME.DELIVSCHD;)?, (%DATETIME.EARLSTSHIP;)?,
(%DATETIME.LOADING;)?, (%DATETIME.NEEDDELV;)?, (%DATETIME.PROMDELV;)?,
(%DATETIME.PROMSHIP;)?, (%DATETIME.SHIP;)?, (%DATETIME.SHIPSCHD;)?,
(%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?,
(%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
(%QUANTITY.SHIPUNIT;)?, (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?,
(%TEMPRATURE.ACTUAL;)?, (%TEMPRATURE.DELIVERY;)?,
(%TEMPRATURE.LOADING;)?, DOCUMENTID, SHIPPERID, SYNCID, CARRIER?,
CARRSRVLVL?, COUNTRYDST?, COUNTRYORG?, DESCRIPTN?, DISTCENTER?, DOCKID?,
DOCTYPE?, DOCUMENTRV?, DROPSHIP?, EXPORTLIC?, FRGHTCLS?, FRGHTTERMS?,
HAZRDMATL?, HDRSTATUS?, IMPORTLIC?, LOADPOINT?, NOTES*, PARTIALSHP?,
PRIORITY?, ROUTEID?, SHIPNOTES?, SHIPPOINT?, SHIPPRIOR?, SHIPSRVLVL?,
SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, WAREHOUSE?,
USERAREA?, PARTNER+, SHIPITEM+, ATTCHREF*, CHARGE*, DOCUMNTREF*)
@

```

Sample XML Transactions for Third Party Warehousing

Third Party Warehousing Sample XML

The following table lists each of the ShowShipment elements/attributes, whether they are required or optional, and whether they are for inbound, outbound, or both. A sample OUTBOUND ShowShipment Request XML transaction for third party warehousing follows the table and a sample INBOUND ShowShipment Request XML transaction for third party warehousing follows the OUTBOUND sample.

The Required? column indicates whether the element/attribute is required or optional for INBOUND (I), OUTBOUND (O), or BOTH (B).

Third Party Warehousing XML Field Information

Field Name	Required?	Transaction Table.Column	Description
CNTROL AREA	Required (B)		The fields included in this area provide information about the XML document
BSR	Required (B)		Shows the Business Service Request name per OAGI
VERB	Required (B)		Value is Show
NOUN	Required (B)		Value is Shipment
REVISION	Required (B)		Value is 005
SENDER	Required (B)		Provides information on the system that sends the document
LOGICALID	Required (B)		Sender system identifier
COMPONENT	Required (B)		Sender application name. Value is FTE
TASK	Required (B)		Event or Action
REFERENCEID	Required (B)		Unique reference ID for this document
CONFIRMATION	Required (B)		Confirmation when document is received
LANGUAGE	Required (B)		Language in which the text fields are transmitted
CODEPAGE	Required (B)		Character set used in this XML document

Field Name	Required?	Transaction Table.Column	Description
AUTHID	Required (B)		System ID of sender. Value is APPS
DATETIME (CREATION)(T)	Required (B)		Creation date and time of the XML document
SHIPMENT	Required (B)		Contains the general document information for goods being transported to a particular partner destination
SHIPMENT.DATETIME	Required (B)	Sys date time	Contains date and time stamp of the latest update to the SHIPMENT document
SHIPMENT.DOCUMENTID (SHIPMENT)	Required (B)	WSH_TRANSACTION_HISTORY.DOCUMENT_ID	Document ID
SHIPMENT.SHIPPER ID	Required (B)	LOCATION_CODE from HR_LOCATIONS_ALL_TL based on the WSH_NEW_DELIVERIES.INITIAL_PICKUP_LOCATION_ID	Ship From party

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.SYNCID	Required (B)		EDI code conversions: "N" for Add, "R" for Change, "F" for Cancel. XML: "A" = Add, "D" = Delete "C" = Change EDI: "N" for Add, "F" for Delete
SHIPMENT.AMOUNT (DECLAREVAL)	Not used	Not used	Not used
SHIPMENT.AMOUNT (DOCUMENT)	Not used	Not used	Not used
SHIPMENT.AMOUNT (ESTFREIGHT)	Not used	Not used	Not used
SHIPMENT.AMOUNT (FREIGHT)	Not used	Not used	Not used
SHIPMENT.AMOUNT (ITEM)	Not used	Not used	Not used
SHIPMENT.CARRIER	Optional (B)	WSH_NEW_DELIVERIES.CARRIER_ID,	Freight carrier
SHIPMENT.CARRIER RSRVLVL	Optional (B)	WSH_NEW_DELIVERIES.SERVICE_LEVEL	Carrier service level
SHIPMENT.COUNT RYDST	Required (B)	WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID,	Destination country
SHIPMENT.COUNT RYORG	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.DATETIME (CREATION)	Optional (O)	WSH_NEW_DELIVERIES.CREATION_DATE	Date and time of delivery creation
SHIPMENT.DATETIME (DELIVACT)	Optional (B)	WSH_TRIP_STOPS.ACTUAL_ARRIVAL_DATE	Actual delivery date and time
SHIPMENT.DATETIME (DELIVSCHED)	Optional (B)	WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_DATE	Scheduled delivery date and time
SHIPMENT.DATETIME (EARLSTSHIP)	Not used	Not used	Not used
SHIPMENT.DATETIME (LOADING)	Not used	Not used	Not used
SHIPMENT.DATETIME (NEEDELV)	Not used	Not used	Not used
SHIPMENT.DATETIME (PROMDELV)	Not used	Not used	Not used
SHIPMENT.DATETIME (PROMSHIP)	Not used	Not used	Not used
SHIPMENT.DATETIME (SHIP)	Optional (B)	WSH_TRIP_STOPS.ACTUAL_DEPARTURE_DATE	Actual ship date/time
SHIPMENT.DATETIME (SHIPSCHED)	Optional (B)	WSH_NEW_DELIVERIES.INITIAL_PICKUP_DATE	Scheduled ship date and time
SHIPMENT.DESCRPTION	Optional (B)	WSH_NEW_DELIVERIES.DESCRPTION	Description of the shipment
SHIPMENT.DISTCENTER	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.DOCKID	Not used	Not used	Not used
SHIPMENT.DOCTYPE	Required (B)	'SR' (SHIPMENT REQUEST FOR OUTBOUND) AND 'SA' (SHIPMENT ADVICE FOR INBOUND) at the Supplier Instance. 'SA' (SHIPMENT ADVICE FOR OUTBOUND) AND 'SR' (SHIPMENT REQUEST FOR INBOUND) at the Third Party Warehouse Instance.	The OAG document name. SHIPMENT REQUEST for outbound and SHIPMENT ADVICE for inbound
SHIPMENT.DOCUMENTENTRY	Optional (B)		Revision number of this document. '001' for 'Add' and '002' for 'Delete'
SHIPMENT.DROSHIP	Not used	Not used	Not used
SHIPMENT.EXPORTLIC	Not used	Not used	Not used
SHIPMENT.FRGHTCLS	Not used	Not used	Not used
SHIPMENT.FRGHTTERMS	Optional (B)	WSH_NEW_DELIVERIES.FREIGHT_TERMINS_CODE	Indicates how the freight cost for a shipment of goods should be billed
SHIPMENT.HAZRDMATL	Not used	Not used	Not used
SHIPMENT.HDRSTATUS	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.IMPORT LIC	Not used	Not used	Not used
SHIPMENT.LOADPOINT	Not used	Not used	Not used
SHIPMENT.NOTES1 – NOTES9	Not used	Not used	Not used
SHIPMENT.OPERAMT (FREIGHT)	Not used	Not used	Not used
SHIPMENT.PARTIALSHIP	Not used	Not used	Not used
SHIPMENT.PRIORITY	Not used	Not used	Not used
SHIPMENT.QUANTITY (ESTWEIGHT)	Optional (B)	WSH_NEW_DELIVERIES.GROSS_WEIGHT WSH_NEW_DELIVERIES.WEIGHT_UOM_CODE	The estimated weight of a shipment
SHIPMENT.QUANTITY (LOADINGWT)	Optional (B)	WSH_NEW_DELIVERIES.GROSS WEIGHT – WSH_NEW_DELIVERIES.NET WEIGHT	Indicates the weight of the shipment at loading time
SHIPMENT.QUANTITY (NETWEIGHT)	Optional (B)	WSH_NEW_DELIVERIES.NET_WEIGHT WSH_NEW_DELIVERIES.WEIGHT_UOM_CODE	Indicates the net weight of the shipment
SHIPMENT.QUANTITY (SHIPUNIT)	Optional (B)	WSH_NEW_DELIVERIES.NUMBER_OF_LPN	Indicates the quantity of units shipped, such as containers or boxes

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.QUANTITY (TOTWEIGHT)	Optional (B)	WSH_NEW_DELIVERIES.GROSS_WEIGHT WSH_NEW_DELIVERIES.WEIGHT_UOM_CODE	Indicates the total weight of a shipment
SHIPMENT.QUANTITY (VOLUME)	Optional (B)	WSH_NEW_DELIVERIES.VOLUME WSH_NEW_DELIVERIES.VOLUME_UOM_CODE	Indicates the volume of the shipment
SHIPMENT.ROUTEID	Not used	Not used	Not used
SHIPMENT.RELEASENUM	Not used	Not used	Not used
SHIPMENT.RELEASENUMTOT	Not used	Not used	Not used
SHIPMENT.ROUTETYPE	Not used	Not used	Not used
SHIPMENT.SHIPNOTES	Optional (B)	WSH_NEW_DELIVERIES.SHIPPING_MARKS	This is a free form area that contains special instructions
SHIPMENT.SHIPPOINT	Required (O)	LOCATION FROM HZ_CUST_SITE_USES_ALL based on the WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID	Ship to location
SHIPMENT.SHIPPRIORITY	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.SHIPSRV LVL	Optional (B)	WSH_NEW_DELIVERIES.SERVICE_LEVEL	Service level of the shipment
SHIPMENT.SPECIAL HND	Not used	Not used	Not used
SHIPMENT.STAGEPOINT	Not used	Not used	Not used
SHIPMENT.TEMPRACTURE (ACTUAL)	Not used	Not used	Not used
SHIPMENT.TEMPRACTURE (DELIVERY)	Not used	Not used	Not used
SHIPMENT.TEMPRACTURE (LOADING)	Not used	Not used	Not used
SHIPMENT.TRANSMETHOD	Optional (B)	WSH_NEW_DELIVERIES.MODE_OF_TRANSPORT	Identifies the general type of carrier transportation used to deliver goods
SHIPMENT.TRANSFERMS	Optional (B)	WSH_NEW_DELIVERIES.FOB_CODE	Ownership transfer terms
SHIPMENT.WAREHOUSE	Optional (B)	WSH_NEW_DELIVERIES.ORGANIZATION_ID	The storage facility for inventory

Field Name	Required?	Transaction Table.Column	Description
SHIPMENT.USERAREA	Optional (B)	WSH_NEW_DELIVERIES.DELIVERY_NAME	The USERAREA is defined by embedding elements and attributes for each new Field Identifier or Segment needed within the USERAREA. When a new Field Identifier or a new Segment is determined to be necessary, and it is not included in the OAGIS specification, new tags can be developed by the project team for the specific integration project. These new tags can then be used to describe the fields and segments within the USERAREA. The USERAREA may contain multiple fields or segments coded in this way. These are all embedded within the one special field called USERAREA.
ORACLE.DELIVERY_NAME		WSH_NEW_DELIVERIES.LOADING_SEQUENCE	
ORACLE.LOADING_SEQUENCE		WSH_NEW_DELIVERY descriptive flex fields	
ORACLE.VEHICLE_NUM_PREFIX		WSH_TRIPS.VEHICLE_NUM_PREFIX	
ORACLE.VEHICLE_NUMBER		WSH_TRIPS.VEHICLE_NUMBER	
ORACLE.ROUTE_ID		WSH_TRIPS.ROUTE_ID	
ORACLE.ROUTING_INSTRUCTIONS		WSH_TRIPS.ROUTING_INSTRUCTIONS	
ORACLE.DEPARTURE_SEAL_CODE		WSH_TRIP_STOPS.DEPARTURE_SEAL_CODE	
ORACLE.ORIG_DOC_ID			
ORACLE.CUSTOMER_NAME			
ORACLE.CUSTOMER_NUMBER			
ORACLE.WAYBILL			
ORACLE.INTMED_SHIP_TO_LOCATION			
ORACLE.MANUAL_WV			
ORACLE.SHIPMENT_DIRECTION			
ORACLE.OPERATOR			
ORACLE.DEScriptive_FLEXFIELDS			

Field Name	Required?	Transaction Table.Column	Description
PARTNER.	Required (O)		Partner area for Customer, Supplier, Ship From, or Ship To
PARTNER.NAME	Required (O)	<p>- If this partner segment is the Ship To segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID</p> <p>- If this partner segment is the Ship from segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ORGANIZATION_ID</p> <p>- If this partner segment is the Customer segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.CUSTOMER_ID</p> <p>- If this partner segment is the Supplier segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ORGANIZATION_ID</p>	Corporate name of a business entity in the shipment address
PARTNER.ONETIME	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
PARTNER.PARTNRID	Required (O)	LOCATION FROM HZ_CUST_SITE_USES_ALL based on WSH_NEW_DELIVERIES.ULTIMATE_DROFF_LOCATION_ID converted to external value, or LOCATION_CODE FROM HR_LOCATIONS_ALL_TL based on WSH_NEW_DELIVERIES.INITIAL_PICKUP_LOCATION_ID AND ORGANIZATION_ID converted to external value.	Partner ID

Field Name	Required?	Transaction Table.Column	Description
PARTNER.PARTNRTYPE	Required (O)	WSH_LOOKUPS.MEANING based on lookup type = 'WSH_PARTNER_TYPE' (Values are 'ShipFrom' or 'ShipTo')	<p>Indicates the type of partner entity.</p> <p>Valid values are:</p> <p>1) "ShipTo" - The delivery location of the customer.</p> <p>2) "BillTo" - The customer location the bill goes to.</p> <p>3) "SoldTo" - The customer who is responsible for ordering the goods or services.</p> <p>4) "Supplier" - The partner who is responsible for providing the goods or services.</p> <p>5) "Carrier" - The partner responsible for delivering the goods or services.</p> <p>6) "ShipFrom" - The shipping location of the partner who is responsible for providing the goods or services. Ship from can be at multiple organizational levels</p>
PARTNER.ACTIVE	Not used	Not used	Not used
PARTNER.CURRENCY	Optional (O)	WSH_NEW_DELIVERIES.CURRENCY_CODE	Indicates the currency
PARTNER.DESCRPTION	Optional (O)		Partner description

Field Name	Required?	Transaction Table.Column	Description
PARTNER.DUNSNUMBER	Optional (O)	HZ_PARTIES.DUNS_NUMBER for Ship To	Partner Dun and Bradstreet number
PARTNER.GLENTITYS	Not used	Not used	Not used
PARTNER.NAME2-NAME9	Not used	Not used	Not used
PARTNER.PARENTID	Not used	Not used	Not used
PARTNER.PARTNIDX	Not used	Not used	Not used
PARTNER.PARTNRATG	Not used	Not used	Not used
PARTNER.PARTNRROLE	Not used	Not used	Not used
PARTNER.PAYMETHOD	Not used	Not used	Not used
PARTNER.TAXEXEMPT	Not used	Not used	Not used
PARTNER.TAXID	Not used	Not used	Not used
PARTNER.TERMID	Not used	Not used	Not used
PARTNER.USERAREA	Optional (O)	WSH_NEW_DELIVERIES.INTMED_SHIP_TO_LOCATION_ID WSH_NEW_DELIVERIES.POOLED_SHIP_TO_LOCATION_ID	Partner data user area

Field Name	Required?	Transaction Table.Column	Description
PARTNER – ADDRESS.	Required (O)		This area is for the partner address information
PARTNER– ADDRESS.ADDRLIN E1-ADDRLINE9	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner address
PARTNER– ADDRESS.ADDRTYPE	Not used	Not used	Not used
PARTNER– ADDRESS.CITY	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner city
PARTNER– ADDRESS.COUNTRY	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner country

Field Name	Required?	Transaction Table.Column	Description
PARTNER- ADDRESS.COUNTY	Optional (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner county
PARTNER- ADDRESS.DESCRIP TION	Optional (O)		Partner address description
PARTNER- ADDRESS.FAX1-FAX 9	Optional (O)	HZ_CONTACT_POINTS.RAW_PHONE_NUMBER for the PARTY_ID, and CONTACT_POINT_TYPE = "FAX" for "Ship To" and HR_LOCATIONS_ALL.TELEPHONE_NUMBER_2 for "Ship From"	Partner fax numbers
PARTNER- ADDRESS.POSTALC ODE	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner postal code

Field Name	Required?	Transaction Table.Column	Description
PARTNER- ADDRESS.REGION	Optional (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner region
PARTNER- ADDRESS.STATEPROVN	Required (O)	Extracted from the WSH_LOCATIONS using the location id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES table.	Partner state or province
PARTNER- ADDRESS.TAXJURSDCTN	Not used	Not used	Not used
PARTNER- ADDRESS.TELEPHONE1-TELEPHONE9	Required (O)	HZ_CONTACT_POINTS.RAW_PHONE_NUMBER for the PARTY_ID, and CONTACT_POINT_TYPE = "GEN" for "Ship To" and HZ_LOCATIONS_ALL.TELEPHONE_NUMBER_1 for "Ship From".	Partner telephone numbers
PARTNER- ADDRESS.URL	Optional (O)	HZ_CONTACT_POINTS.URL for the PARTY_ID and CONTACT_POINT_TYPE = "GEN" for "Ship To"	Partner web site URL

Field Name	Required?	Transaction Table.Column	Description
PARTNER- ADDRESS.USERARE A	Not used	Not used	Not used
PARTNER – CONTACT.	Required (O)		Partner contact information
PARTNER- CONTACT.NAME1	Required (O)	HZ_PARTIES.PERSON_FIRST_NAME + PERSON_MIDDLE_NAME + PERSON_LAST_NAME based on customer and ship to location on wsh_new_deliveries and based on ship_to_contact_id on wsh_delivery_details	Partner contact name
PARTNER- CONTACT.CONTACT TYPE	Not used	Not used	Not used
PARTNER- CONTACT.DESCRIP TION	Not used	Not used	Not used
PARTNER- CONTACT.EMAIL	Not used	Not used	Not used
PARTNER- CONTACT.FAX1 - FAX9	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
PARTNER- CONTACT.NAME2 - NAME9	Optional (O)	HZ_PARTIES.PERSON_FIRST_NAME + PERSON_MIDDLE_NAME + PERSON_LAST_NAME based on customer and ship to location on wsh_new_deliveries and based on ship_to_contact_id on wsh_delivery_details.	Additional partner contact names
PARTNER- CONTACT.TELEPHONE1 -TELEPHONE9	Optional (O)	HZ_CONTACT_POINTS.RAW_PHONE_NUMBER based on customer and ship to location on wsh_new_deliveries and based on ship_to_contact_id on wsh_delivery_details.	Partner contact phone numbers
PARTNER- CONTACT.USERAREA	Not used	Not used	Not used
DOCUMENTREF.	Required (O)		This area contains the details of the document type

Field Name	Required?	Transaction Table.Column	Description
DOCUMNTREF.DOC TYPE	Required (O)	'SR' (SHIPMENT REQUEST FOR OUTBOUND) AND 'SA' (SHIMENT ADVICE FOR INBOUND) at the Supplier Instance. 'SA' (SHIPMENT ADVICE FOR OUTBOUND) AND 'SR' (SHIMENT REQUEST FOR INBOUND) at the Third Party Warehouse Instance.	The document type: SHIPMENT_REQUE ST
DOCUMNTREF.DOC UMENTID	Required (O)	WSH_TRANSACTION_ HISTORY.ORIG_ DOCUMENT_NUMB ER	The document ID
DOCUMNTREF.PAR TNRID	Optional (O)	ORGANIZATION_C ODE based onWSH_NEW_DELI VERIES.ORGANIZA TION_ID	The partner ID
DOCUMNTREF.PAR TNRTYPE	Not used	Not used	Not used
DOCUMNTREF.DES CRIPTN	Not used	Not used	Not used
DOCUMNTREF.DOC UMENTRV	Not used	Not used	Not used
DOCUMNTREF.LIN ENUM	Not used	Not used	Not used
DOCUMNTREF.NOT ES1 – NOTES9	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
DOCUMNTREF.SCH LINENUM	Not used	Not used	Not used
DOCUMNTREF.SUB LINENUM	Not used	Not used	Not used
DOCUMNTREF.USE RAREA	Not used	Not used	Not used
SHIPITEM.	Required (O)		Represents the request to ship specific quantity of goods
SHIPITEM.ITEM	Required (O)	WSH_DELIVERY_DE TAILS.INVENTORY_ ITEM_ID (foreign key to MTL_SYSTEM_ITEM S) Derive Item_Number from MTL_SYSTEM_ITEM S	Contains the identifier of the item
SHIPITEM.QUANTIT Y (ITEM)	Required (O)	WSH_DELIVERY_DE TAILS.REQUESTED_ QUANTITY	Indicates the quantity of items to be shipped
SHIPITEM.AMOUNT (EXTENDED)	Not used	Not used	Not used
SHIPITEM.AMOUNT (ITEM)	Not used	Not used	Not used
SHIPITEM.COUNTR YORG	Required (O)	WSH_DELIVERY_DE TAILS.COUNTRY_O F_ORIGIN	Country of origin
SHIPITEM.DISPOSIT N	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.EXPORTLIC	Not used	Not used	Not used
SHIPITEM.FRGHTCLS	Optional (O)	MTL_CATEGORY_SE TS_VL.CATEGORY_S ET_NAME based on WSH_DELIVERY_DE TAILS.FREIGHT_CL ASS_CAT_ID converted to external value	Qualifies goods for purposes of freight cost calculation
SHIPITEM.FRGHTITEM	Not used	Not used	Not used
SHIPITEM.HAZRDMATL	Optional (O)	PO_HAZARD_CLAS SES_VL . HAZARD_CLASS based on WSH_DELIVERY_DE TAILS.HAZARD_CL ASS_ID converted to external value.	Contains a hazardous material code or description that can be a free form area or an agreed upon set of codes for the applications
SHIPITEM.IMPORTLIC	Not used	Not used	Not used
SHIPITEM.ITEMDESC	Optional (O)	WSH_DELIVERY_DE TAILS.ITEM_DESCR PTION	Item description
SHIPITEM.ITEMRV	Optional (O)	WSH_DELIVERY_DE TAILS.REVISION	Item revision code
SHIPITEM.ITEMRVX	Not used	Not used	Not used
SHIPITEM.ITEMX	Optional (O)	MTL_CUSTOMER_IT EMS.CUSTOMER_IT EM_NUMBER based on WSH_DELIVERY_DE TAILS.CUSTOMER_I TEM_ID	Customer's item number

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.LINENUM	Not used	Not used	Not used
SHIPITEM.LOTLEVEL1 – LOTLEVEL2	Not used	Not used	Not used
SHIPITEM.NOTES1 – NOTES9	Optional (O)	Notes 1 WSH_DELIVERY_DE TAILS.PACKING_IN STRUCTIONS	Item notes
SHIPITEM.OWNRSH PCDE	Optional (O)	WSH_DELIVERY_DE TAILS.FOB_CODE	Used to identify who or which organization owns the item or part. The content of this Field ID is user defined based on a specific Customer or Supplier
SHIPITEM.PACKING	Not used	Not used	Not used
SHIPITEM.PACKING DESC	Not used	Not used	Not used
SHIPITEM.PARTIAL SHIP	Not used	Not used	Not used
SHIPITEM.QUANTITY (BACKORDERD)	Required (I)	Calculate WSH_DELIVERY_DE TAILS.BACKORDER ED_QUANTITY.	Quantity of items backordered
SHIPITEM.QUANTITY (OPEN)	Not used	Not used	Not used
SHIPITEM.QUANTITY (ORDERED)	Required (B)	WSH_DELIVERY_DE TAILS.SRC_REQUESTED_QUANTITY	The quantity of a material or product ordered by the customer

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.QUANTITY (SHIPPED)	Required (B)	WSH_DELIVERY_DE TAILS.SHIPPED_QUANTITY	Represents the actual quantity shipped
SHIPITEM.QUANTITY (VOLUME)	Optional (B)	WSH_DELIVERY_DE TAILS.VOLUME	The volume, or amount of space, that is occupied by an item, container or shipment
SHIPITEM.QUANTITY (WEIGHT)	Required (B)	WSH_DELIVERY_DE TAILS.NET_WEIGHT	The weight of an item, container, or shipment
SHIPITEM.SHIPMATERIALID	Not used	Not used	Not used
SHIPITEM.SHIPNOTES	Optional (B)	WSH_DELIVERY_DE TAILS.SHIPPING_INSTRUCTIONS	Notes area used for any additional information
SHIPITEM.SHIPPRIOR	Not used	Not used	Not used
SHIPITEM.SHIPMTLDESC	Not used	Not used	Not used
SHIPITEM.UPC	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPITEM.USERAREA	Optional (B)	WSH_DELIVERY_DE TAILS.DELIVERY_D ETAIL_ID	Ship item user area
ORACLE.DELIVERY _DETAIL_NUMBER		WSH_DELIVERY_AS SIGNMENTS.PAREN T_DELIVERY_DETAI L_ID	
ORACLE.PARENT_D ELIVERY_DETAIL_N UMBER		WSH_DELIVERY_DE TAILS.SHIP_TOLER ANCE_ABOVE	
ORACLE.SHIP_TOLE RANCE_ABOVE		WSH_DELIVERY_DE TAILS.SHIP_TOLER ANCE_BELOW	
ORACLE.SHIP_TOLE RANCE_BELOW		WSH_DELIVERY_DE TAILS.LOAD_SEQUE NCE_NUMBER	
ORACLE.LOAD_SEQ UENCE_NUMBER		WSH_DELIVERY_DE TAILS descriptive flexfields	
ORACLE.SHIP_SET_ NUMBER		WSH_DELIVERY_DE TAILS.SHIP_SET_ID	
ORACLE.SUBINVEN TORY		WSH_DELIVERY_DE TAILS.SUBINVENTO RY	
ORACLE.TOP_MOD EL_LINE_NUMBER		WSH_DELIVERY_DE TAILS.TOP_MODEL_ LINE_ID	
ORACLE.SHIP_MOD EL_COMPLETE_FLA G		WSH_DELIVERY_DE TAILS.SHIP_MODEL_ _COMPLETE_FLAG	
ORACLE.COMMODI TY_CODE_CATEGO RY		WSH_DELIVERY_DE TAILS.COMMODITY _CODE_CAT_ID converted to code	
ORACLE.SHIPMENT _PRIORITY_CODE		WSH_DELIVERY_DE TAILS.SHIPMENT_P RORITY_CODE converted to "01", "02"	
ORACLE.DELIVER_ TO_LOCATION			
ORACLE.CUSTOME R_NAME			
ORACLE.CUSTOME R_NUMBER			
ORACLE.TXN_SRC_ LINE_NUMBERQUA			

Field Name	Required?	Transaction Table.Column	Description
NTITY (TOTWEIGHT)		, or "03" for XML,	
ORACLE.MANUAL_ WV		WSH_DELIVERY_DE TAILS.DELIVER_TO_ LOCATION_ID	
ORACLE.LINE_DIRE CTION		CONTACT segment sends CONTACT NAME and PHONE NUMBER of the contact based on the ship_to_contact_id on wsh_delivery_details.	
ORACLE.REQUEST_ DATE_TYPE_CODE			
DATETIME (ORACLE.EARLSTPI CK)			
SHIPITEM.USERARE A (continued)			
DATETIME (ORACLE.LATESTPI CK)			
DATETIME (ORACLE.EARLSTD ROP)			
DATETIME (ORACLE.LATESTD ROP)			
DATETIME (ORACLE.SCHEDUL ED)			
DATETIME (ORACLE.REQUEST ED)			
CONTACT Segment			
CHARGE.	Optional (B)		Represents additional freight or transportation charges associated with the SHPITEM

Field Name	Required?	Transaction Table.Column	Description
CHARGE.CHARGEID	Required (B)	WSH_FREIGHT_COST_TYPES.NAME based on WSH_FREIGHT_COST_TYPES.FREIGHT_COST_TYPE_ID converted to external value	Charge ID
CHARGE.OPERAMT (EXTENDED) (T)	Required (B)	WSH_FREIGHT_COST_TYPES.UNIT_AMOUNT WSH_FREIGHT_COST_TYPES.CURRENCY_CODE	Extended operating cost amount
CHARGE.AMOUNT (EXTENDED) (T)	Not used	Not used	Not used
CHARGE.CHARGETYPE	Optional (B)	WSH_FREIGHT_COST_TYPES.NAME based on WSH_FREIGHT_COST_TYPES.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	Charge type
CHARGE.CHGLINE NUM	Not used	Not used	Not used
CHARGE.DESCRPTION	Not used	Not used	Not used
CHARGE.USERAREA	Optional (I)	WSH_FREIGHT_COST_TYPES.descriptive flexfields	User area
INVDETAIL.	Required (I)		Inventory item attribute detail associated with the SHIPITEM

Field Name	Required?	Transaction Table.Column	Description
INVDETAIL.DESCRIP PTN	Optional (I)	WSH_DELIVERY_DE TAILS.ITEM_DESCRIP TION	Inventory item description
INVDETAIL.DISPOSTN	Not used	Not used	Not used
INVDETAIL.LOTLEVEL1 – LOTLEVEL2	Optional (B)	WSH_DELIVERY_DE TAILS.LOT_NUMBER WSH_DELIVERY_DE TAILS.SUBLOT_NUM BER	Lot is the identifier that is assigned to a specific manufactured or purchased item/product in a specific quantity and is used to track that grouping of item/product. The Field Identifiers LOTLEVEL1 - LOTLEVEL2 are used to identify lots and subdivisions of lots Use of Field Identifier LOTLEVEL2 requires that LOTLEVEL1 must exist and have value
INVDETAIL.NOTES1 - NOTES9	Not used	Not used	Not used
INVDETAIL.QUANTITY (ITEM)	Required (B)	WSH_DELIVERY_DE TAILS.REQUESTED QUANTITY/WSH_D ELIVERY_DETAILS.R EQUESTED_QUANTI TY_UOM	Item quantity
INVDETAIL.SERIAL NUM	Optional (B)	WSH_DELIVERY_DE TAILS.SERIAL_NUM BER	Item serial number

Field Name	Required?	Transaction Table.Column	Description
INVDETAIL.USERAREA	Optional (B)	WSH_DELIVERY_DE TAILS.TO_SERIAL_N UMBER WSH_DELIVERY_DE TAILS.PREFERRED_ GRADE	Item user area
DOCUMNTREF.	Required (O)		This area contains the details of the document type
DOCUMNTREF.DOC TYPE	Required (O)	Literal Purchase Order	The document type: SHIPMENT_REQUE ST
DOCUMNTREF.DOC UMENTID	Optional (O)	WSH_DELIVERY_DE TAILS.CUST_PO_NU MBER	The document ID
DOCUMNTREF.PAR TNRID	Optional (O)	DELIVER TO LOCATION based on the WSH_DELIVERY_DE TAILS.DELIVER_TO_ LOCATION_ID	The partner ID
DOCUMNTREF.PAR TNRTYPE	Not used	Not used	Not used
DOCUMNTREF.DES CRIPTN	Not used	Not used	Not used
DOCUMNTREF.DOC UMENTRV	Not used	Not used	Not used
DOCUMNTREF.LIN ENUM	Not used	Not used	Not used
DOCUMNTREF.NOT ES1 – NOTES9	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
DOCUMNTREF.SCH LINENUM	Not used	Not used	Not used
DOCUMNTREF.SUB LINENUM	Not used	Not used	Not used
DOCUMNTREF.USE RAREA	Not used	Not used	Not used
CHARGE.	Required (I)		Represents additional freight or transportation charges associated with the shipment
CHARGE.CHARGEID	Required (I)	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	Charge ID
CHARGE.OPERAMT (EXTENDED)	Required (I)	WSH_FREIGHT_COSTS.TOTAL_AMOUNT WSH_FREIGHT_COSTS.CURRENCY_CODE	Total operations charges
CHARGE.AMOUNT (EXTENDED) (T)	Required (I)		Total charges
CHARGE.CHARGETYPE	Optional (I)	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	Charge type
CHARGE.CHGLINE NUM	Optional (I)		Line number that the charge is in reference to
CHARGE.DESCRPTION	Optional (I)		Description of the charge

Field Name	Required?	Transaction Table.Column	Description
CHARGE.USERAREA	Optional (I)	WSH_FREIGHT_COS	Charge user area
A		TS descriptive flexfields	
DISTRIBUTION.	Not used	Not used	Not used
DISTRIBUTION.BUS NAREA	Not used	Not used	Not used
DISTRIBUTION.COSTCENTER	Not used	Not used	Not used
DISTRIBUTION.DEPARTMENT	Not used	Not used	Not used
DISTRIBUTION.DIVISION	Not used	Not used	Not used
DISTRIBUTION.ELEMENT1 - ELEMENT999	Not used	Not used	Not used
DISTRIBUTION.FUND	Not used	Not used	Not used
DISTRIBUTION.GEOGRAPHY	Not used	Not used	Not used
DISTRIBUTION.GLNTITYS	Not used	Not used	Not used
DISTRIBUTION.GLNOMACCT	Not used	Not used	Not used
DISTRIBUTION.OPERAMT (EXTENDED) (T)	Not used	Not used	Not used
DISTRIBUTION.PROFITCTR	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
DISTRIBUTION.PROJECT	Not used	Not used	Not used
DISTRIBUTION.UNIT	Not used	Not used	Not used
DISTRIBUTION.WAREHOUSE	Not used	Not used	Not used
DISTRIBUTION.USERAREA	Not used	Not used	Not used
SHIPUNIT.	Required (B)		Represents a single trackable shipping unit. A shipping unit is a uniquely identifiable assembly or container that holds one or more items to be shipped
SHIPUNIT.CARRIER	Optional (B)	WSH_DELIVERY_DE TAILS.CARRIER_ID converted to external value	Carrier of the ship unit
SHIPUNIT.SHIPPERID	Required (B)	HR_LOCATIONS_TL _ALL.LOCATION_CODE based on WSH_DELIVERY_DE TAILS.SHIP_FROM_ LOCATION_ID converted to external value.	The identifier of the partner who is responsible for packaging inventory goods for shipment
SHIPUNIT.TRACKINGID	Optional (B)	WSH_DELIVERY_DE TAILS.TRACKING_NUMBER	A unique identifier for the purpose of tracking an individual package or shipment
SHIPUNIT.AMOUNT (DECLAREVAL)	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.AMOUNT (FREIGHT) (T)	Not used	Not used	Not used
SHIPUNIT.AMOUNT (ITEM) (T)	Not used	Not used	Not used
SHIPUNIT.AMOUNT (TOTAL) (T)	Not used	Not used	Not used
SHIPUNIT.CARRSERVICELEVEL	Optional (B)	WSH_NEW_DELIVERIES.SERVICE_LEVEL	Service level for the ship unit
SHIPUNIT.CONTAINERSEAL	Optional (B)	WSH_DELIVERY_DETAILS.SEAL_CODE	Seal number on the ship unit
SHIPUNIT.COUNTRYDST	Not used	Not used	Not used
SHIPUNIT.COUNTRYORG	Optional (B)	WSH_DELIVERY_DETAILS.COUNTRY_OF_ORIGIN	Container country organization
SHIPUNIT.DATETIME (DELIVACT)	Optional (B)	WSH_TRIP_STOPS.ACTUAL_ARRIVAL_DATE	Actual delivery date and time
SHIPUNIT.DATETIME (DELIVSCHED)	Optional (B)	WSH_DELIVERY_DETAILS.DATE_SCHEDULED	Delivery schedule date and time
SHIPUNIT.DATETIME (LOADING)	Not used	Not used	Not used
SHIPUNIT.DATETIME (SHIP)	Required (I)	WSH_TRIP_STOPS.ACTUAL_DEPARTURE_DATE	Shipment date and time
SHIPUNIT.DESCRPTION	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.EXPORTLIC	Not used	Not used	Not used
SHIPUNIT.FRGHTCLS	Optional (B)	MTL_CATEGORY_SE TS_VL.CATEGORY_SE ET_NAME based on WSH_DELIVERY_DE TAILS.FREIGHT_CL ASS_CAT_ID converted to external value	Freight class of the ship unit
SHIPUNIT.FRGHTTERMS	Optional (B)	WSH_DELIVERY_DE TAILS.FREIGHT_TER MS_CODE converted to external value	Ship unit freight terms
SHIPUNIT.HAZRDMATL	Optional (B)	WSH_DELIVERY_DE TAILS.HAZARD_CL ASS_ID converted to external value	Contains a hazardous material code or description that can be a free form area or an agreed upon set of codes for the applications involved in an integration scenario
SHIPUNIT.IMPORTLIC	Not used	Not used	Not used
SHIPUNIT.LOADPOINT	Not used	Not used	Not used
SHIPUNIT.NOTES1 - NOTES9	Optional (B)	WSH_DELIVERY_DE TAILS.PACKING_IN STRUCTIONS	Notes for the ship unit
SHIPUNIT.OPERAM T (FREIGHT)	Not used	Not used	Not used
SHIPUNIT.PACKING	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.PACKINGDESC	Not used	Not used	Not used
SHIPUNIT.QUANTITY (ESTWEIGHT)	Not used	Not used	Not used
SHIPUNIT.QUANTITY (HEIGHT)	Optional (B)	Populate from MTL_SYSTEM_ITEMS	Height of the ship unit
SHIPUNIT.QUANTITY (LENGTH)	Optional (B)	Populate from MTL_SYSTEM_ITEMS	Length of the ship unit
SHIPUNIT.QUANTITY (LOADINGWT)	Optional (B)	Calculate WSH_DELIVERY_DETAILED.GROSS WEIGHT - WSH_DELIVERY_DETAILED.NET WEIGHT	Loading weight of the ship unit
SHIPUNIT.QUANTITY (NETWEIGHT)	Optional (B)	WSH_DELIVERY_DETAILED.NET_WEIGHT + WSH_DELIVERY_DETAILED.WEIGHT_UOM_CODE for delivery lines = container item	Net weight of the ship unit
SHIPUNIT.QUANTITY (TOTWEIGHT)	Required (B)	WSH_DELIVERY_DETAILED.GROSS_WEIGHT + WSH_DELIVERY_DETAILED.WEIGHT_UOM_CODE	Total weight of the ship unit

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.QUANTITY (VOLUME)	Optional (B)	WSH_DELIVERY_DE TAILS.VOLUME + WSH_DELIVERY_DE TAILS.VOLUME_UO M_CODE for delivery lines = container item	Volume of the ship unit
SHIPUNIT.QUANTITY (WIDTH)	Not used	Not used	Not used
SHIPUNIT.SHIPMAT LID	Optional (B)	WSH_DELIVERY_DE TAILS.INVENTORY_ ITEM_ID converted to Item Number by joining to mtl_system_items	The delivery detail id of the container record
SHIPUNIT.SHIPNOTES	Optional (B)	WSH_DELIVERY_DE TAILS.SHIPPING_IN STRUCTIONS	The shipping instructions or notes
SHIPUNIT.SHIPPOINT	Optional (B)	WSH_DELIVERY_DE TAILS.DELIVER_TO_ LOCATION_ID converted to LOCATION_CODE	Identifies the location the goods are to be shipped
SHIPUNIT.SHIPSRV LVL	Optional (B)	WSH_NEW_DELIVERIES.SERVICE_LEVEL	A generic code that specifies a particular type or quality of carrier delivery service to be used for transporting goods
SHIPUNIT.SHPMATERIAL DESC	Optional (B)	WSH_DELIVERY_DE TAILS.ITEM_DESCRIPTION	Describes the material or container a product or item is shipped in
SHIPUNIT.SHIPUNIT SEQ	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.SHPUNIT STS	Not used	Not used	Not used
SHIPUNIT.SHPUNIT TOT	Not used	Not used	Not used
SHIPUNIT.SPECIAL HND	Not used	Not used	Not used
SHIPUNIT.STAGEPO INT	Not used	Not used	Not used
SHIPUNIT.TEMPRA TURE (ACTUAL)	Not used	Not used	Not used
SHIPUNIT.TEMPRA TURE (DELIVERY)	Not used	Not used	Not used
SHIPUNIT.TEMPRA TURE (LOADING)	Not used	Not used	Not used
SHIPUNIT.TRACKIN GIDTYPE	Not used	Not used	Not used
SHIPUNIT.TRANSM ETHD	Optional (B)	WSH_NEW_DELIVERIES.MODE_OF_TRANSPORT	Transportation method
SHIPUNIT.TRANSTERMS	Optional (B)	WSH_NEW_DELIVERIES.FOB_CODE	A contractual code used to describe the point where delivery occurs (goods ownership is transferred) in a contract of sale

Field Name	Required?	Transaction Table.Column	Description
SHIPUNIT.USERAREA	Optional (B)	WSH_DELIVERY_DE TAILS.DELIVERY_D ETAIL_ID (To be called as DELIVERY_DETAIL_ NUMBER) WSH_DELIVERY_DE TAILS.PACKING_IN STRUCTIONS WSH_DELIVERY_DE TAILS.DESCRPTIVE _FLEXFIELDS WSH_DELIVERY_DE TAILS.LOT_NUMBE R WSH_DELIVERY_DE TAILS.SUBLOT_NU MBER WSH_DELIVERY_DE TAILS.REVISION WSH_DELIVERY_DE TAILS.SERIAL NUMBER WSH_DELIVERY_DE TAILS.TO_SERIAL NUMBER	User area for the ship unit
CONTAINER.	Optional (B)		Represents information about an intermediate packaging level within the shipping unit. A CONTAINER may or may not have INVITEM inventory associated with it

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.CONTAINRID	Optional (B)	WSH_DELIVERY_DE TAILS.CONTAINER_ NAME	Container ID
CONTAINER.CONTRSEAL	Optional (B)	WSH_DELIVERY_DE TAILS.SEAL_CODE	Identifies the tamper-proof seal placed on a shipping container to prevent pilfering of the contents
CONTAINER.CONTRTYPE	Optional (B)	WSH_DELIVERY_DE TAILS.CONTAINER_ TYPE_CODE	A classification of containers
CONTAINER.DESCRIPTN	Not used	Not used	Not used
CONTAINER.NOTES 1 - NOTES 9	Not used	Not used	Not used
CONTAINER.PACKING	Not used	Not used	Not used
CONTAINER.PACKINGDESC	Not used	Not used	Not used
CONTAINER.PARTNTRID	Not used	Not used	Not used
CONTAINER.SHIPMENT ATLID	Optional (B)	WSH_DELIVERY_DE TAILS.INVENTORY_ ITEM_ID converted to ITEM NUMBER by joining to mtl_system_items	Container part number

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.USER AREA	Optional (B)	WSH_DELIVERY_AS SIGNMENTS.DELIVERY_DETAIL_ID (To be called as DELIVERY_DETAIL_NUMBER) WSH_DELIVERY_AS SIGNMENTS.PARENT_DELIVERY_DETAIL_ID (To be called as PARENT_DELIVERY_DETAIL_NUMBER) HEIGHT – From mtl_system_items LENGTH - From mtl_system_items WEIGHT- WSH_DELIVERY_DETAILS. GROSS_WEIGHT WSH_DELIVERY_DETAILS.NET_WEIGHT VOLUME – WSH_DELIVERY_DETAILS.VOLUME WIDTH- From mtl_system_items LOT- WSH_DELIVERY_DETAILS.LOT_NUMBER SUBLOT-WSH_DELIVERY_DETAILS.SUB LOT_NUMBER WSH_DELIVERY_DETAILS. REVISION	User area for additional information on the container

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.USERA REA (continued)		WSH_DELIVERY_DE TAILS.SERIAL NUMBER	
		WSH_DELIVERY_DE TAILS.TO_SERIAL NUMBER	
		WSH_DELIVERY_DE TAILS.SHIP_FROM_ LOCATION_ID converted to code (To be called as SHIP_FROM_LOCAT ION LOCATION)	
		WSH_DELIVERY_DE TAILS.DELIVER_TO_ LOCATION_ID (To be called as DELIVER_TO_LOCA TION)	
		WSH_DELIVERY_DE TAILS.TRACKING_N UMBER	
		WSH_DELIVERY_DE TAILS.COUNTRY_O F_ORIGIN	
		WSH_TRIP_STOPS.A CTUAL_ARRIVAL_D ATE	
		WSH_DELIVERY_DE TAILS.DATE_SCH EDED	
		WSH_TRIP_STOPS.A CTUAL_DEPARTUR E_DATE	
		WSH_DELIVERY_DE TAILS.SHIP_METHO D_CODE	

Field Name	Required?	Transaction Table.Column	Description
CONTAINER.USERA REA (continued)		WSH_DELIVERY_DE TAILS.FREIGHT_CL ASS_CAT_ID converted to external value (To be called as FREIGHT_CLASS_C ATEGORY)	
		WSH_DELIVERY_DE TAILS.FREIGHT_TER MS_CODE converted to external value	
		WSH_DELIVERY_DE TAILS.HAZARD_CL ASS_ID converted to external value (To be called as HAZARD_CLASS or as HAZARD_CLASS_C ODE)	
		WSH_DELIVERY_DE TAILS.FOB_CODE	
		WSH_DELIVERY_DE TAILS.SHIPPING_IN STRUCTIONS	
		WSH_DELIVERY_DE TAILS.PACKING_IN STRUCTIONS	
		WSH_DELIVERY_DE TAILS. Descriptive Flexfields	

Field Name	Required?	Transaction Table.Column	Description
INVITEM.	Optional (B)		Represents the occurrence of a specific quantity of goods inventory packed within the SHIPUNIT or CONTAINER. The inventory item occurrence typically refers to a line item or shipping schedule line of a sales order, purchase order or other business document
INVITEM.ITEM	Required (I)	MTL_SYSTEM_ITEM S_KFV. CONCATENATED_S EGMENTS	Item number
INVITEM.QUANTIT Y (ITEM)	Not used	Not used	Not used
INVITEM.AMOUNT (EXTENDED) (T)	Not used	Not used	Not used
INVITEM.AMOUNT (ITEM) (T)	Not used	Not used	Not used
INVITEM.FRGTCL S	Not used	Not used	Not used
INVITEM.FRGTITE M	Not used	Not used	Not used
INVITEM.HAZRDM ATL	Not used	Not used	Not used
INVITEM.ITEMDESC	Not used	Not used	Not used
INVITEM.ITEMRV	Not used	Not used	Not used

Field Name	Required?	Transaction Table.Column	Description
INVITEM.ITEMRVX	Not used	Not used	Not used
INVITEM.LOTLEVEL 1 - LOTLEVEL2	Not used	Not used	Not used
INVITEM.NOTES 1 - NOTES 9	Not used	Not used	Not used
INVITEM.OWNRSH PCDE	Not used	Not used	Not used
INVITEM.QUANTIT Y (VOLUME)	Not used	Not used	Not used
INVITEM.QUANTIT Y (WEIGHT)	Not used	Not used	Not used
INVITEM.SHIPNOTE S	Not used	Not used	Not used
INVITEM.UPC	Not used	Not used	Not used
INVITEM.USERARE A	Not used	Not used	Not used

The following represents the SHOW_SHIPMENT REQUEST XML data for third party warehousing in Oracle Shipping Execution.

```

<?xml version = '1.0' encoding = 'UTF-8' standalone = 'no'?>
<!-- Oracle eXtensible Markup Language Gateway Server -->
<!DOCTYPE SHOW_SHIPMENT_005 SYSTEM "161_show_shipment_005.dtd">
<SHOW_SHIPMENT_005>
  <CNTROLAREA>
    <BSR>
      <VERB value="SHOW">SHOW</VERB>
      <NOUN value="SHIPMENT">SHIPMENT</NOUN>
      <REVISION value="005">005</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>FTE</COMPONENT>
      <TASK>SSNO</TASK>

    <REFERENCEID>P11510C2.US.ORACLE.COM:oracle.apps.wsh.sup.ssro:TPW346</REFERENCEID>

    <CONFIRMATION>0</CONFIRMATION>
    <LANGUAGE>US</LANGUAGE>
    <CODEPAGE>UTF8</CODEPAGE>
    <AUTHID>APPS</AUTHID>
  </SENDER>
  <DATETIME qualifier="CREATION">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>09</HOUR>
    <MINUTE>44</MINUTE>
    <SECOND>41</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
  </DATETIME>
</CNTROLAREA>
<DATAAREA>
  <SHOW_SHIPMENT>
    <SHIPMENT>
      <DATETIME qualifier="DOCUMENT">
        <YEAR>2005</YEAR>
        <MONTH>04</MONTH>
        <DAY>15</DAY>
        <HOUR>09</HOUR>
        <MINUTE>44</MINUTE>
        <SECOND>41</SECOND>
        <SUBSECOND>0000</SUBSECOND>
        <TIMEZONE>+0000</TIMEZONE>
      </DATETIME>
      <AMOUNT qualifier="DECLAREVAL">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
      </AMOUNT>
      <AMOUNT qualifier="DOCUMENT" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
      </AMOUNT>
      <AMOUNT qualifier="ESTFREIGHT">

```

```

<VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<DATETIME qualifier="CREATION">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>09</HOUR>
    <MINUTE>43</MINUTE>
    <SECOND>57</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="DELIVACT">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="DELIVSCHED">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>23</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>00</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="EARLSTSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="LOADING">

```

```

<YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="NEEDELV">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>23</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>00</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="PROMDELV">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="PROMSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIPSCHEDE">
    <YEAR>2005</YEAR>
    <MONTH>04</MONTH>
    <DAY>15</DAY>
    <HOUR>23</HOUR>
    <MINUTE>59</MINUTE>
    <SECOND>00</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<OPERAMT qualifier="FREIGHT" type="T">
    <VALUE/>

```

```

<NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <UOMVALUE/>
    <UOMNUMDEC/>
    <UOM/>
</OPERAMT>
<QUANTITY qualifier="ESTWEIGHT">
    <VALUE>180</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="LOADINGWT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="NETWEIGHT">
    <VALUE>180</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPUNIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="TOTWEIGHT">
    <VALUE>180</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
    <VALUE>12</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>FT3</UOM>
</QUANTITY>
<TEMPRATURE qualifier="ACTUAL" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="DELIVERY" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="LOADING" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<DOCUMENTID>340</DOCUMENTID>
<SHIPPERID>V1- New York City</SHIPPERID>
<SYNCID>A</SYNCID>

```

```

<CARRIER/>
  <CARRSRVLVL/>
  <COUNTRYDST>US</COUNTRYDST>
  <COUNTRYORG/>
  <DESCRIPTN/>
  <DISTCENTER/>
  <DOCKID/>
  <DOCTYPE>SR</DOCTYPE>
  <DOCUMENTRV/>
  <DROPSHIP/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTTERMS/>
  <HAZRDMATL/>
  <HDRSTATUS/>
  <IMPORTLIC/>
  <LOADPOINT/>
  <NOTES index="1"/>
  <PARTIALSHP/>
  <PRIORITY/>
  <ROUTEID/>
  <SHIPNOTES/>
  <SHIPPOINT>Chattanooga (OPS)</SHIPPOINT>
  <SHIPPRIOR/>
  <SHIPSRLVL/>
  <SPECIALHND/>
  <STAGEPOINT/>
  <TRANSMETHD/>
  <TRANSTERMS/>
  <WAREHOUSE>V1</WAREHOUSE>
  <USERAREA/>
  <PARTNER>
    <NAME index="1">Phoenix Manufacturing</NAME>
    <ONETIME/>
    <PARTNRID>V1- New York City</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER>0</DUNSNUMBER>
    <GLENTITYS/>
    <NAME index="2"/>
    <PARENTID/>
    <PARTNRIDX/>
    <PARTNRRATG/>
    <PARTNRROLE/>
    <PAYMETHOD/>
    <TAXEXEMPT/>
    <TAXID/>
    <TERMID/>
    <USERAREA/>
    <ADDRESS>
      <ADDRLINE index="1">3200 East Camelback</ADDRLINE>
      <ADDRLINE index="2">Suite 255</ADDRLINE>
      <ADDRLINE index="3"/>
      <ADDRLINE index="4"/>
      <ADDRTYPE/>
      <CITY>Phoenix</CITY>
      <COUNTRY>US</COUNTRY>
      <COUNTY/>
    
```



```

<DESCRIPTN/>
    <FAX index="1">602-224-6250</FAX>
    <POSTALCODE>85018</POSTALCODE>
    <REGION/>
    <STATEPROVN>AZ</STATEPROVN>
    <TAXJRSRCTN/>
    <TELEPHONE index="1">602-224-6200</TELEPHONE>
    <URL/>
    <USERAREA/>
</ADDRESS>
<ATTCHREF>
    <DATETIME qualifier="CREATION">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <CMPSNSTYPE/>
    <CMPSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
</ATTCHREF>
<CONTACT>
    <NAME index="1"/>
    <CONTACTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>
</CONTACT>
</PARTNER>
<PARTNER>
    <NAME index="1">Computer Service and Rentals</NAME>
    <ONETIME/>
    <PARTNRID/>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER/>
    <GLENTITYS/>
    <NAME index="2"/>
    <PARENTID/>
    <PARTNRIDX/>

```

```

<PARTNRRATG/>
  <PARTNRROLE/>
  <PAYMETHOD/>
  <TAXEXEMPT/>
  <TAXID/>
  <TERMID/>
  <USERAREA/>
  <ADDRESS>
    <ADDRLINE index="1">301 Summit Hill Drive</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRTYPE/>
    <CITY>Chattanooga</CITY>
    <COUNTRY>US</COUNTRY>
    <COUNTY>Hamilton</COUNTY>
    <DESCRIPTN/>
    <FAX index="1"/>
    <POSTALCODE>37401</POSTALCODE>
    <REGION/>
    <STATEPROVN>TN</STATEPROVN>
    <TAXJRS DCTN/>
    <TELEPHONE index="1">615-522-3600</TELEPHONE>
    <URL/>
    <USERAREA/>
  </ADDRESS>
  <ATTCHREF>
    <DATETIME qualifier="CREATION">
      <YEAR/>
      <MONTH/>
      <DAY/>
      <HOUR/>
      <MINUTE/>
      <SECOND/>
      <SUBSECOND/>
      <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <CMPSNTYPE/>
    <CMPSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
  </ATTCHREF>
  <CONTACT>
    <NAME index="1">Gerry Brown</NAME>
    <CONTCTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1">+1 (703) 8441212</TELEPHONE>
    <USERAREA/>

```

```

</CONTACT>
</PARTNER>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <QUANTITY qualifier="BACKORDERD">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="OPEN">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="ORDERED">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <QUANTITY qualifier="SHIPPED">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="VOLUME">
    <VALUE>2</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>FT3</UOM>
  </QUANTITY>
  <QUANTITY qualifier="WEIGHT">
    <VALUE>30</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
  </QUANTITY>
  <ITEM>AS54888</ITEM>
  <COUNTRYORG/>

```

```

<DISPOSITN/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTITEM/>
  <HAZRDMATL/>
  <IMPORTLIC/>
  <ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
  <ITEMRV/>
  <ITEMRVX/>
  <ITEMX/>
  <LOTLEVEL index="1"/>
  <NOTES index="1"/>
  <OWNRSHPCDE>Destination</OWNRSHPCDE>
  <PACKING/>
  <PARTIALSHP>0</PARTIALSHP>
  <SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>

```

```

</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <CURRENCY/>
  <DRCR/>
</AMOUNT>
<QUANTITY qualifier="BACKORDERD">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="OPEN">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="ORDERED">
  <VALUE>1</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Ea</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPPED">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>2</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
  <VALUE>30</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>
<PARTIALSHP>0</PARTIALSHP>

```

```

<SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
  <QUANTITY qualifier="ITEM">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
  </QUANTITY>
  <AMOUNT qualifier="EXTENDED" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
  </AMOUNT>
  <QUANTITY qualifier="BACKORDERD">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <QUANTITY qualifier="OPEN">

```

```

<VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="ORDERED">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Ea</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPPED">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
    <VALUE>2</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
    <VALUE>30</VALUE>
    <NUMOFDEC/>
    <SIGN>+</SIGN>
    <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>
<PARTIALSHP>0</PARTIALSHP>
<SHIPNOTES/>
<SHIPPRIOR/>
<UPC/>
<USERAREA/>
<DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>

```

```

<USERAREA/>
    </DOCUMNTREF>
    <INVDETAIL>
        <QUANTITY qualifier="ITEM">
            <VALUE/>
            <NUMOFDEC/>
            <SIGN/>
            <UOM/>
        </QUANTITY>
        <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
        <DISPOSITN/>
        <LOTLEVEL index="1"/>
        <LOTLEVEL index="2"/>
        <NOTES index="1"/>
        <SERIALNUM/>
        <USERAREA/>
    </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
    <QUANTITY qualifier="ITEM">
        <VALUE>1</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>Ea</UOM>
    </QUANTITY>
    <AMOUNT qualifier="EXTENDED" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="ITEM" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <QUANTITY qualifier="BACKORDERD">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="OPEN">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="ORDERED">
        <VALUE>1</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="SHIPPED">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>

```


<UOM/>

```
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>2</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
  <VALUE>30</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>
<PARTIALSHP>0</PARTIALSHP>
<SHIPNOTES/>
<SHIPPRIOR/>
<UPC/>
<USERAREA/>
<DOCUMNTREF>
  <DOCTYPE>PURCHASEORDER</DOCTYPE>
  <DOCUMENTID/>
  <PARTNRID>Chattanooga (OPS)</PARTNRID>
  <PARTNRTYPE>ShipTo</PARTNRTYPE>
  <DESCRIPTN/>
  <DOCUMENTRV/>
  <LINENUM/>
  <NOTES index="1"/>
  <SCHLINENUM/>
  <SUBLINENUM/>
  <USERAREA/>
</DOCUMNTREF>
<INVDETAIL>
  <QUANTITY qualifier="ITEM">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
  <DISPOSITN/>
  <LOTLEVEL index="1"/>
  <LOTLEVEL index="2"/>
  <NOTES index="1"/>
  <SERIALNUM/>
```

```

<USERAREA/>
    </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
    <QUANTITY qualifier="ITEM">
        <VALUE>1</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>Ea</UOM>
    </QUANTITY>
    <AMOUNT qualifier="EXTENDED" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="ITEM" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <QUANTITY qualifier="BACKORDERD">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="OPEN">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="ORDERED">
        <VALUE>1</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="SHIPPED">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <QUANTITY qualifier="VOLUME">
        <VALUE>2</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>FT3</UOM>
    </QUANTITY>
    <QUANTITY qualifier="WEIGHT">
        <VALUE>30</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>Lbs</UOM>
    </QUANTITY>
</ITEM>AS54888</ITEM>

```

```

<COUNTRYORG/>
    <DISPOSITN/>
    <EXPORTLIC/>
    <FRGHTCLS/>
    <FRGHTITEM/>
    <HAZRDMATL/>
    <IMPORTLIC/>
    <ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
    <ITEMRV/>
    <ITEMRVX/>
    <ITEMX/>
    <LOTLEVEL index="1"/>
    <NOTES index="1"/>
    <OWNRSHPCDE>Destination</OWNRSHPCDE>
    <PACKING/>
    <PARTIALSHP>0</PARTIALSHP>
    <SHIPNOTES/>
    <SHIPPRIOR/>
    <UPC/>
    <USERAREA/>
    <DOCUMNTREF>
        <DOCTYPE>PURCHASEORDER</DOCTYPE>
        <DOCUMENTID/>
        <PARTNRID>Chattanooga (OPS)</PARTNRID>
        <PARTNRTYPE>ShipTo</PARTNRTYPE>
        <DESCRIPTN/>
        <DOCUMENTRV/>
        <LINENUM/>
        <NOTES index="1"/>
        <SCHLINENUM/>
        <SUBLINENUM/>
        <USERAREA/>
    </DOCUMNTREF>
    <INVDETAIL>
        <QUANTITY qualifier="ITEM">
            <VALUE/>
            <NUMOFDEC/>
            <SIGN/>
            <UOM/>
        </QUANTITY>
        <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
        <DISPOSITN/>
        <LOTLEVEL index="1"/>
        <LOTLEVEL index="2"/>
        <NOTES index="1"/>
        <SERIALNUM/>
        <USERAREA/>
    </INVDETAIL>
</SHIPITEM>
<SHIPITEM>
    <QUANTITY qualifier="ITEM">
        <VALUE>1</VALUE>
        <NUMOFDEC/>
        <SIGN>+</SIGN>
        <UOM>Ea</UOM>
    </QUANTITY>
    <AMOUNT qualifier="EXTENDED" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>

```

```

<DRCR/>
</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <CURRENCY/>
  <DRCR/>
</AMOUNT>
<QUANTITY qualifier="BACKORDERD">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="OPEN">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="ORDERED">
  <VALUE>1</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Ea</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIPPED">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>2</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
  <VALUE>30</VALUE>
  <NUMOFDEC/>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<ITEM>AS54888</ITEM>
<COUNTRYORG/>
<DISPOSITN/>
<EXPORTLIC/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<IMPORTLIC/>
<ITEMDESC>Sentinel Standard Desktop</ITEMDESC>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE>Destination</OWNRSHPCDE>
<PACKING/>

```

```

<PARTIALSHP>0</PARTIALSHP>
  <SHIPNOTES/>
  <SHIPPRIOR/>
  <UPC/>
  <USERAREA/>
  <DOCUMNTREF>
    <DOCTYPE>PURCHASEORDER</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>Chattanooga (OPS)</PARTNRID>
    <PARTNRTYPE>ShipTo</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
  <INVDETAIL>
    <QUANTITY qualifier="ITEM">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <DESCRIPTN>Sentinel Standard Desktop</DESCRIPTN>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <LOTLEVEL index="2"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
  </INVDETAIL>
</SHIPITEM>
<ATTCHREF>
  <DATETIME qualifier="CREATION">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
  </DATETIME>
  <QUANTITY qualifier="FILESIZE">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
  </QUANTITY>
  <CMPRSNTYPE/>
  <CMPRSNID/>
  <DESCRIPTN/>
  <FILETYPE/>
  <NOTES index="1"/>
  <TITLE/>
  <FILENAME/>
  <USERAREA/>
</ATTCHREF>
<DOCUMNTREF>

```

```

<DOCTYPE>SR</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>V1</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
  </DOCUMNTREF>
</SHIPMENT>
</SHOW_SHIPMENT>
</DATAAREA>
</SHOW_SHIPMENT_005>

```

Third Party Warehousing INBOUND Sample XML

The following represents the SHOW_SHIPMENT ADVICE XML data for third party warehousing in Oracle Shipping Execution.

```

<?xml version = '1.0' encoding = 'UTF-8' standalone = 'no'?>
<!-- Oracle eXtensible Markup Language Gateway Server -->
<!DOCTYPE SHOW_SHIPMENT_005 SYSTEM "161_show_shipment_005.dtd">
<SHOW_SHIPMENT_005>
  <CNTRLAREA>
    <BSR>
      <VERB value="SHOW">SHOW</VERB>
      <NOUN value="SHIPMENT">SHIPMENT</NOUN>
      <REVISION value="005">005</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID/>
      <COMPONENT>WSH</COMPONENT>
      <TASK>SSNO</TASK>

    <REFERENCEID>HM000A:oracle.apps.wsh.sup.ssro:CMS437</REFERENCEID>
      <CONFIRMATION>0</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>APPS</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2003</YEAR>
      <MONTH>01</MONTH>
      <DAY>15</DAY>
      <HOUR>15</HOUR>
      <MINUTE>18</MINUTE>
      <SECOND>41</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+0000</TIMEZONE>
    </DATETIME>
  </CNTRLAREA>
  <DATAAREA>
    <SHOW_SHIPMENT>
      <SHIPMENT>
        <DATETIME qualifier="DOCUMENT">
          <YEAR>2003</YEAR>
          <MONTH>01</MONTH>
          <DAY>15</DAY>
          <HOUR>15</HOUR>
          <MINUTE>18</MINUTE>
          <SECOND>40</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+0000</TIMEZONE>
        </DATETIME>
        <AMOUNT qualifier="DECLAREVAL">
          <VALUE/>
          <NUMOFDEC/>
          <SIGN/>
          <CURRENCY/>
          <DRCR/>
        </AMOUNT>
        <AMOUNT qualifier="DOCUMENT" type="T">
          <VALUE/>
          <NUMOFDEC/>
          <SIGN/>
          <CURRENCY/>
          <DRCR/>
        </AMOUNT>
        <AMOUNT qualifier="ESTFREIGHT">
          <VALUE/>

```

```

<NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<AMOUNT qualifier="ITEM" type="T">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<DATETIME qualifier="CREATION">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>54</MINUTE>
    <SECOND>57</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="DELIVACT">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="DELIVSCHED">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>10</MINUTE>
    <SECOND>17</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="EARLSTSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="LOADING">
    <YEAR/>

```



```

<MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="NEEDDELV">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>10</MINUTE>
    <SECOND>17</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<DATETIME qualifier="PROMDELV">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="PROMSHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIP">
    <YEAR/>
    <MONTH/>
    <DAY/>
    <HOUR/>
    <MINUTE/>
    <SECOND/>
    <SUBSECOND/>
    <TIMEZONE/>
</DATETIME>
<DATETIME qualifier="SHIPSCHEDE">
    <YEAR>2003</YEAR>
    <MONTH>01</MONTH>
    <DAY>15</DAY>
    <HOUR>14</HOUR>
    <MINUTE>10</MINUTE>
    <SECOND>17</SECOND>
    <SUBSECOND>0000</SUBSECOND>
    <TIMEZONE>+0000</TIMEZONE>
</DATETIME>
<OPERAMT qualifier="FREIGHT" type="T">
    <VALUE/>
    <NUMOFDEC/>

```

```

<SIGN/>
    <CURRENCY/>
    <UOMVALUE/>
    <UOMNUMDEC/>
    <UOM/>
</OPERAMT>
<QUANTITY qualifier="ESTWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="LOADINGWT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="NETWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="SHIPUNIT">
    <VALUE>1</VALUE>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="TOTWEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<TEMPRATURE qualifier="ACTUAL" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="DELIVERY" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<TEMPRATURE qualifier="LOADING" type="FAHRENHEIT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
</TEMPRATURE>
<DOCUMENTID>367-I</DOCUMENTID>
<SHIPPERID>M3- Dallas</SHIPPERID>
<SYNCID>A</SYNCID>
<CARRIER>WSH-DHL</CARRIER>

```

```

<CARRSRVLVL>1DM</CARRSRVLVL>
  <COUNTRYDST>US</COUNTRYDST>
  <COUNTRYORG/>
  <DESCRIPTN/>
  <DISTCENTER/>
  <DOCKID/>
  <DOCTYPE>SA</DOCTYPE>
  <DOCUMENTRV/>
  <DROPSHIP/>
  <EXPORTLIC/>
  <FRGHTCLS/>
  <FRGHTTERMS>Paid</FRGHTTERMS>
  <HAZRDMATL/>
  <HDRSTATUS/>
  <IMPORTLIC/>
  <LOADPOINT/>
  <NOTES index="1"/>
  <PARTIALSHP/>
  <PRIORITY/>
  <ROUTEID/>
  <SHIPNOTES/>
  <SHIPPOINT>Chattanooga (OPS)</SHIPPOINT>
  <SHIPPRIOR/>
  <SHIPSRLVL>1DM</SHIPSRLVL>
  <SPECIALHND/>
  <STAGEPOINT/>
  <TRANSMETHD>PARCEL</TRANSMETHD>
  <TRANSTERMS>Destination</TRANSTERMS>
  <WAREHOUSE>M3</WAREHOUSE>

<USERAREA><ORACLE.SHOW_SHIPMENT_005.SHIPMENT.USERAREA><ORACLE.DELIVERY_NAME>16401</ORACLE.DELIVERY_NAME><ORACLE.LOADING_SEQUENCE/><ORACLE.VEHICLE_NUM_PREFIX/><ORACLE.VEHICLE_NUMBER/><ORACLE.ROUTE_ID/><ORACLE.ROUTING_INSTRUCTIONS/><ORACLE.DEPARTURE_SEAL_CODE/><ORACLE.Orig_DOCID>367</ORACLE.Orig_DOCID><ORACLE.CUSTOMER_NAME>Computer Service and Rentals</ORACLE.CUSTOMER_NAME><ORACLE.CUSTOMER_NUMBER>1006</ORACLE.CUSTOMER_NUMBER><ORACLE.WAYBILL/><ORACLE.INTMED_SHIP_TO_LOCATION/><ORACLE.DESCRPTIVE_FLEXFIELD><ORACLE.ATTRIBUTE_CATEGORY/><ORACLE.ATTRIBUTE1/><ORACLE.ATTRIBUTE2/><ORACLE.ATTRIBUTE3/><ORACLE.ATTRIBUTE4/><ORACLE.ATTRIBUTE5/><ORACLE.ATTRIBUTE6/><ORACLE.ATTRIBUTE7/><ORACLE.ATTRIBUTE8/><ORACLE.ATTRIBUTE9/><ORACLE.ATTRIBUTE10/><ORACLE.ATTRIBUTE11/><ORACLE.ATTRIBUTE12/><ORA

CLE.ATTRIBUTE13/><ORACLE.ATTRIBUTE14/><ORACLE.ATTRIBUTE15/></ORACLE.DESCRPTIVE_FLEXFIELD></ORACLE.SHOW_SHIPMENT_005.SHIPMENT.USERAREA></USERAREA>

  <PARTNER>
    <NAME index="1">Dallas Manufacturing</NAME>
    <ONETIME/>
    <PARTNRID>M3- Dallas</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <SYNCIND/>
    <ACTIVE/>
    <CURRENCY/>
    <DESCRIPTN/>
    <DUNSNUMBER>0</DUNSNUMBER>
    <GLENITYS/>
    <NAME index="2"/>
    <PARENTID/>

```

```

<PARTNRIDX/>
    <PARTNRRATG/>
    <PARTNRROLE/>
    <PAYMETHOD/>
    <TAXEXEMPT/>
    <TAXID/>
    <TERMID/>

<USERAREA><ORACLE.SHOW_SHIPMENT_005.SHIPMENT.PARTNER.USERAREA><ORACLE.IN
TMED_SHIP_TO_LOCATION/><ORACLE.POOLED_SHIP_TO_LOCATION/></ORACLE.SHOW_SH
IPMENT_005.SHIPMENT.PARTNER.USERAREA></USERAREA>
    <ADDRESS>
        <ADDRLINE index="1">222 West Las Colinas
Blvd</ADDRLINE>
        <ADDRLINE index="2"/>
        <ADDRLINE index="3"/>
        <ADDRLINE index="4"/>
        <ADDRTYPE/>
        <CITY>Irving</CITY>
        <COUNTRY>US</COUNTRY>
        <COUNTY/>
        <DESCRIPTN/>
        <FAX index="1"/>
        <POSTALCODE>75039</POSTALCODE>
        <REGION/>
        <STATEPROVN/>
        <TAXJRSRCTN/>
        <TELEPHONE index="1">972-868-4457</TELEPHONE>
        <URL/>
        <USERAREA/>
    </ADDRESS>
    <ATTCHREF>
        <DATETIME qualifier="CREATION">
            <YEAR/>
            <MONTH/>
            <DAY/>
            <HOUR/>
            <MINUTE/>
            <SECOND/>
            <SUBSECOND/>
            <TIMEZONE/>
        </DATETIME>
        <QUANTITY qualifier="FILESIZE">
            <VALUE/>
            <NUMOFDEC/>
            <SIGN/>
            <UOM/>
        </QUANTITY>
        <CMPSNTYPE/>
        <CMPSNID/>
        <DESCRIPTN/>
        <FILETYPE/>
        <NOTES index="1"/>
        <TITLE/>
        <FILENAME/>
        <USERAREA/>
    </ATTCHREF>
    <CONTACT>
        <NAME index="1"/>
        <CONTCTTYPE/>
        <DESCRIPTN/>

```

```

<EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>
  </CONTACT>
</PARTNER>
<PARTNER>
  <NAME index="1">Computer Service and Rentals</NAME>
  <ONETIME/>
  <PARTNRID>Chattanooga (OPS)</PARTNRID>
  <PARTNRTYPE>ShipTo</PARTNRTYPE>
  <SYNCIND/>
  <ACTIVE/>
  <CURRENCY/>
  <DESCRIPTN/>
  <DUNSNUMBER/>
  <GLENITYS/>
  <NAME index="2"/>
  <PARENTID/>
  <PARTNRIDX/>
  <PARTNRRATG/>
  <PARTNRROLE/>
  <PAYMETHOD/>
  <TAXEXEMPT/>
  <TAXID/>
  <TERMID/>

  <USERAREA><ORACLE.SHOW_SHIPMENT_005.SHIPMENT.PARTNER.USERAREA><ORACLE.IN
TMED_SHIP_TO_LOCATION/><ORACLE.POOLED_SHIP_TO_LOCATION/></ORACLE.SHOW_SH
IPMENT_005.SHIPMENT.PARTNER.USERAREA></USERAREA>
  <ADDRESS>
    <ADDRLINE index="1">301 Summit Hill Drive</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRTYPE/>
    <CITY>Chattanooga</CITY>
    <COUNTRY>US</COUNTRY>
    <COUNTY>Hamilton</COUNTY>
    <DESCRIPTN/>
    <FAX index="1"/>
    <POSTALCODE>37401</POSTALCODE>
    <REGION/>
    <STATEPROVN>TN</STATEPROVN>
    <TAXJRSDCTN/>
    <TELEPHONE index="1">615-522-3600</TELEPHONE>
    <URL/>
    <USERAREA/>
  </ADDRESS>
  <ATTCHREF>
    <DATETIME qualifier="CREATION">
      <YEAR/>
      <MONTH/>
      <DAY/>
      <HOUR/>
      <MINUTE/>
      <SECOND/>
      <SUBSECOND/>
      <TIMEZONE/>
    </DATETIME>
  </ATTCHREF>

```

```

<QUANTITY qualifier="FILESIZE">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<CMPRSNTYPE/>
<CMPRSNID/>
<DESCRIPTN/>
<FILETYPE/>
<NOTES index="1"/>
<TITLE/>
<FILENAME/>
<USERAREA/>
</ATTCHREF>
<CONTACT>
    <NAME index="1"/>
    <CONTCTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>
    <NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>
</CONTACT>
</PARTNER>
<ATTCHREF>
    <DATETIME qualifier="CREATION">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <CMPRSNTYPE/>
    <CMPRSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
</ATTCHREF>
<CHARGE>
    <OPERAMT qualifier="EXTENDED" type="T">
        <VALUE>75</VALUE>
        <NUMOFDEC>1</NUMOFDEC>
        <SIGN>+</SIGN>
        <CURRENCY>USD</CURRENCY>
        <UOMVALUE>D</UOMVALUE>
        <UOMNUMDEC/>
        <UOM/>
    </OPERAMT>
</CHARGE>

```

```

</OPERAMT>
    <CHARGEID>Handling</CHARGEID>
    <CHARGETYPE>Handling</CHARGETYPE>
    <CHGLINENUM/>
    <DESCRIPTN/>

<USERAREA><ORACLE.SHOW_SHIPMENT_005.SHIPMENT.CHARGE.USERAREA><ORACLE.DES
CRIPTIVE_FLEXFIELD><ORACLE.ATTRIBUTE_CATEGORY/><ORACLE.ATTRIBUTE1/><ORAC
LE.ATTRIBUTE2/><ORACLE.ATTRIBUTE3/><ORACLE.ATTRIBUTE4/><ORACLE.ATTRIBUTE
5/><ORACLE.ATTRIBUTE6/><ORACLE.ATTRIBUTE7/><ORACLE.ATTRIBUTE8/><ORACLE.A
TTRIBUTE9/><ORACLE.ATTRIBUTE10/><ORACLE.ATTRIBUTE11/><ORACLE.ATTRIBUTE12
/><ORACLE.ATTRIBUTE13/><ORACLE.ATTRIBUTE14/><ORACLE.ATTRIBUTE15/></ORACL
E.DESCRPTIVE_FLEXFIELD></ORACLE.SHOW_SHIPMENT_005.SHIPMENT.CHARGE.USERA
REA></USERAREA>
    <DISTRIBUTN>
        <OPERAMT qualifier="EXTENDED" type="T">
            <VALUE/>
            <NUMOFDEC/>
            <SIGN/>
            <CURRENCY/>
            <UOMVALUE/>
            <UOMNUMDEC/>
            <UOM/>
        </OPERAMT>
        <BUSNAREA/>
        <COSTCENTER/>
        <DEPARTMENT/>
        <DIVISION/>
        <ELEMENT index="1"/>
        <FUND/>
        <GEOGRAPHY/>
        <GLENTITYS/>
        <GLNOMACCT/>
        <PROFITCTR/>
        <PROJECT/>
        <UNIT/>
        <WAREHOUSE/>
        <USERAREA/>
    </DISTRIBUTN>
</CHARGE>
<DOCUMNTREF>
    <DOCTYPE>SR</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>M3</PARTNRID>
    <PARTNRTYPE>ShipFrom</PARTNRTYPE>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
</DOCUMNTREF>
</SHIPMENT>
<SHIPUNIT>
    <AMOUNT qualifier="DECLAREVAL">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>

```

```

</AMOUNT>
    <AMOUNT qualifier="FREIGHT" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="ITEM" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="TOTAL" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <DATETIME qualifier="DELIVACT">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <DATETIME qualifier="DELIVSCHED">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <DATETIME qualifier="LOADING">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <DATETIME qualifier="SHIP">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>

```



```

</DATETIME>
<OPERAMT qualifier="FREIGHT" type="T">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <CURRENCY/>
  <UOMVALUE/>
  <UOMNUMDEC/>
  <UOM/>
</OPERAMT>
<QUANTITY qualifier="ESTWEIGHT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="HEIGHT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="LENGTH">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="LOADINGWT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="NETWEIGHT">
  <VALUE>1</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="TOTWEIGHT">
  <VALUE>1001</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="VOLUME">
  <VALUE>1</VALUE>
  <NUMOFDEC>0</NUMOFDEC>
  <SIGN>+</SIGN>
  <UOM>FT3</UOM>
</QUANTITY>
<QUANTITY qualifier="WIDTH">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<TEMPERATURE qualifier="ACTUAL" type="FAHRENHEIT">
  <VALUE/>
  <NUMOFDEC/>

```

```

<SIGN/>
    </TEMPRATURE>
    <TEMPRATURE qualifier="DELIVERY" type="FAHRENHEIT">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
    </TEMPRATURE>
    <TEMPRATURE qualifier="LOADING" type="FAHRENHEIT">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
    </TEMPRATURE>
    <CARRIER>WSH-DHL</CARRIER>
    <SHIPPERID>M3- Dallas</SHIPPERID>
    <TRACKINGID>1Z4444441500000227</TRACKINGID>
    <CARRSRVLVL>1DM</CARRSRVLVL>
    <CONTAINRID>118081</CONTAINRID>
    <CONTRSEAL/>
    <CONTRNRTYPE>ALC</CONTRNRTYPE>
    <COUNTRYDST/>
    <COUNTRYORG/>
    <DESCRIPTN/>
    <EXPORTLIC/>
    <FRGHTCLS/>
    <FRGHTITEM/>
    <FRGHTTERMS>Paid</FRGHTTERMS>
    <HAZRDMATL/>
    <IMPORTLIC/>
    <LOADPOINT/>
    <NOTES index="1"/>
    <SHIPMATLID>A1 DOM</SHIPMATLID>
    <SHIPNOTES/>
    <SHIPPOINT>Chattanooga (OPS)</SHIPPOINT>
    <SHPSRVVLVL>1DM</SHPSRVVLVL>
    <SHPMTLDESC>A1 Domestic Airline Container</SHPMTLDESC>
    <SHPUNITSEQ/>
    <SHPUNITSTS/>
    <SHPUNITTOT/>
    <SPECIALHND/>
    <STAGEPOINT/>
    <TRANSMETHD>PARCEL</TRANSMETHD>
    <TRANSTERMS>Destination</TRANSTERMS>

<USERAREA><ORACLE.SHOW_SHIPMENT_005.SHIPUNIT.USERAREA><ORACLE.DELIVERY_
DETAIL_NUMBER>118081</ORACLE.DELIVERY_DETAIL_NUMBER><ORACLE.LOT_NUMBER/><
ORACLE.SUBLOT_NUMBER/><ORACLE.REVISION/><ORACLE.SERIAL_NUMBER/><ORACLE.T
O_SERIAL_NUMBER/><ORACLE.PACKING_INSTRUCTIONS/><ORACLE.DESCRPTIVE_FLEXF
IELD><ORACLE.ATTRIBUTE_CATEGORY/><ORACLE.ATTRIBUTE1/><ORACLE.ATTRIBUTE2/
><ORACLE.ATTRIBUTE3/><ORACLE.ATTRIBUTE4/><ORACLE.ATTRIBUTE5/><ORACLE.ATT
RIBUTE6/><ORACLE.ATTRIBUTE7/><ORACLE.ATTRIBUTE8/><ORACLE.ATTRIBUTE9/><OR
ACLE.ATTRIBUTE10/><ORACLE.ATTRIBUTE11/><ORACLE.ATTRIBUTE12/><ORACLE.ATTR
IBUTE13/><ORACLE.ATTRIBUTE14/><ORACLE.ATTRIBUTE15/></ORACLE.DESCRPTIVE_
FLEXFIELD></ORACLE.SHOW_SHIPMENT_005.SHIPUNIT.USERAREA></USERAREA>
    <PARTNER>
        <NAME index="1">Computer Service and Rentals</NAME>
        <ONETIME/>
        <PARTNRID>Chattanooga (OPS)</PARTNRID>
        <PARTNRTYPE>ShipTo</PARTNRTYPE>
        <SYNCIND/>
        <ACTIVE/>
        <CURRENCY/>

```

```

<DESCRIPTN/>
  <DUNSNUMBER/>
  <GLENTITYS/>
  <NAME index="1"/>
  <PARENTID/>
  <PARTNRIDX/>
  <PARTNRRATG/>
  <PARTNRROLE/>
  <PAYMETHOD/>
  <TAXEXEMPT/>
  <TAXID/>
  <TERMID/>
  <USERAREA/>
  <ADDRESS>
    <ADDRLINE index="1">301 Summit Hill Drive</ADDRLINE>
    <ADDRTYPE/>
    <CITY>Chattanooga</CITY>
    <COUNTRY>US</COUNTRY>
    <COUNTY/>
    <DESCRIPTN/>
    <FAX index="1"/>
    <POSTALCODE>37401</POSTALCODE>
    <REGION/>
    <STATEPROVN>TN</STATEPROVN>
    <TAXJRS DCTN/>
    <TELEPHONE index="1">615-522-3600</TELEPHONE>
    <URL/>
    <USERAREA/>
  </ADDRESS>
  <ATTCHREF>
    <DATETIME qualifier="CREATION">
      <YEAR/>
      <MONTH/>
      <DAY/>
      <HOUR/>
      <MINUTE/>
      <SECOND/>
      <SUBSECOND/>
      <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
      <VALUE/>
      <NUMOFDEC/>
      <SIGN/>
      <UOM/>
    </QUANTITY>
    <CMPSRNTYPE/>
    <CMPSRNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
  </ATTCHREF>
  <CONTACT>
    <NAME index="1"/>
    <CONTCTTYPE/>
    <DESCRIPTN/>
    <EMAIL/>
    <FAX index="1"/>

```

```

<NAME index="1"/>
    <TELEPHONE index="1"/>
    <USERAREA/>
</CONTACT>
</PARTNER>
<ATTCHREF>
    <DATETIME qualifier="CREATION">
        <YEAR/>
        <MONTH/>
        <DAY/>
        <HOUR/>
        <MINUTE/>
        <SECOND/>
        <SUBSECOND/>
        <TIMEZONE/>
    </DATETIME>
    <QUANTITY qualifier="FILESIZE">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <CMPRSNTYPE/>
    <CMPRSNID/>
    <DESCRIPTN/>
    <FILETYPE/>
    <NOTES index="1"/>
    <TITLE/>
    <FILENAME/>
    <USERAREA/>
</ATTCHREF>
<DOCUMNTREF>
    <DOCTYPE/>
    <DOCUMENTID/>
    <PARTNRID/>
    <PARTNRTYPE/>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
</DOCUMNTREF>
<INVITEM>
    <QUANTITY qualifier="ITEM">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <AMOUNT qualifier="EXTENDED" type="T">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <CURRENCY/>
        <DRCR/>
    </AMOUNT>
    <AMOUNT qualifier="ITEM" type="T">
        <VALUE/>
        <NUMOFDEC/>

```

```

<SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<QUANTITY qualifier="VOLUME">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<QUANTITY qualifier="WEIGHT">
    <VALUE/>
    <NUMOFDEC/>
    <SIGN/>
    <UOM/>
</QUANTITY>
<ITEM/>
<COUNTRYORG/>
<FRGHTCLS/>
<FRGHTITEM/>
<HAZRDMATL/>
<ITEMDESC/>
<ITEMRV/>
<ITEMRVX/>
<ITEMX/>
<LOTLEVEL index="1"/>
<NOTES index="1"/>
<OWNRSHPCDE/>
<SHIPNOTES/>
<UPC/>
<USERAREA/>
<DOCUMNTREF>
    <DOCTYPE/>
    <DOCUMENTID/>
    <PARTNRID/>
    <PARTNRTYPE/>
    <DESCRIPTN/>
    <DOCUMENTRV/>
    <LINENUM/>
    <NOTES index="1"/>
    <SCHLINENUM/>
    <SUBLINENUM/>
    <USERAREA/>
</DOCUMNTREF>
<INVDETAIL>
    <QUANTITY qualifier="ITEM">
        <VALUE/>
        <NUMOFDEC/>
        <SIGN/>
        <UOM/>
    </QUANTITY>
    <DESCRIPTN/>
    <DISPOSITN/>
    <LOTLEVEL index="1"/>
    <NOTES index="1"/>
    <SERIALNUM/>
    <USERAREA/>
</INVDETAIL>
</INVITEM>
</SHIPUNIT>
</SHOW_SHIPMENT>

```

```
</DATAAREA>  
</SHOW_SHIPMENT_005>
```

Using XML Messaging with International Trade Management

Oracle Shipping Execution enables you to send and receive XML messages to and from third party international trade management (ITM) systems. ITM systems provide up-to-date information on country specific rules and regulations for importing and exporting goods.

Oracle International Trade Management Adapter is an open interface that enables the communication of Oracle Applications with third party ITM systems

Before using XML messaging with ITM, you must:

- Set up Oracle XML Gateway
- Set up Oracle ITM Adapter
- Set up Oracle XML Transport Agent

The XML transactions used for ITM are:

- Item Synchronization Request
- Party Synchronization Request
- Order Compliance Request
- Delivery Compliance Request
- Export Compliance Response

ItemSynchronization Request Transaction for ITM

Item Synchronization Request transaction is a modified version of the Open Applications Group (OAG) document type definition (DTD). This DTD is used to send item information from Oracle Shipping Execution to a third party international trade management system, which is used for classification of the item. The ITM system does not send a response to the ItemSynchronization Request.

This transaction contains information such as:

- Item number
- Item description
- Hazard classification
- Unit of measure

PartySynchronization Request Transaction for ITM

Party Synchronization Request is a modified version of the Open Applications Group (OAG) document type definition (DTD). This DTD is used to send export screening data elements from Oracle Shipping execution to a third party international trade management system. The ITM system does not send a response to the PartySynchronization Request.

This transaction contains information such as:

- Customer number
- Supplier number
- Federal tax ID number
- State tax ID number
- Operating unit

OrderCompliance Request for ITM

Order Compliance Request transaction is a modified version of the Open Applications Group (OAG) document type definition (DTD). This DTD is used to send order and order line information to a third party international trade management system for verification of compliance. Each order line is evaluated for export compliance. Upon completion of the evaluation, the ITM system sends verification information (pass or fail) back to Oracle Shipping Execution.

This transaction contains information such as:

- Order number
- Order type
- Customer PO number
- Item type

DeliveryCompliance Request Transaction for ITM

Delivery Compliance Request transaction is a modified version of the Open Applications Group (OAG) document type definition (DTD) show_shipment_005. This DTD is used by Oracle Shipping Execution to send delivery compliance information to the third party international trade management system.

This transaction contains information such as:

- Delivery name
- Customer
- Ship to address
- Ship to contact
- Requested pick up date and time
- Requested delivery date and time
- Weight
- Carrier
- Service level

ExportCompliance Response Transaction for ITM

Export Compliance Response transaction is a modified version of the Open Applications Group (OAG) document type definition (DTD) show_shipment_005. This DTD is used to receive export compliance status information (order and order line) from a third party international trade management system to Oracle Shipping Execution. This transaction is sent in response to the Delivery Compliance Request and Order Compliance Request transactions.

This transaction contains information such as:

- Delivery name
- Customer
- Ship to address
- Ship to contact
- Requested pick up date and time
- Requested delivery date and time
- Weight
- Carrier
- Service level

Overview of XML and DTD for ITM

The following contains the DTD and a sample XML used for each of the international trade management XML transactions in Oracle Shipping Execution . Also, a table is detailed for each ITM XML transaction, which displays all required element/attributes, a description of each element, and the Oracle table and column for each element/attribute, where applicable.

ITM DTDs

The following is the Item Synchronization Request DTD used for ITM in Oracle Shipping Execution.

```

<!ENTITY % TYPEREQ "TYPE CDATA #REQUIRED">
<!ENTITY % LABELREQ "LABEL CDATA #REQUIRED">
<!ELEMENT ITEM_SYNC (CNTROLAREA, DATAAREA+)>
<!ELEMENT CNTROLAREA (BSR, SENDER, DATETIME)>
<!ELEMENT BSR (VERB, NOUN, REVISION)>
<!ELEMENT VERB (#PCDATA)>
<!ELEMENT NOUN (#PCDATA)>
<!ELEMENT REVISION (#PCDATA)>
<!ELEMENT SENDER (LOGICALID, COMPONENT, TASK, REFERENCEID, CONFIRMATION,
LANGUAGE, CODEPAGE, AUTHID)>
<!ELEMENT LOGICALID (#PCDATA)>
<!ELEMENT COMPONENT (#PCDATA)>
<!ELEMENT TASK (#PCDATA)>
<!ELEMENT REFERENCEID (#PCDATA)>
<!ELEMENT CONFIRMATION (#PCDATA)>
<!ELEMENT AUTHID (#PCDATA)>
<!ELEMENT LANGUAGE (#PCDATA)>
<!ELEMENT CODEPAGE (#PCDATA)>
<!ELEMENT TIMEZONE (#PCDATA)>
<!ELEMENT DATETIME (YEAR, MONTH, DAY, HOUR, MINUTE, SECOND, SUBSECOND,
TIMEZONE)>
<!ATTLIST DATETIME
    qualifier CDATA #REQUIRED
    type CDATA #IMPLIED
    index CDATA #IMPLIED
>
<!ELEMENT YEAR (#PCDATA)>
<!ELEMENT MONTH (#PCDATA)>
<!ELEMENT DAY (#PCDATA)>
<!ELEMENT HOUR (#PCDATA)>
<!ELEMENT MINUTE (#PCDATA)>
<!ELEMENT SECOND (#PCDATA)>
<!ELEMENT SUBSECOND (#PCDATA)>
<!ELEMENT DATAAREA (ITEM+)>
<!ELEMENT ITEM (ITEM_ID, ITEM_CODE?, DESCRIPTION, HAZARD_CLASS?,
ORGANIZATION_ID?, SRC_ORGANIZATION_ID?,
ITEM_TYPE?, UNIT_OF_MEASURE?, ITEM_VALUE?, CURRENCY?, COUNTRY*,
USERAREA?)>
<!ELEMENT ITEM_ID (#PCDATA)>
<!ELEMENT ITEM_CODE (#PCDATA)>
<!ELEMENT DESCRIPTION (#PCDATA)>
<!ELEMENT HAZARD_CLASS (#PCDATA)>
<!ELEMENT ORGANIZATION_ID (#PCDATA)>
<!ELEMENT SRC_ORGANIZATION_ID (#PCDATA)>
<!ELEMENT ITEM_TYPE (#PCDATA)>
<!ELEMENT UNIT_OF_MEASURE (#PCDATA)>
<!ELEMENT ITEM_VALUE (#PCDATA)>
<!ELEMENT CURRENCY (#PCDATA)>
<!ELEMENT COUNTRY (#PCDATA)>
<!ELEMENT USERAREA ANY>
<!ELEMENT ORACLE.ITEM_SYNC.ITEM.USERAREA (ORACLE.LONG_DESCRIPTION?,
ORACLE.OPERATING_UNIT?, ORACLE.PARAMETERS*)>
<!ELEMENT ORACLE.LONG_DESCRIPTION (#PCDATA)>
<!ELEMENT ORACLE.OPERATING_UNIT (#PCDATA)>
<!ELEMENT ORACLE.PARAMETERS (ORACLE.PARAMETER*)>
<!ATTLIST ORACLE.PARAMETERS
    TYPE (VENDOR | SERVICE | DATA) #REQUIRED
>
<!ELEMENT ORACLE.PARAMETER EMPTY>
<!ATTLIST ORACLE.PARAMETER
    INDEX CDATA #REQUIRED

```

```

>
<!ATTLIST ORACLE.PARAMETER
  NAME CDATA #IMPLIED
  VALUE CDATA #REQUIRED
> d

```

The following is the Party Synchronization Request DTD used for ITM in Oracle Shipping Execution.

```

sync_customerdt
<!-- $Revision: 7.2.1 $
      $Date: 31 October 2001 $
      Open Applications Group DTD
      Copyright 1998-2001, All Rights Reserved

      $Name: 007_sync_customer_005.dtd $

```

Structure Overview

```

SYNC_CUSTOMER (PARTNER)
  PARTNER ()

```

Notes:

```

- >
<!-- ===== - >
<!ENTITY % RESOURCES SYSTEM "oagis_resources.dtd">
%RESOURCES;
<!-- ===== - >
<!ELEMENT SYNC_CUSTOMER_005 (CNTROLAREA, DATAAREA+)>
<!ATTLIST VERB
  value CDATA #FIXED "SYNC"
>
<!ATTLIST NOUN
  value CDATA #FIXED "CUSTOMER"
>
<!ATTLIST REVISION
  value CDATA #FIXED "005"
>
<!ELEMENT DATAAREA (SYNC_CUSTOMER)>
<!ELEMENT SYNC_CUSTOMER (PARTNER)>

```

The following is the Order Compliance Request DTD used for ITM in Oracle Shipping Execution.

```

<!--
  $Author - Joe Smith$
  $Date: 10 April 2003 $
  Oracle
  All Rights Reserved
-->
<!--ELEMENT request (message_id?,admin, parameters?, transaction_date?,
addl_info?, (dp_request | order_line)+)>
  <!--ELEMENT message_id (#PCDATA)>
  <!--ELEMENT admin (subscriber_id, password, request_generator, revision)>
    <!--ELEMENT subscriber_id (#PCDATA)>
    <!--ELEMENT password (#PCDATA)>
    <!--ELEMENT request_generator (#PCDATA)>
    <!--ELEMENT revision (#PCDATA)>
  <!--ELEMENT parameters (parameter+)>
    <!--ATTLIST parameter name CDATA #REQUIRED>
    <!--ATTLIST parameter value CDATA #REQUIRED>
  <!--ELEMENT transaction_date (#PCDATA)>
  <!--ELEMENT addl_info (info*)>
    <!--ELEMENT info (#PCDATA) >
    <!--ATTLIST info index CDATA #REQUIRED>
    <!--ATTLIST info name CDATA #REQUIRED>
  <!--ELEMENT order_line (transaction_key, order_number?, order_type?,
org_id?, cust_po_num?,
    transactional_curr_code?,
    conversion_type_code?, conversion_rate?, ordered_date?,
shipping_method_code?,
    request_date?, freight_terms_code?, paymet_terms_name?,
payment_term_id?, ordered_quantity?,
    unit_selling_price?, line_number?, unit_list_price?,
parameters?,
    item_id, party+ )>
    <!--ELEMENT transaction_key (#PCDATA)>
    <!--ELEMENT order_number (#PCDATA)>
    <!--ELEMENT order_type (#PCDATA)>
    <!--ELEMENT org_id (#PCDATA)>
    <!--ELEMENT cust_po_num (#PCDATA)>
    <!--ELEMENT transactional_curr_code (#PCDATA)>
    <!--ELEMENT conversion_type_code (#PCDATA)>
    <!--ELEMENT conversion_rate (#PCDATA)>
    <!--ELEMENT ordered_date (#PCDATA)>
    <!--ELEMENT shipping_method_code (#PCDATA)>
    <!--ELEMENT request_date (#PCDATA)>
    <!--ELEMENT freight_terms_code (#PCDATA)>
    <!--ELEMENT paymet_terms_name (#PCDATA)>
    <!--ELEMENT payment_term_id (#PCDATA)>
    <!--ELEMENT ordered_quantity (#PCDATA)>
    <!--ELEMENT unit_selling_price (#PCDATA)>
    <!--ELEMENT line_number (#PCDATA)>
    <!--ELEMENT unit_list_price (#PCDATA)>
    <!--ELEMENT item_id (#PCDATA)>
    <!--ELEMENT dp_request (ship_from_country, ship_to_country?, parameters,
party+ )>
    <!--ELEMENT ship_from_country (#PCDATA)>
    <!--ELEMENT ship_to_country (#PCDATA)>

  <!-- Note: that ONLY PARTY ID is mandatory. Hence only ID would be
used for Normal
    parties and the details would be filled up for One-Time parties -->

  <!--ELEMENT party (party_id, type?, name?, address1?, address2?,

```

```

address3?, address4?, address5?,city?,
  state_or_province?, country, postal_code?, contact_name?, email?,
phone?, fax?,
  web?, addl_info?)>
  <!ELEMENT party_id (#PCDATA)>
  <!ELEMENT type (#PCDATA)>
  <!ELEMENT name (#PCDATA)>
  <!ELEMENT address1 (#PCDATA)>
  <!ELEMENT address2 (#PCDATA)>
  <!ELEMENT address3 (#PCDATA)>
  <!ELEMENT address4 (#PCDATA)>
  <!ELEMENT address5 (#PCDATA)>
  <!ELEMENT city (#PCDATA)>
  <!ELEMENT state_or_province (#PCDATA)>
  <!ELEMENT country (#PCDATA)>
  <!ELEMENT postal_code (#PCDATA)>
  <!ELEMENT contact_name (#PCDATA)>
  <!ELEMENT email (#PCDATA)>
  <!ELEMENT phone (#PCDATA)>
  <!ELEMENT fax (#PCDATA)>
  <!ELEMENT web (#PCDATA)>

```

The following is the Delivery Compliance Request DTD used for ITM in Oracle Shipping Execution.

```

<! - ===== - >
<!ENTITY % RESOURCES SYSTEM "oagis_resources.dtd">
%RESOURCES;
<! - ===== - >
<!ELEMENT SHOW_SHIPMENT_005 (CNTROLAREA, DATAAREA+)>
<!ATTLIST VERB
  value CDATA #FIXED "SHOW">
<!ATTLIST NOUN
  value CDATA #FIXED "SHIPMENT">
<!ATTLIST REVISION
  value CDATA #FIXED "005">
<!ELEMENT DATAAREA (SHOW_SHIPMENT)>
<!ELEMENT SHOW_SHIPMENT (SHIPMENT, SHIPUNIT*)>
<!ELEMENT SHIPMENT ((%DATETIME.DOCUMENT;)?, (%AMOUNT.DECLAREVAL;)?,
(%AMOUNT.DOCUMENT.T;)?, (%AMOUNT.ESTFREIGHT;)?, (%AMOUNT.FREIGHT.T;)?,
(%AMOUNT.ITEM.T;)?, (%DATETIME.CREATION;)?, (%DATETIME.DELIVACT;)?,
(%DATETIME.DELIVSCHED;)?, (%DATETIME.EARLSTSHIP;)?,
(%DATETIME.LOADING;)?, (%DATETIME.NEEDDELV;)?, (%DATETIME.PROMDELV;)?,
(%DATETIME.PROMSHIP;)?, (%DATETIME.SHIP;)?, (%DATETIME.SHIPSCHED;)?,
(%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?,
(%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
(%QUANTITY.SHIPUNIT;)?, (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?,
(%TEMPERATURE.ACTUAL;)?, (%TEMPERATURE.DELIVERY;)?,
(%TEMPERATURE.LOADING;)?, DOCUMENTID, SHIPPERID, SYNCID, CARRIER?,
CARRSRVLVL?, COUNTRYDST?, COUNTRYORG?, DESCRIPTN?, DISTCENTER?, DOCKID?,
DOCTYPE?, DOCUMENTTRV?, DROPSHIP?, EXPORTLIC?, FRGHTCLS?, FRGHTTERMS?,
HAZRDMATL?, HDRSTATUS?, IMPORTLIC?, LOADPOINT?, NOTES*, PARTIALSHP?,
PRIORITY?, ROUTEID?, SHIPNOTES?, SHIPPOINT?, SHIPPRIOR?, SHIPSRVLVL?,
SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, WAREHOUSE?,
USERAREA?, PARTNER+, SHIPITEM*, ATTCHREF*, CHARGE*, DOCUMNTREF*)>
<!ELEMENT SHIPITEM ((%QUANTITY.ITEM;)?, (%AMOUNT.EXTENDED.T;)?,
(%AMOUNT.ITEM.T;)?, (%QUANTITY.BACKORDERD;)?, (%QUANTITY.OPEN;)?,
(%QUANTITY.ORDERED;)?, (%QUANTITY.SHIPPED;)?, (%QUANTITY.VOLUME;)?,
(%QUANTITY.WEIGHT;)?, ITEM, COUNTRYORG?, DISPOSITN?, EXPORTLIC?,
FRGHTCLS?, FRGHTITEM?, HAZRDMATL?, IMPORTLIC?, ITEMDESC?, ITEMRV?,
ITEMRVX?, ITEMX?, LOTLEVEL*, NOTES*, OWNRSHPCDE?, PACKING?, PARTIALSHP?,
SHIPNOTES?, SHIPPRIOR?, UPC?, USERAREA?, DOCUMNTREF*, INVDETAIL*,
CHARGE*)
>
<!ELEMENT INVDETAIL ((%QUANTITY.ITEM;)?, DESCRIPTN?, DISPOSITN?,
LOTLEVEL*, NOTES*, SERIALNUM?, USERAREA?)
>
<!ELEMENT SHIPUNIT ((%AMOUNT.DECLAREVAL;)?, (%AMOUNT.FREIGHT.T;)?,
(%AMOUNT.ITEM.T;)?, (%AMOUNT.TOTAL.T;)?, (%DATETIME.DELIVACT;)?,
(%DATETIME.DELIVSCHED;)?, (%DATETIME.LOADING;)?, (%DATETIME.SHIP;)?,
(%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?, (%QUANTITY.HEIGHT;)?,
(%QUANTITY.LENGTH;)?, (%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
(%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?, (%QUANTITY.WIDTH;)?,
(%TEMPERATURE.ACTUAL;)?, (%TEMPERATURE.DELIVERY;)?,
(%TEMPERATURE.LOADING;)?, CARRIER, SHIPPERID, TRACKINGID, CARRSRVLVL?,
CONTAINRID?, CONTRNSEAL?, CONTRNRTYPE?, COUNTRYDST?, COUNTRYORG?,
DESCRIPTN?, EXPORTLIC?, FRGHTCLS?, FRGHTITEM?, FRGHTTERMS?, HAZRDMATL?,
IMPORTLIC?, LOADPOINT?, NOTES*, SHIPMATLID?, SHIPNOTES?, SHIPPOINT?,
SHIPSRVLVL?, SHPMTLDESC?, SHPUNITSEQ?, SHPUNITSTS?, SHPUNITTOT?,
SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, USERAREA?, PARTNER+,
ATTCHREF*, CHARGE*, CONTAINER*, DOCUMNTREF*, INVITEM*)
>
<!ELEMENT CONTAINER (CONTAINRID?, CONTRNSEAL?, CONTRNRTYPE?, DESCRIPTN?,
NOTES*, PACKING?, PACKNGDESC?, PARCNTRID?, SHIPMATLID?, SHPMTLDESC?,
USERAREA?, DOCUMNTREF*, INVITEM*, CHARGE*)
>

```

```

<!ELEMENT INVITEM ((%QUANTITY.ITEM;), (%AMOUNT.EXTENDED.T;)?,
(%AMOUNT.ITEM.T;)?, (%QUANTITY.VOLUME;)?, (%QUANTITY.WEIGHT;)?, ITEM,
COUNTRYORG?, FRGHTCLS?, FRGHTITEM?, HAZRDMATL?, ITEMDESC?, ITEMRV?,
ITEMRVX?, ITEMX?, LOTLEVEL*, NOTES*, OWNRSHPCODE?, SHIPNOTES?, UPC?,
USERAREA?, DOCUMNTREF*, INVDETAIL*)>
@
115.1.11510.1
log
@'mkbranch'
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ ->
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@
115.1.1159.1
log
@mkbranch
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ ->
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@
115.1.1158.1
log
@mkbranch
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.1 2002/03/28 22:47:30
rvishnuv ship $ ->
d6 2
a7 2
    $Revision: 115.1 $
    $Date: 2002/03/28 22:47:30 $
@
115.0
log
@initial version
@
text
@d1 1
a1 1
<! - $Header: 161_show_shipment_005.dtd 115.0 2002/02/15 05:27:39
rvishnuv noship $ ->
d6 2
a7 2
    $Revision: 7.2.1 $
    $Date: 31 October 2001 $

```

```

d11 1
a11 1
    $Name: 161_show_shipment_005.dtd $
d22 1
a22 1
    SHIPMENT ( PARTNER+, SHIPITEM+, ATTCHREF*, CHARGE*, DOCUMNTREF*,)
d59 1
a59 1
<!ELEMENT SHIPMENT ((%DATETIME.DOCUMENT;), (%AMOUNT.DECLAREVAL;)?,
(%AMOUNT.DOCUMENT.T;)?, (%AMOUNT.ESTFREIGHT;)?, (%AMOUNT.FREIGHT.T;)?,
(%AMOUNT.ITEM.T;)?, (%DATETIME.CREATION;)?, (%DATETIME.DELIVACT;)?,
(%DATETIME.DELIVSCHED;)?, (%DATETIME.EARLSTSHIP;)?,
(%DATETIME.LOADING;)?, (%DATETIME.NEEDDELV;)?, (%DATETIME.PROMDELV;)?,
(%DATETIME.PROMSHIP;)?, (%DATETIME.SHIP;)?, (%DATETIME.SHIPSCHED;)?,
(%OPERAMT.FREIGHT.T;)?, (%QUANTITY.ESTWEIGHT;)?,
(%QUANTITY.LOADINGWT;)?, (%QUANTITY.NETWEIGHT;)?,
(%QUANTITY.SHIPUNIT;)?, (%QUANTITY.TOTWEIGHT;)?, (%QUANTITY.VOLUME;)?,
(%TEMPERATURE.ACTUAL;)?, (%TEMPERATURE.DELIVERY;)?,
(%TEMPERATURE.LOADING;)?, DOCUMENTID, SHIPPERID, SYNCID, CARRIER?,
CARRSRVLVL?, COUNTRYDST?, COUNTRYORG?, DESCRIPTN?, DISTCENTER?, DOCKID?,
DOCTYPE?, DOCUMENTTRV?, DROPSHIP?, EXPORTLIC?, FRGHTCLS?, FRGHTTERMS?,
HAZRDMATL?, HDRSTATUS?, IMPORTLIC?, LOADPOINT?, NOTES*, PARTIALSHP?,
PRIORITY?, ROUTEID?, SHIPNOTES?, SHIPPOINT?, SHIPPRIOR?, SHIPSRVLVL?,
SPECIALHND?, STAGEPOINT?, TRANSMETHD?, TRANSTERMS?, WAREHOUSE?,
USERAREA?, PARTNER+, SHIPITEM+, ATTCHREF*, CHARGE*, DOCUMNTREF*)
@

```

The following is the Export Compliance Response DTD used for ITM in Oracle Shipping Execution.


```

<!--
$Author - Joe Smith$
$Date: 16 April 2003 $
Oracle Cooperation
All Rights Reserved
-->
<!--ELEMENT EXPORT_COMPL_RESPONSE ( CNTROLAREA, DATAAREA+)>
  <!--ELEMENT CNTROLAREA (BSR, SENDER, DATETIME)>
    <!--ELEMENT BSR (VERB, NOUN, REVISION)>
      <!--ELEMENT VERB (#PCDATA)>
      <!--ELEMENT NOUN (#PCDATA)>
      <!--ELEMENT REVISION (#PCDATA)>
      <!--ELEMENT SENDER (LOGICALID, COMPONENT, TASK, REFERENCEID,
CONFIRMATION, LANGUAGE, CODEPAGE, AUTHID)>
        <!--ELEMENT LOGICALID (#PCDATA)>
        <!--ELEMENT COMPONENT (#PCDATA)>
        <!--ELEMENT TASK (#PCDATA)>
        <!--ELEMENT REFERENCEID (#PCDATA)>
        <!--ELEMENT CONFIRMATION (#PCDATA)>
        <!--ELEMENT LANGUAGE (#PCDATA)>
        <!--ELEMENT CODEPAGE (#PCDATA)>
        <!--ELEMENT AUTHID (#PCDATA)>
        <!--ELEMENT DATETIME (YEAR, MONTH, DAY, HOUR, MINUTE, SECOND, SUBSECOND,
TIMEZONE)>
          <!--ELEMENT YEAR (#PCDATA)>
          <!--ELEMENT MONTH (#PCDATA)>
          <!--ELEMENT DAY (#PCDATA)>
          <!--ELEMENT HOUR (#PCDATA)>
          <!--ELEMENT MINUTE (#PCDATA)>
          <!--ELEMENT SECOND (#PCDATA)>
          <!--ELEMENT SUBSECOND (#PCDATA)>
          <!--ELEMENT TIMEZONE (#PCDATA)>
        <!--ELEMENT DATAAREA ( MESSAGE+)>
          <!--ELEMENT MESSAGE (TRANSACTION_KEY, TRIGGERING_POINT,STATUS,
EXPORT_COMP_RESULT*, ERROR*)>
            <!--ELEMENT TRANSACTION_KEY (#PCDATA)>
            <!--ELEMENT TRIGGERING_POINT (#PCDATA)>
            <!--ELEMENT STATUS (#PCDATA)>
            <!--ELEMENT EXPORT_COMP_RESULT (TYPE, SUCCESS, DESCRIPTION)>
              <!--ELEMENT TYPE (#PCDATA)>
              <!--ELEMENT SUCCESS (#PCDATA)>
              <!--ELEMENT DESCRIPTION (#PCDATA)>
            <!--ELEMENT ERROR (TYPE, CODE, DESCRIPTION)>
              <!--ELEMENT CODE (#PCDATA)>

```

Sample ITM XML Transactions

ITM Sample Party Synchronization XML

The following table lists each of the ITM Party Synchronization XML elements/attributes and whether they are required or optional. A sample XML transaction follows the table.

ITM XML Field Information

Field Name	Required?	Transaction Table.Column	Description
CNTROL.AREA	Required		The fields included in this area provide information about the XML document
CNTROLAREA..BSR	Required		Shows the Business Service Request name per OAGI
CNTROLAREA.VER B	Required		Value is SYNC
CNTROLAREA.NO UN	Required		Value is CUSTOMER
CNTROLAREA.RE VISION	Required		Value is 005
CNTROLAREA.S ENDER	Required		Provides information on the system that sends the document
CNTROLAREA.S ENDER.LOGICAL ID	Required		Sender system identifier
CNTROLAREA.S ENDER.COMPON ENT	Required		Sender application name. Value is WSH
CNTROLAREA.S ENDER.TASK	Required		Event or Action
CNTROLAREA.S ENDER.REFER ENCEID	Required		Unique reference ID for this document
CNTROLAREA.S ENDER.CONFIR MATION	Required		Confirmation when document is received

CNTROLAREA.SENDER.LANGUAGE	Required	Language in which the text fields are transmitted
CNTROLAREA.SENDER.CODEPAGE	Required	Character set used in this XML document
CNTROLAREA.SENDER.AUTHID	Required	System ID of sender. Value is APPS
CNTROLAREA.DATETIME qualifier="CREATION"	Required	Creation date and time of the XML document
CNTROLAREA.DATETIME.YEAR	Required	Creation year of the XML document
CNTROLAREA.DATETIME.MONTH	Required	Creation month of the XML document
CNTROLAREA.DATETIME.DAY	Required	Creation day of the XML document
CNTROLAREA.DATETIME.HOUR	Required	Creation hour of the XML document
CNTROLAREA.DATETIME.MINUTE	Required	Creation minute of the XML document
CNTROLAREA.DATETIME.SECOND	Required	Creation second of the XML document
CNTROLAREA.DATETIME.SUBSECOND	Required	Creation subsecond of the XML document
CNTROLAREA.DATETIME.TIMEZONE	Required	Creation timezone of the XML document
DATAAREA		
DATAAREA.SYNC_CUSTOMER		

DATAAREA.SYNC_
CUSTOMER.PARTN
ER

DATAAREA.SYNC_
CUSTOMER.PARTN
ER.NAME index="1"

HZ_PARTIES.party_
name Partner name

DATAAREA.SYNC_
CUSTOMER.PARTN
ER.PARTNRID

HZ_CUST_SITE_USE
S_ALL.Site_use_id PARTNRID is the
 identifier of the
 partner that the
 PARTNRTYPE
 defines. (Site Use Id
 in Oracle
 Applications)

DATAAREA.SYNC_ CUSTOMER.PARTN ER.PARTNRTYPE	FND_LOOKUPS.me aning (ForFND_LOOKUPS. code = HZ_CUST_SITE_USE S.Site_Use_Code = fnd_lookups.lookup_ code)	<p>Indicates the type of partner entity or the site use code. Valid values are:</p> <p>Bill To: The customer location where the invoice goes to.</p> <p>Ship To: Location of the customer where goods are sent.</p> <p>Sold To: The customer who is responsible for ordering the goods or services.</p> <p>Drawee: A customer account site responsible for paying bills.</p> <p>Dunning: The location where the customers dunning letters are sent.</p> <p>Invoices Legal: Acts as the customer account's legal site. A legal site is responsible for all government reporting requirements.</p> <p>Marketing: Marketing collateral is sent to this address.</p> <p>Self Service User Statements: The Customer location where the customers statements are sent.</p> <p>Bills of Lading</p> <p>Credit Memos</p> <p>Credit Contact</p>
---	--	---

		Deliver To
		Acknowledgments
DATAAREA.SYNC_ CUSTOMER.PARTN ER.PARTNRIDX	HZ_CUST_ACCOUN TS.Account_Number	The identifier of the partner that the receiving application may use to identify the PARTNER. The Customer Number will be sent here.
		Synonyms include:
		Customer Number, Supplier Number, Customer ID, and Supplier ID
DATAAREA.SYNC_ CUSTOMER.PARTN ER.TAXID	Hz_parties.tax_refere nce	The tax identifier of the business partner.
		Synonyms include:
		VAT ID, Federal Tax ID, and State Tax ID
DATAAREA.SYNC_ CUSTOMER.PARTN ER.USERAREA		PARTNER LEVEL
DATAAREA.SYNC_ CUSTOMER.PARTN ER.USERAREA.SITE_ NUMBER	HZ_PARTY_SITES.Pa rty_Site_Number	Site number
DATAAREA.SYNC_ CUSTOMER.PARTN ER.USERAREA.OPE RATING_UNIT	HZ_Cust_Acct_sites_ all.OrgID	Operating unit

DATAAREA.SYNC_ CUSTOMER.PARTN ER.USERAREA.PAR AMETERS		This element bounds any additional parameters that are to be sent. This element has an attribute named TYPE, which can have any of the following values: VENDOR, SERVICE, or DATA.
DATAAREA.SYNC_ CUSTOMER.PARTN ER.USERAREA.PAR AMETERS. PARAMETER	Parameters of type 'VENDOR'WSH_ITM _VENDOR_PARAME TERS.Parameter_name / WSH_ITM_VENDOR _PARAMETERS.Par meter_value Parameters of type 'SERVICE'WSH_ITM _SERVICE_PARAME TERS.Prameter_name / WSH_ITM_SERVICE _PARAMETERS.Pra meter_value Parameters of type 'DATA'HZ_CUST_A CCOUNTS.Attribute1 ..15	Every Parameter has attributes Index, Name, and Value
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESS		The Data Type ADDRESS represents the data concerning the location of the business partner.
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSADDRL INE index="1" - "4"	HZ_LOCATIONS.AD DRESS1..ADDRESS4	The lines needed to describe street address or post office box information.
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSADDRT YPE	HZ_CUST_SITE_USE S_ALL.Primary_Flag	A qualification of the address of the business partner

DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSCITY	HZ_LOCATIONS.CI TY	The city or town of the business partner.
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSCOUN TRY	HZ_LOCATIONS.CO UNTRY	The country within an address. The list of valid codes can be found in ISO Standard number 3166.
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSPOSTA LCODE	HZ_LOCATIONS.PO STAL_CODE	The postal code within a mailing address. Synonym: Zip code
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSSTATE PROVN	HZ_LOCATIONS.ST ATE	The state or province within the mailing address.
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSUSERA REA		ADDRESS LEVEL
DATAAREA.SYNC_ CUSTOMER.PARTN ER.ADDRESSUSERA REA.PARAMETERS		This element bounds any additional parameters to be sent. This element has an attribute named TYPE which has value DATA - Additional Data Information / Descriptive Flexfield Information. All of the Descriptive Flexfield attributes (attribute1 through attribute15) of the Customer Site Level can be passed in this region.

DATAAREA.SYNC_CUSTOMER.PARTNER.ADDRESSUSERAREA.PARAMETERS.PARAMETER	HZ_CUST_ACCT_SITES_ALL.attribute1 through 15	Every Parameter has attributes 'Index, Name, and Value
DATAAREA.CONTACT		Represents people associated with the PARTNER and various ways to contact them, such as telephone or fax.
DATAAREA.CONTACT.NAME	HZ_PARTIES.Party_name (Having a record in HZ_PARTY_RELATIONSHIPS with relationship_type = 'CONTACT_OF'	First Name and the Last Name
DATAAREA.CONTACT.NAME.CONTCTTYPE		Describes the contact person's responsibilities. Synonym: Job Title
DATAAREA.CONTACT.NAME.EMAIL	HZ_CONTACT_POINTS.Email	Contact email address
DATAAREA.CONTACT.NAME.FAX	HZ_CONTACT_POINTS.PHONE_NUMBER(With Phone_Line_Type = 'FAX')	Contact fax number
DATAAREA.CONTACT.NAME.TELEPHONE	HZ_CONTACT_POINTS.PHONE_NUMBER(With Phone_Line_Type = 'PHONE')	Contact phone number

```

<?xml version="1.0" encoding = "UTF-8"?><SYNC_CUSTOMER_005>
  <CNTROLAREA>
    <BSR>
      <VERB>SYNC</VERB>
      <NOUN>PARTY</NOUN>
      <REVISION>001</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>ITM Application</COMPONENT>
      <TASK>PARTY SYNCHRONIZATION</TASK>
      <REFERENCEID>111~PARTY_SYNC~61</REFERENCEID>
      <CONFIRMATION>Y</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>vsathiya-venkat</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2004</YEAR>
      <MONTH>04</MONTH>
      <DAY>16</DAY>
      <HOUR>12</HOUR>
      <MINUTE>00</MINUTE>
      <SECOND>54</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+00:00</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SYNC_CUSTOMER>
      <PARTNER>
        <NAME index="1">SV200</NAME>
        <PARTNRID>10217</PARTNRID>
        <PARTNRRTYPE>BILL_TO</PARTNRRTYPE>
        <PARTNRIDX>R2000</PARTNRIDX>
        <TAXID/>
        <USERAREA>
          <ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
            <ORACLE.SITE_NUMBER>10595</ORACLE.SITE_NUMBER>
            <ORACLE.OPERATING_UNIT>204</ORACLE.OPERATING_UNIT>
            <ORACLE.PARAMETERS TYPE="VENDOR">
              <ORACLE.PARAMETER INDEX="1" NAME="Para1"
VALUE="Para1"/>
              <ORACLE.PARAMETER INDEX="2" NAME="Para5"
VALUE="para5"/>
              <ORACLE.PARAMETER INDEX="3" NAME="Para3"
VALUE="Para3"/>
            </ORACLE.PARAMETERS>
            <ORACLE.PARAMETERS TYPE="SERVICE"/>
            <ORACLE.PARAMETERS TYPE="DATA"/>
          </ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
        </USERAREA>
        <ADDRESS>
          <ADDRLINE index="1">139</ADDRLINE>
          <ADDRLINE index="2"/>
          <ADDRLINE index="3"/>
          <ADDRLINE index="4"/>
          <ADDRLINE index="5"/>
          <ADDRTYPE>N</ADDRTYPE>
          <CITY>Atlanta</CITY>
          <COUNTRY>US</COUNTRY>

```

```

<POSTALCODE>30301</POSTALCODE>
  <STATEPROVN>GA</STATEPROVN>
  <USERAREA>
    <ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
      <ORACLE.PARAMETERS TYPE="DATA"/>
    </ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
  </USERAREA>
</ADDRESS>
</PARTNER>
</SYNC_CUSTOMER>
</DATAAREA>
<DATAAREA>
  <SYNC_CUSTOMER>
    <PARTNER>
      <NAME index="1">SV200</NAME>
      <PARTNRID>10216</PARTNRID>
      <PARTNRTYPE>SHIP_TO</PARTNRTYPE>
      <PARTNRIDX>R2000</PARTNRIDX>
      <TAXID/>
      <USERAREA>
        <ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
          <ORACLE.SITE_NUMBER>10595</ORACLE.SITE_NUMBER>
          <ORACLE.OPERATING_UNIT>204</ORACLE.OPERATING_UNIT>
          <ORACLE.PARAMETERS TYPE="VENDOR">
            <ORACLE.PARAMETER INDEX="1" NAME="Para1"
VALUE="Para1"/>
            <ORACLE.PARAMETER INDEX="2" NAME="Para5"
VALUE="para5"/>
            <ORACLE.PARAMETER INDEX="3" NAME="Para3"
VALUE="Para3"/>
          </ORACLE.PARAMETERS>
          <ORACLE.PARAMETERS TYPE="SERVICE"/>
          <ORACLE.PARAMETERS TYPE="DATA"/>
        </ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
      </USERAREA>
    <ADDRESS>
      <ADDRLINE index="1">139</ADDRLINE>
      <ADDRLINE index="2"/>
      <ADDRLINE index="3"/>
      <ADDRLINE index="4"/>
      <ADDRLINE index="5"/>
      <ADDRTYPE>N</ADDRTYPE>
      <CITY>Atlanta</CITY>
      <COUNTRY>US</COUNTRY>
      <POSTALCODE>30301</POSTALCODE>
      <STATEPROVN>GA</STATEPROVN>
      <USERAREA>
        <ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
          <ORACLE.PARAMETERS TYPE="DATA"/>
        </ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
      </USERAREA>
    </ADDRESS>
  </PARTNER>
</SYNC_CUSTOMER>
</DATAAREA>
<DATAAREA>
  <SYNC_CUSTOMER>
    <PARTNER>
      <NAME index="1">SV200</NAME>
      <PARTNRID>10214</PARTNRID>
      <PARTNRTYPE>BILL_TO</PARTNRTYPE>

```

```

<PARTNRIDX>R2000</PARTNRIDX>
  <TAXID/>
  <USERAREA>
    <ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
      <ORACLE.SITE_NUMBER>SR1</ORACLE.SITE_NUMBER>
      <ORACLE.OPERATING_UNIT>204</ORACLE.OPERATING_UNIT>
      <ORACLE.PARAMETERS TYPE="VENDOR">
        <ORACLE.PARAMETER INDEX="1" NAME="Para1"
VALUE="Para1"/>
        <ORACLE.PARAMETER INDEX="2" NAME="Para5"
VALUE="para5"/>
        <ORACLE.PARAMETER INDEX="3" NAME="Para3"
VALUE="Para3"/>
      </ORACLE.PARAMETERS>
      <ORACLE.PARAMETERS TYPE="SERVICE"/>
      <ORACLE.PARAMETERS TYPE="DATA"/>
    </ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
  </USERAREA>
  <ADDRESS>
    <ADDRLINE index="1">145</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRLINE index="5"/>
    <ADDRTYPE>Y</ADDRTYPE>
    <CITY>Atlanta</CITY>
    <COUNTRY>US</COUNTRY>
    <POSTALCODE>30301</POSTALCODE>
    <STATEPROVN>GA</STATEPROVN>
    <USERAREA>
      <ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
        <ORACLE.PARAMETERS TYPE="DATA"/>
      </ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
    </USERAREA>
  </ADDRESS>
</PARTNER>
</SYNC_CUSTOMER>
</DATAAREA>
<DATAAREA>
  <SYNC_CUSTOMER>
    <PARTNER>
      <NAME index="1">SV200</NAME>
      <PARTNRID>10210</PARTNRID>
      <PARTNRTYPE>SHIP_TO</PARTNRTYPE>
      <PARTNRIDX>R2000</PARTNRIDX>
      <TAXID/>
      <USERAREA>
        <ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>
          <ORACLE.SITE_NUMBER>SR1</ORACLE.SITE_NUMBER>
          <ORACLE.OPERATING_UNIT>204</ORACLE.OPERATING_UNIT>
          <ORACLE.PARAMETERS TYPE="VENDOR">
            <ORACLE.PARAMETER INDEX="1" NAME="Para1"
VALUE="Para1"/>
            <ORACLE.PARAMETER INDEX="2" NAME="Para5"
VALUE="para5"/>
            <ORACLE.PARAMETER INDEX="3" NAME="Para3"
VALUE="Para3"/>
          </ORACLE.PARAMETERS>
          <ORACLE.PARAMETERS TYPE="SERVICE"/>
          <ORACLE.PARAMETERS TYPE="DATA"/>
        </ORACLE.SYNC_CUSTOMER_005.PARTNER.USERAREA>

```

```

</USERAREA>
  <ADDRESS>
    <ADDRLINE index="1">145</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRLINE index="5"/>
    <ADDRTYPE>Y</ADDRTYPE>
    <CITY>Atlanta</CITY>
    <COUNTRY>US</COUNTRY>
    <POSTALCODE>30301</POSTALCODE>
    <STATEPROVN>GA</STATEPROVN>
    <USERAREA>
      <ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
        <ORACLE.PARAMETERS TYPE="DATA"/>
      </ORACLE.SYNC_CUSTOMER_005.PARTNER.ADDRESS.USERAREA>
    </USERAREA>
  </ADDRESS>
</PARTNER>
</SYNC_CUSTOMER>
</DATAAREA>
</SYNC_CUSTOMER_005>

```

ITM Sample Item Synchronization XML

The following table lists each of the ITM Item Synchronization XML elements/attributes and whether they are required or optional. A sample XML transaction follows the table.

ITM XML Field Information

Field Name	Required?	Transaction Table.Column	Description
CNTR0L.AREA	Required		The fields included in this area provide information about the XML document
CNTR0LAREA..BSR	Required		Shows the Business Service Request name per OAGI
CNTR0LAREA.VER B	Required		Value is SYNC
CNTR0LAREA.NO N	Required		Value is CUSTOMER
CNTR0LAREA.REV SION	Required		Value is 005

CNTROLAREA.SENDER	Required	Provides information on the system that sends the document
CNTROLAREA.SENDER.LOGICALID	Required	Sender system identifier
CNTROLAREA.SENDER.COMPONENT	Required	Sender application name. Value is WSH
CNTROLAREA.SENDER.TASK	Required	Event or Action
CNTROLAREA.SENDER.REFERENCEID	Required	Unique reference ID for this document
CNTROLAREA.SENDER.CONFIRMATION	Required	Confirmation when document is received
CNTROLAREA.SENDER.LANGUAGE	Required	Language in which the text fields are transmitted
CNTROLAREA.SENDER.CODEPAGE	Required	Character set used in this XML document
CNTROLAREA.SENDER.AUTHID	Required	System ID of sender. Value is APPS
CNTROLAREA.DATETIME qualifier="CREATION"	Required	Creation date and time of the XML document
CNTROLAREA.DATETIME.YEAR	Required	Creation year of the XML document
CNTROLAREA.DATETIME.MONTH	Required	Creation month of the XML document
CNTROLAREA.DATETIME.DAY	Required	Creation day of the XML document

CNTROLAREA.DAT ETIME.HOUR	Required		Creation hour of the XML document
CNTROLAREA.DAT ETIME.MINUTE	Required		Creation minute of the XML document
CNTROLAREA.DAT ETIME.SECOND	Required		Creation second of the XML document
CNTROLAREA.DAT ETIME.SUBSECOND	Required		Creation subsecond of the XML document
CNTROLAREA.DAT ETIME.TIMEZONE	Required		Creation timezone of the XML document
DATAAREA			
DATAAREA.ITEM			
DATAAREA.ITEM.ITEM_ID		MTL_SYSTEM_ITEM S_B.inventory_item_id	Unique Identifier of an Item (Inventory_item_id)
DATAAREA.ITEM.ITEM_CODE		MTL_SYSTEM_ITEM S_B.segment1..segment15	Product Code. Concatenated segments of the Item Flexfield.
DATAAREA.ITEM.DESCRPTION		MTL_SYSTEM_ITEM S_TL.Description	Description of the item
DATAAREA.ITEM.HAZARD_CLASS		PO_HAZARD_CLASSES_TL.(For PO_HAZARD_CLASSES_TL.hazard_class_id = MTL_SYSTEM_ITEM S_B.hazard_class_id)	Hazard class classification
DATAAREA.ITEM.ORGANIZATION_ID		MTL_SYSTEM_ITEM S_B.Organization_id	Organization to which this item is assigned

DATAAREA.ITEM.S RC_ORGANIZATION_ID	MTL_SYSTEM_ITEM S_B.src_organization_id	Master Organization of the item.
DATAAREA.ITEM.ITEM_TYPE	FND_COMMON_LOOKUPS.meaning (forLookup_type =ITEM_TYPE andFND_COMMON_LOOKUPS.lookup_code = MTL_SYSTEM_ITEM S_B.item_type	User Defined Item Type
DATAAREA.ITEM.UNIT_OF_MEASURE	MTL_SYSTEM_ITEM S_B.PRIMARY_UOM_CODE	Unit of measure
DATAAREA.ITEM.ITEM_VALUE	MTL_SYSTEM_ITEM S_B.LIST_PRICE_PER_UNIT	Value of the item
DATAAREA.ITEM.CURRENCY		Currency
DATAAREA.ITEM.COUNTRY	HR_LOCATIONS_ALL.Country(Select hl.country from hr_all_organization_units hu, hr_locations hl where hu.location_id = hl.location_id and hu.organization_id = MTL_SYSTEM_ITEM _B.Organization_id)	Country of the organization to which the item is assigned in Oracle Application.
DATAAREA.ITEM.USERAREA		

```

<?xml version="1.0" encoding = "UTF-8"?><ITEM_SYNC>
  <CNTROLAREA>
    <BSR>
      <VERB>SYNC</VERB>
      <NOUN>ITEM</NOUN>
      <REVISION>001</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>ITM Application</COMPONENT>
      <TASK>ITEM SYNCHRONIZATION</TASK>
      <REFERENCEID>142~ITEM_SYNC~61</REFERENCEID>
      <CONFIRMATION>Y</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>vsathiya-venkat</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2004</YEAR>
      <MONTH>06</MONTH>
      <DAY>05</DAY>
      <HOUR>02</HOUR>
      <MINUTE>53</MINUTE>
      <SECOND>51</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+00:00</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <ITEM>
      <ITEM_ID>11746</ITEM_ID>
      <ITEM_CODE>WSH-Standard</ITEM_CODE>
      <DESCRIPTION>WSH-Standard</DESCRIPTION>
      <HAZARD_CLASS/>
      <ORGANIZATION_ID>8259</ORGANIZATION_ID>
      <SRC_ORGANIZATION_ID/>
      <ITEM_TYPE>Finished good</ITEM_TYPE>
      <UNIT_OF_MEASURE>Ea</UNIT_OF_MEASURE>
      <ITEM_VALUE/>
      <CURRENCY>USD</CURRENCY>
      <COUNTRY>CN</COUNTRY>
      <USERAREA>
        <ORACLE.ITEM_SYNC.ITEM.USERAREA>
          <ORACLE.ITEM_SYNC.ITEM.LONG_DESCRIPTION/>
          <ORACLE.OPERATING_UNIT/>
          <ORACLE.PARAMETERS TYPE="VENDOR">
            <ORACLE.PARAMETER INDEX="1" NAME="PPar-1"
VALUE="PVal-1"/>
          </ORACLE.PARAMETERS>
          <ORACLE.PARAMETERS TYPE="SERVICE">
            <ORACLE.PARAMETER INDEX="1" NAME="Par1" VALUE="Val1"/>
          </ORACLE.PARAMETERS>
          <ORACLE.PARAMETERS TYPE="DATA"/>
        </ORACLE.ITEM_SYNC.ITEM.USERAREA>
      </USERAREA>
    </ITEM>
  </DATAAREA>
</DATAAREA>
  <ITEM>
    <ITEM_ID>11746</ITEM_ID>
    <ITEM_CODE>WSH-Standard</ITEM_CODE>

```

```

<DESCRIPTION>WSH-Standard</DESCRIPTION>
  <HAZARD_CLASS/>
  <ORGANIZATION_ID>8260</ORGANIZATION_ID>
  <SRC_ORGANIZATION_ID/>
  <ITEM_TYPE>Finished good</ITEM_TYPE>
  <UNIT_OF_MEASURE>Ea</UNIT_OF_MEASURE>
  <ITEM_VALUE/>
  <CURRENCY>USD</CURRENCY>
  <COUNTRY>AU</COUNTRY>
  <USERAREA>
    <ORACLE.ITEM_SYNC.ITEM.USERAREA>
      <ORACLE.ITEM_SYNC.ITEM.LONG_DESCRIPTION/>
      <ORACLE.OPERATING_UNIT/>
      <ORACLE.PARAMETERS TYPE="VENDOR">
        <ORACLE.PARAMETER INDEX="1" NAME="PPar-1"
VALUE="PVal-1"/>
      </ORACLE.PARAMETERS>
      <ORACLE.PARAMETERS TYPE="SERVICE">
        <ORACLE.PARAMETER INDEX="1" NAME="Par1" VALUE="Val1"/>
      </ORACLE.PARAMETERS>
      <ORACLE.PARAMETERS TYPE="DATA"/>
    </ORACLE.ITEM_SYNC.ITEM.USERAREA>
  </USERAREA>
</ITEM>
</DATAAREA>
<DATAAREA>
  <ITEM>
    <ITEM_ID>11746</ITEM_ID>
    <ITEM_CODE>WSH-Standard</ITEM_CODE>
    <DESCRIPTION>WSH-Standard</DESCRIPTION>
    <HAZARD_CLASS/>
    <ORGANIZATION_ID>8261</ORGANIZATION_ID>
    <SRC_ORGANIZATION_ID/>
    <ITEM_TYPE>Finished good</ITEM_TYPE>
    <UNIT_OF_MEASURE>Ea</UNIT_OF_MEASURE>
    <ITEM_VALUE/>
    <CURRENCY>USD</CURRENCY>
    <COUNTRY>AU</COUNTRY>
    <USERAREA>
      <ORACLE.ITEM_SYNC.ITEM.USERAREA>
        <ORACLE.ITEM_SYNC.ITEM.LONG_DESCRIPTION/>
        <ORACLE.OPERATING_UNIT/>
        <ORACLE.PARAMETERS TYPE="VENDOR">
          <ORACLE.PARAMETER INDEX="1" NAME="PPar-1"
VALUE="PVal-1"/>
        </ORACLE.PARAMETERS>
        <ORACLE.PARAMETERS TYPE="SERVICE">
          <ORACLE.PARAMETER INDEX="1" NAME="Par1" VALUE="Val1"/>
        </ORACLE.PARAMETERS>
        <ORACLE.PARAMETERS TYPE="DATA"/>
      </ORACLE.ITEM_SYNC.ITEM.USERAREA>
    </USERAREA>
  </ITEM>
</DATAAREA>
<DATAAREA>
  <ITEM>
    <ITEM_ID>11746</ITEM_ID>
    <ITEM_CODE>WSH-Standard</ITEM_CODE>
    <DESCRIPTION>WSH-Standard</DESCRIPTION>
    <HAZARD_CLASS/>
    <ORGANIZATION_ID>8262</ORGANIZATION_ID>

```

```

<SRC_ORGANIZATION_ID/>
  <ITEM_TYPE>Finished good</ITEM_TYPE>
  <UNIT_OF_MEASURE>Ea</UNIT_OF_MEASURE>
  <ITEM_VALUE/>
  <CURRENCY>USD</CURRENCY>
  <COUNTRY>AU</COUNTRY>
  <USERAREA>
    <ORACLE.ITEM_SYNC.ITEM.USERAREA>
      <ORACLE.ITEM_SYNC.ITEM.LONG_DESCRIPTION/>
      <ORACLE.OPERATING_UNIT/>
      <ORACLE.PARAMETERS TYPE="VENDOR">
        <ORACLE.PARAMETER INDEX="1" NAME="PPar-1"
VALUE="PVal-1"/>
      </ORACLE.PARAMETERS>
      <ORACLE.PARAMETERS TYPE="SERVICE">
        <ORACLE.PARAMETER INDEX="1" NAME="Par1" VALUE="Val1"/>
      </ORACLE.PARAMETERS>
      <ORACLE.PARAMETERS TYPE="DATA"/>
    </ORACLE.ITEM_SYNC.ITEM.USERAREA>
  </USERAREA>
</ITEM>
</DATAAREA>
</ITEM_SYNC>

```

ITM Sample Order Export Compliance XML

The following table lists each of the ITM Order Export Compliance XML elements/attributes and whether they are required or optional. A sample XML transaction follows the table.

ITM XML Field Information

Field Name	Required?	Transaction Table.Column	Description
ORDER	Required	Represents the Order line information. Every Order line would map to one Order Tag and multiple Order Tags would be present in a message for same Order Number.	Contains the Order Header Information.
ORDER_LINE	Required		
ORDER_LINE.ORDER_NUMBER	Required	OE_ORDER_HEADER.ORDER_NUMBER	Order number

ORDER_LINE.ORDER_TYPE	Required	OE_ORDER_HEADERS_ALL.ORDER_TYPE	Order type
ORDER_LINE.ORG_ID	Required	OE_ORDER_HEADERS_ALL.ORG_ID	Operating unit
ORDER_LINE.CUSTOMER_PO_NUM	Optional	OE_ORDER_HEADERS_ALL.CUSTOMER_PO_NUM	Purchase order number
ORDER_LINE.TRANSACTIONAL_CURRENCY_CODE	Optional	OE_ORDER_HEADERS_ALL.TRANSACTIONAL_CURRENCY_CODE	Currency
ORDER_LINE.CONVERSION_TYPE_CODE	Optional	OE_ORDER_HEADERS_ALL.CONVERSION_TYPE_CODE	Conversion code
ORDER_LINE.CONVERSION_RATE	Optional	OE_ORDER_HEADERS_ALL.CONVERSION_RATE	Conversion rate
ORDER_LINE.ORDER_DATE	Optional	OE_ORDER_HEADERS_ALL.ORDERED_DATE	Order date
ORDER_LINE.SHIPPING_METHOD_CODE	Optional	OE_ORDER_LINES_ALL.SHIPPING_METHOD_CODE	Shipping method
ORDER_LINE.REQUEST_DATE	Optional	OE_ORDER_LINES_ALL.REQUEST_DATE	Request date
ORDER_LINE.FREIGHT_TERMS_CODE	Optional	OE_ORDER_LINES_ALL.FREIGHT_TERMS_CODE	Freight terms
ORDER_LINE.PAYMENT_TERMS_NAME	Optional	RA_TERMS_TL.NAME	Name of the payment terms
ORDER_LINE.PAYMENT_TERM_ID	Optional	OE_ORDER_HEADERS_ALL.PAYMENT_TERM_ID	Payment terms ID

ORDER_LINE.ORDERED_QUANTITY	Optional	OE_ORDER_LINES.ORDERED_QUANTITY	Quantity ordered
ORDER_LINE.LINE_NUMBER	Required	OE_ORDER_LINES_ALL.LINE_NUMBER	Line number
ORDER_LINE.UNIT_LIST_PRICE	Optional	OE_ORDER_LINES_ALL.UNIT_LIST_PRICE	List price
ORDER_LINE.UNIT_SELLING_PRICE	Optional	OE_ORDER_LINES_ALL.UNIT_SELLING_PRICE	Sell price
ORDER_LINE.ITEM_ID	Required	ITEM_ID	Item ID, which 3rd Party Partner application can recognize as a primary key.
ORDER_LINE.PARTY	Required		
ORDER_LINE.PARTY.PARTY_ID	Required		Identification from ITM adapter
ORDER_LINE.PARTY.TYPE	Optional	FND_LOOKUPS.meaning (For FND_LOOKUPS.code = HZ_CUST_SITE_USES.Site_Use_Code = fnd_lookups.lookup_code)	The party type is just an informative field. Type SHIP_FROM, SHIP_TO, BILL_TO, DELIVER To, SOLD_TO, and Intermediate Ship To are selected.
ORDER_LINE.PARTY.NAME	Optional	HZ_PARTIES.party_name	Party name
ORDER_LINE.PARTY.ADDRESS1-5	Optional	HZ_LOCATIONS.ADDRESS1..ADDRESS5	Party Address 1 through 5
ORDER_LINE.PARTY.CITY	Optional	HZ_LOCATIONS.CITY	City

ORDER_LINE.PART Y.STATE_OR_PROVI NCE	Optional	HZ_LOCATIONS.ST ATE	State or province
ORDER_LINE.PART Y.COUNTRY CODE="US" NAME="US"	Optional	HZ_LOCATIONS.CO UNTRY	Country code
ORDER_LINE.PART Y.POSTAL_CODE	Optional	HZ_LOCATIONS.PO STAL_CODE	Postal code
ORDER_LINE.PART Y.CONTACT_NAME	Optional	HZ_PARTIES.Party_ name (Having a record in HZ_PARTY_RELATI ONSHIPS with relationship_type = 'CONTACT_OF')	Contact name
ORDER_LINE.PART Y.EMAIL	Optional	HZ_CONTACT_POI NTS.EMAIL	Contact email adress
ORDER_LINE.PART Y.PHONE	Optional	HZ_CONTACT_POI NTS.PHONE_NUMB ER	Contact phone number
ORDER_LINE.PART Y.FAX	Optional	HZ_CONTACT_POI NTS.PHONE_NUMB ER	Contact fax number
ORDER_LINE.PART Y.WEB	Optional		Contact web site URL
ORDER_LINE.PART Y.ADDL_INFO	Optional	From WSH_ITM_PARTIES table, ATTRIBUTE1_NAME to ATTRIBUTE20_NAM E and ATTRIBUTE1_VALU E to ATTRIBUTE20_VAL UE.	Attributes to include additional attributes from Name, Value pairs or via pre process custom APIs.

```

<?xml version="1.0" encoding = "UTF-8"?><request>
  <message_id>7832~OM_EXPORT_COMPLIANCE~61</message_id>
  <admin>
    <subscriber_id>vsathiya</subscriber_id>
    <password>xxxxxx</password>
    <request_generator>OM</request_generator>
    <revision>1.0</revision>
  </admin>
  <parameters>
    <parameter name="Para1" value="Para1"/>
    <parameter name="Para5" value="para5"/>
    <parameter name="Para3" value="Para3"/>
  </parameters>
  <transaction_date>4/28/2004 6:35:21</transaction_date>
  <addl_info/>
  <order_line>
    <transaction_key>7832~50027-Standard_ECE-1</transaction_key>
    <order_number>50027</order_number>
    <order_type>Standard_ECE</order_type>
    <org_id>204</org_id>
    <cust_po_num/>
    <transactional_curr_code>USD</transactional_curr_code>
    <conversion_type_code/>
    <conversion_rate/>
    <ordered_date>4/28/2004 6:25:57</ordered_date>
    <shipping_method_code>000001_United
Par_A_3DS</shipping_method_code>
    <request_date>4/28/2004 6:25:57</request_date>
    <freight_terms_code>Due</freight_terms_code>
    <payment_terms_name/>
    <payment_term_id>4</payment_term_id>
    <ordered_quantity>5</ordered_quantity>
    <line_number>1</line_number>
    <unit_list_price>50</unit_list_price>
    <unit_selling_price>50</unit_selling_price>
    <item_id>29585</item_id>
    <party>
      <party_id>7832-498</party_id>
      <type>SHIP_FROM</type>
      <name>Vision Operations</name>
      <address1>90 Fifth Avenue,,</address1>
      <address2/>
      <address3/>
      <address4/>
      <address5/>
      <city>New York</city>
      <state_or_province/>
      <country code="US" name="US"/>
      <postal_code>10022-3422</postal_code>
      <contact_name/>
      <email/>
      <phone/>
      <fax/>
      <web/>
      <addl_info/>
    </party>
    <party>
      <party_id>7832-499</party_id>
      <type>SHIP_TO</type>
      <name>Computer Service and Rentals</name>
      <address1>301 Summit Hill Drive 1,,,</address1>

```

```

<address2/>
    <address3/>
    <address4/>
    <address5/>
    <city>Chattanooga</city>
    <state_or_province/>
    <country code="US" name="US"/>
    <postal_code>37401</postal_code>
    <contact_name/>
    <email/>
    <phone/>
    <fax/>
    <web/>
    <addl_info/>
</party>
<party>
    <party_id>7832-500</party_id>
    <type>DELIVER_TO</type>
    <name>Computer Service and Rentals</name>
    <address1>234 Lester B. Pearson Road,,,</address1>
    <address2/>
    <address3/>
    <address4/>
    <address5/>
    <city>Iqualuit</city>
    <state_or_province/>
    <country code="CA" name="CA"/>
    <postal_code>I2W 3E4</postal_code>
    <contact_name/>
    <email/>
    <phone/>
    <fax/>
    <web/>
    <addl_info/>
</party>
<party>
    <party_id>7832-501</party_id>
    <type>BILL_TO</type>
    <name>Computer Service and Rentals</name>
    <address1>301 Summit Hill Drive 1,,,</address1>
    <address2/>
    <address3/>
    <address4/>
    <address5/>
    <city>Chattanooga</city>
    <state_or_province/>
    <country code="US" name="US"/>
    <postal_code>37401</postal_code>
    <contact_name/>
    <email/>
    <phone/>
    <fax/>
    <web/>
    <addl_info/>
</party>
<party>
    <party_id>7832-502</party_id>
    <type>SOLD_TO</type>
    <name>Computer Service and Rentals</name>
    <address1>301 Summit Hill Drive 1,,,</address1>
    <address2/>

```



```

<address3/>
    <address4/>
    <address5/>
    <city>Chattanooga</city>
    <state_or_province/>
    <country_code="US" name="US"/>
    <postal_code>37401</postal_code>
    <contact_name/>
    <email/>
    <phone/>
    <fax/>
    <web/>
    <addl_info/>
  </party>
</order_line>
</request>

```

ITM Sample Delivery Export Compliance XML

The following table lists each of the ITM Delivery Export Compliance XML elements/attributes and whether they are required or optional. A sample XML transaction follows the table.

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.DATE TTIME (Document)	Required	Sys date time	
SHIPMENT.DOC UMENTID (Shipment)	Required	WSH_TRANSACTIONS_H ISTORY.DOCUMENT_ID	
SHIPMENT.SHIP PERID	Required	WSH_NEW_DELIVERIES.I NITIAL_PICKUP_LOCATI ON_ID converted to code	
SHIPMENT.SYNC ID	Required	XML: "A" = Add, "D" = Delete "C" = Change EDI: " N" for Add, "F" for Delete	
SHIPMENT.AMO UNT (DECLAREVAL)	Optional		
SHIPMENT.AMO UNT (DOCUMENT) (T)	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.AMO UNT (ESTFREIGHT)	Optional		
SHIPMENT.AMO UNT (FREIGHT) (T)	Optional	WSH_FREIGHT_COSTS.T OTAL_AMOUNT	
SHIPMENT.AMO UNT (ITEM) (T)	Optional		
SHIPMENT.CARR IER	Optional	WSH_NEW_DELIVERIES. CARRIER_ID, converted to external value	
SHIPMENT.CARR IERSRVLV	Optional	WSH_NEW_DELIVERIES.S ERVICE_LEVEL, converted to external value	
SHIPMENT.COU NTRYDST	Optional	For wsh_new_deliveries. Ultimate_dropoff_location_ id, Get the country from hz_locations	
SHIPMENT.COU NTRYORG	Optional		
SHIPMENT.DATE TIME (CREATION)	Optional	WSH_NEW_DELIVERIES. CREATION_DATE	
SHIPMENT.DATE TIME (DELIVACT)	Optional	WSH_TRIP_STOPS.ACTU AL_ARRIVAL_DATE	
SHIPMENT.DATE TIME (DELIVSCHED)	Optional	WSH_NEW_DELIVERIES. .ULTIMATE_DROPOFF_D ATE	
SHIPMENT.DATE TIME (EARLSTSHIP)	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.DATE TIME (LOADING)	Optional		
SHIPMENT.DATE TIME (NEEDDELV)	Optional	WSH_NEW_DELIVERIES. .ULTIMATE_DROPOFF_D ATE	
SHIPMENT.DATE TIME (PROMDELV)	Optional		
SHIPMENT.DATE TIME (PROMSHIP)	Optional		
SHIPMENT.DATE TIME (SHIP)	Optional	WSH_TRIP_STOPS.ACTU AL_DEPARTURE_DATE	
SHIPMENT.DATE TIME (SHIPSCHED)	Optional	WSH_NEW_DELIVERIES.I NITIAL_PICKUP_DATE	
SHIPMENT.DESC RIPTN	Optional	WSH_NEW_DELIVERIES. DESCRIPTION	
SHIPMENT.DIST CENTER	Optional		
SHIPMENT.DOC KID	Optional		
SHIPMENT.DOCT YPE	Optional	"SHIPMENT REQUEST" for 940 "SHIPMENT ADVICE" for 945	
SHIPMENT.DOC UMENTRV	Optional	"001" for ADD, "002" for DELETE, "003" for CHANGE	
SHIPMENT.DROP SHIP	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.EXPO RTLIC	Optional		
SHIPMENT.FRGH TCLS	Optional		
SHIPMENT.FRGH TTERMS	Optional	WSH_NEW_DELIVERIES. FREIGHT_TERMS_CODE	
SHIPMENT.HAZ RDMATL	Optional		
SHIPMENT.HDRS TATUS	Optional		
SHIPMENT.IMPO RTLIC	Optional		
SHIPMENT.LOA DPOINT	Optional		
SHIPMENT.NOTE S1 – NOTES9	Optional		
SHIPMENT.OPER AMT (FREIGHT) (T)	Optional		
SHIPMENT.PART IALSHP	Optional		
SHIPMENT.PRIO RITY	Optional		
SHIPMENT.QUA NTITY (ESTWEIGHT)	Optional	WSH_NEW_DELIVERIES. GROSS_WEIGHT, WSH_NEW_DELIVERIES. WEIGHT_UOM	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.QUANTITY (LOADINGWT)	Optional	Calculate using WSH_NEW_DELIVERIES. GROSS WEIGHT – WSH_NEW_DELIVERIES. NET WEIGHT	
SHIPMENT.QUANTITY (NETWEIGHT)	Optional	WSH_NEW_DELIVERIES. NET_WEIGHT WSH_NEW_DELIVERIES. WEIGHT_UOM	
SHIPMENT.QUANTITY (SHIPUNIT)	Optional	WSH_NEW_DELIVERIES. NUMBER_OF_LPN	
SHIPMENT.QUANTITY (TOTWEIGHT)	Optional	WSH_NEW_DELIVERIES. GROSS_WEIGHT, WSH_NEW_DELIVERIES. WEIGHT_UOM	
SHIPMENT.QUANTITY (VOLUME)	Optional	WSH_NEW_DELIVERIES. VOLUME WSH_NEW_DELIVERIES. VOLUME_UOM	
SHIPMENT.ROUTEID	Optional		
SHIPMENT.RELEASENUM	Optional		
SHIPMENT.RELEASENUMTOT	Optional		
SHIPMENT.ROUTE ETYPE	Optional	Type of routing.	
SHIPMENT.SHIPPING NOTES	Optional	WSH_NEW_DELIVERIES.S HIPPING_MARKS	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.SHIP POINT	Optional	WSH_NEW_DELIVERIES. ULTIATE_DROPOFF_LOC ATION_ID converted to code	
SHIPMENT.SHIP PRIOR	Optional		
SHIPMENT.SHIPS RVLVL	Optional	WSH_NEW_DELIVERIES.S ERVICE_LEVEL	
SHIPMENT.SPECI ALHND	Optional		
SHIPMENT.STAG EPOINT	Optional		
SHIPMENT.TEMP RATURE (ACTUAL)	Optional		
SHIPMENT.TEMP RATURE (DELIVERY)	Optional		
SHIPMENT.TEMP RATURE (LOADING)	Optional		
SHIPMENT.TRA NSMETHD	Optional	WSH_NEW_DELIVERIES. MODE_OF_TRANSPORT	
SHIPMENT.TRA NSTERMS	Optional	WSH_NEW_DELIVERIES. FOB_CODE	
SHIPMENT.WAR EHOUSE	Optional	WSH_NEW_DELIVERIES. ORGANIZATION_ID converted to code	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPMENT.USER AREA	Optional	WSH_NEW_DELIVERIES. DELIVERY_NAME, WSH_NEW_DELIVERIES. LOADING_SEQUENCE, WSH_NEW_DELIVERY descriptive flex fields, WSH_TRIPS.VEHICLE_N UM_PREFIX, WSH_TRIPS.VEHICLE_N UMBER, WSH_TRIPS.ROUTE_ID, WSH_TRIPS.ROUTING_IN STRUCTIONS, WSH_TRIP_STOPS.DEPAR TURE_SEAL_CODE, WSH_NEW_DELIVERIES.S HIP_METHOD_CODE, NVL(WSH_TRIPS.SHIP_M ETHOD_CODE, WSH_NEW_DELIVERIES.S HIP_METHOD_CODE), WSH_NEW_DELIVERIES. WAYBILL, WSH_ITM_REQUEST_CO NTROL.ATTRIBUTE _n _NA ME/ATTRIBUTE _n _VALUE Where n=1 to 20, WSH_ITM_REQUEST_CO NTROL.TRIGGERING_POI NT	
PARTNER.	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER.NAME 1	Required		If this partner segment is the Ship To segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID, If this partner segment is the Ship from segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ORGANIZATION_ID If this partner segment is the Customer segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.CUSTOMER_ID If this partner segment is the Supplier segment, get the HZ_PARTIES.NAME for the WSH_NEW_DELIVERIES.ORGANIZATION_ID
PARTNER.ONETIME	Required		
PARTNER.PARTNRID	Required		WSH_NEW_DELIVERIES.ULTIMATE_DROPOFF_LOCATION_ID converted to external value, or WSH_NEW_DELIVERIES.ORGANIZATION_ID converted to external value.

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER.PART NRTYPE	Required	PARTNRNRTYPE	indicates the type of partner entity. Valid values are: "ShipTo" - The delivery location of the customer. "BillTo" - The customer location the bill goes to. "SoldTo" - The customer who is responsible for ordering the goods or services. Could be at multiple organizational levels. "Supplier" - The partner who is responsible for providing the goods or services. Supplier can be at multiple organizational levels. "Carrier" - The partner responsible for delivering the goods or services. "ShipFrom" - The shipping location of the partner who is responsible for providing the goods or services. Ship from can be at multiple organizational levels. "Intmd_ShipTo" - Intermeidate Ship for the delivery "FOB"- FOB for the delivery
PARTNER.ACTIV E	Optional		
PARTNER.CURR ENCY	Optional	WSH_NEW_DELIVERIES. CURRENCY_CODE	for the supplier and ship from partner segments
PARTNER.DESCR IPTN	Optional		
PARTNER.DUNS NUMBER	Optional	HZ_PARTIES.DUNS_NUM BER	for all partner types

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER.GLEN TITYS	Optional		
PARTNER.NAME 2-NAME9	Optional		
PARTNER.PARE NTID	Optional		
PARTNER.PART NIDX	Optional	Could use HZ_PARTIES.DUNS_NUM BER for the HZ_PARITES.PARTY_ID that matches the Ship From, Ship To, Customer and Supplier segments	
PARTNER.PART NRRATG	Optional		
PARTNER.PART NRROLE	Optional		
PARTNER.PAYM ETHOD	Optional		
PARTNER.TAXEX EMPT	Optional		
PARTNER.TAXID	Optional		
PARTNER.TERMI D	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER.USERAREA	Optional	WSH_NEW_DELIVERIES.INTMED_SHIP_TO_LOCATION_ID WSH_NEW_DELIVERIES.POOLED_SHIP_TO_LOCATION_ID HZ_CUST_SITE_USES_ALL.SITE_USE_ID HZ_CUST_SITE_USES_ALL.LOCATION	
PARTNER – ADDRESS.	Optional		
PARTNER – ADDRESS.ADDR LINE1-ADDRLINE9	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAILS table, via the HZ_PARTY_SITES table.	
PARTNER – ADDRESS.ADDRTYPE	Optional		
PARTNER – ADDRESS.CITY	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAILS table, via the HZ_PARTY_SITES table.	

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER – ADDRESS.COUNTRY	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAIL S table, via the HZ_PARTY_SITES table.	
PARTNER – ADDRESS.COUNTRY	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAIL S table, via the HZ_PARTY_SITES table.	
PARTNER – ADDRESS.DESCR IPTN	Optional		
PARTNER – ADDRESS.FAX1- FAX9	Optional	HZ_CONTACT_POINTS.R AW_PHONE_NUMBER for the PARTY_ID, and CONTACT_POINT_TYPE = "FAX"	

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER – ADDRESS.POSTA LCODE	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAIL S table, via the HZ_PARTY_SITES table.	
PARTNER – ADDRESS.REGIO N	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAIL S table, via the HZ_PARTY_SITES table.	
PARTNER – ADDRESS.STATE PROVN	Optional	Extracted from the HZ_LOCATIONS using the party id that matches the Ship From, Ship TO, Customer and Supplier from the WSH_NEW_DELIVERIES or WSH_DELIVERY_DETAIL S table, via the HZ_PARTY_SITES table.	
PARTNER – ADDRESS.TAXJR SDCTN	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER – ADDRESS.TELEPHONE1-TELEPHONE9	Optional	HZ_CONTACT_POINTS.RAW_PHONE_NUMBER for the PARTY_ID, and CONTACT_POINT_TYPE = "GEN"	
PARTNER – ADDRESS.URL	Optional	HZ_CONTACT_POINTS.URL for the PARTY_ID and CONTACT_POINT_TYPE = "GEN"	
PARTNER – ADDRESS.USERAREA	Optional		
PARTNER – CONTACT.	Optional		
PARTNER – CONTACT.NAME1	Required		
PARTNER – CONTACT.CONTACTTYPE	Optional		
PARTNER – CONTACT.DESCRPTION	Optional		
PARTNER – CONTACT.EMAIL	Optional		
PARTNER – CONTACT.FAX1 - FAX9	Optional		
PARTNER – CONTACT.NAME2 - NAME9	Optional	Extracted from the HR_LOCATIONS_ALL using the WSH ID	

Field Name	Required?	Transaction Table.Column Name	Description
PARTNER – CONTACT.TELEP HONE1 -TELEPHONE9	Optional		
PARTNER – CONTACT.USER AREA	Optional		
DOCUMNTREF.	Optional		We need to send this segment twice – once to represent the ship from delivery name, and for each ship to purchase order number
DOCUMNTREF.D OCTYPE	Required		"SHIPMENT REQUEST"
DOCUMNTREF.D OCUMENTID	Required		WSH_TRANSACTIONS_H ISTORY.Orig_Document_I D
DOCUMNTREF.P ARTNRID	Required		WSH_NEW_DELIVERIES. ORGANIZATION_ID converted to code
DOCUMNTREF.P ARTNRTYPE	Required		"ShipFrom" - The shipping location of the partner who is responsible for providing the goods or services. Ship from can be at multiple organizational levels
DOCUMNTREF.D ESCRPTN	Optional		
DOCUMNTREF.D OCUMENTRV	Optional		
DOCUMNTREF.L INENUM	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
DOCUMNTREF.N OTES1 – NOTES9	Optional		
DOCUMNTREF.S CHLINENUM	Optional		
DOCUMNTREF.S UBLINENUM	Optional		
DOCUMNTREF.U SERAREA	Optional		
SHIPITEM.	Optional		
SHIPITEM.ITEM	Required	WSH_DELIVERY_DETAIL S.INVENTORY_ITEM_ID (foreign key to MTL_SYSTEM_ITEMS) Derive Item_Number from MTL_SYSTEM_ITEMS	
SHIPITEM.QUAN TITY (ITEM)	Required	WSH_DELIVERY_DETAIL S.REQUESTED_QUANTIT Y	
SHIPITEM.AMOU NT (EXTENDED) (T)	Optional		
SHIPITEM.AMOU NT (ITEM) (T)	Optional		
SHIPITEM.COUN TRYORG	Optional	WSH_DELIVERY_DETAIL S.COUNTRY_OF_ORIGIN	
SHIPITEM.DISPO SITN	Optional		
SHIPITEM.EXPOR TLIC	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
SHIPITEM.FRGH TCLS	Optional	WSH_DELIVERY_DETAIL S.FREIGHT_CLASS_CAT_I D converted to external value	
SHIPITEM.FRGH TITEM	Optional		
SHIPITEM.HAZR DMATL	Optional	WSH_DELIVERY_DETAIL S.HAZARD_CLASS_ID converted to external value. Pass it, but may not be used in the inbound side. Extract the data from tables. This value will be passed in the outbound transaction.	
SHIPITEM.IMPOR TLIC	Optional		
SHIPITEM.ITEMD ESC	Optional	WSH_DELIVERY_DETAIL S.ITEM_DESCRIPTION	
SHIPITEM.ITEMR V	Optional	WSH_DELIVERY_DETAIL S.REVISION	
SHIPITEM.ITEM_I D	Optional	MTL_SYSTEM_ITEMS_B.I NVENTORY_ITEM_ID	
SHIPITEM.ITEMR VX	Optional		
SHIPITEM.ITEMX	Optional	WSH_DELIVERY_DETAIL S.CUSTOMER_ITEM_ID Converted to Item_number by joining to mtl_customer_items	
SHIPITEM.LINEN UM	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
SHIPITEM.LOTLE VEL1 – LOTLEVEL2	Optional	Moved the mapping to INVDDETAIL section.	
SHIPITEM.NOTES 1 – NOTES9	Optional	Notes 1 WSH_DELIVERY_DETAIL S.PACKING_INSTRUCTIO NS	
SHIPITEM.OWNR SHPCODE	Optional	WSH_DELIVERY_DETAIL S.FOB_CODE	
SHIPITEM.PACKI NG	Optional		
SHIPITEM.PACKI NGDESC	Optional		
SHIPITEM.PARTI ALSHIP	Optional	"0" – means partial shipment not allowed (i.e when ship_set_id is not null) "1" – means partial shipment is allowed(i.e when ship_set_id is null)	
SHIPITEM.QUAN TITY (BACKORDERD)	Optional	Calculate WSH_DELIVERY_DETAIL S.REQUESTED_QUANTIT Y - WSH_DELIVERY_DETAIL S.SHIPPED_QUANTITY for 945 inbound only.	
SHIPITEM.QUAN TITY (OPEN)	Optional		
SHIPITEM.QUAN TITY (ORDERED)	Optional	WSH_DELIVERY_DETAIL S.SRC_REQUESTED_QUA NTITY	
SHIPITEM.QUAN TITY (SHIPPED)	Optional	WSH_DELIVERY_DETAIL S.SHIPPED_QUANTITY	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPITEM.QUANTITY (VOLUME)	Optional	WSH_DELIVERY_DETAIL S.VOLUME	
SHIPITEM.QUANTITY (WEIGHT)	Optional	WSH_DELIVERY_DETAIL S.NET_WEIGHT	
SHIPITEM.SHIPMATLID	Optional		
SHIPITEM.SHIPNOTES	Optional	WSH_DELIVERY_DETAIL S.SHIPPING_INSTRUCTIONS	
SHIPITEM.SHIPRIOR	Optional		
SHIPITEM.SHIPMTLDESC	Optional		
SHIPITEM.UPC	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
SHIPITEM.USERAREA	Optional	Use this to store: WSH_DELIVERY_DETAIL S.DELIVERY_DETAIL_ID, WSH_DELIVERY_ASSIGN MENTS.PARENT_DELIVE RY_DETAIL_ID, WSH_DELIVERY_DETAIL S.SHIP_TOLERANCE_AB OVE, WSH_DELIVERY_DETAIL S.SHIP_TOLERANCE_BEL OW, WSH_DELIVERY_DETAIL S.LOAD_SEQUENCE_NU MBER, WSH_DELIVERY_DETAIL S descriptive flexfields, WSH_DELIVERY_DETAIL S.SHIP_SET_ID, WSH_DELIVERY_DETAIL S.SUBINVENTORY, WSH_DELIVERY_DETAIL S.TOP_MODEL_LINE_ID, WSH_DELIVERY_DETAIL S.SHIP_MODEL_COMPLE TE_FLAG, WSH_DELIVERY_DETAIL S.COMMODITY_CODE_C AT_ID converted to code, WSH_DELIVERY_DETAIL S.SHIPMENT_PRIORITY_ CODE converted to "01", " 02", or "03" for XML, WSH_DELIVERY_DETAIL S.DELIVER_TO_LOCATIO N_ID, WSH_DELIVERY_DETAIL S.SOURCE_LINE_NUMBE R, WSH_DELIVERY_DETAIL S.SOURCE_LINE_ID	

Field Name	Required?	Transaction Table.Column Name	Description
DOCUMNTREF.	Optional		We need to send this segment for each customer purchase order number for all the delivery details on this delivery.
DOCUMNTREF.D OCTYPE	Required	"PURCHASEORDER"	We need to send this segment for each customer purchase order number for all the delivery details on this delivery.
DOCUMNTREF.D OCUMENTID	Required		
DOCUMNTREF.D OCTYPE	Required	WSH_DELIVERY_DETAIL S.CUST_PO_NUMBER "	PURCHASEORDER"
DOCUMNTREF.P ARTNRID	Required	WSH_NEW_DELIVERIES. ULTIMATE_DROPOFF_L OCATION_ID converted to external value for for Custome	
DOCUMNTREF.D OCUMENTID	Required	POWSH_DELIVERY_DET AILS.CUST_PO_NUMBER	
DOCUMNTREF.P ARTNRTYPE	Required	"ShipTo" - The delivery location of the customer	
DOCUMNTREF.P ARTNRID	Required	WSH_NEW_DELIVERIES. ULTIMATE_DROPOFF_L OCATION_ID converted to external value for for Customer PO	
DOCUMNTREF.D ESCRPTNDOCU MNTREF.PARTN RTYPE	Required	"ShipTo" - The delivery location of the customer	

Field Name	Required?	Transaction Table.Column Name	Description
DOCUMNTREF.D OCUMENTRVDO CUMNTREF.DES CRIPTN	Optional		
DOCUMNTREF.L INENUMDOCUM NTREF.DOCUME NTRV	Optional		
DOCUMNTREF.N OTES1 – NOTES9DOCUM NTREF.LINENU M	Optional		
DOCUMNTREF.S CHLINENUMDO CUMNTREF.NOT ES1 – NOTES9	Optional		
DOCUMNTREF.S UBLINENUMDO CUMNTREF.SCH LINENUM	Optional		
DOCUMNTREF.U SERAREADOCU MNTREF.SUBLIN ENUM	Optional		
DOCUMNTREF.U SERAREA	Optional		
CHARGE.	Optional		
CHARGE.CHARG EID	Required	WSH_FREIGHT_COSTS.FR EIGHT_COST_TYPE_ID converted to external value	

Field Name	Required?	Transaction Table.Column Name	Description
CHARGE.OPERAMT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.CURRENCY_CODE WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	
CHARGE.AMOUNT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	
CHARGE.OPERAMT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.CURRENCY_CODE	
CHARGE.CHARGE TYPECHARGE. AMOUNT (EXTENDED) (T)	Optional	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	
CHARGE.CHGLINENUM CHARGE.CHARGE TYPE	Optional	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	
CHARGE.DESCRPTION CHARGE.CHGLINENUM	Optional		
CHARGE.USERAREA CHARGE.DESCRPTION	Optional	WSH_FREIGHT_COSTS descriptive flexfields	
INVDDETAIL.	Optional		
CHARGE.USERAREA	Optional	WSH_FREIGHT_COSTS descriptive flexfields	

Field Name	Required?	Transaction Table.Column Name	Description
INVDETAIL.DES CRIPITNINVDETA IL.	Optional	WSH_DELIVERY_DETAIL S.ITEM_DESCRIPTION	
INVDETAIL.DISP OSITNINVDETAI L.DESCRIPITN	Optional	WSH_DELIVERY_DETAIL S.ITEM_DESCRIPTION	
INVDETAIL.LOT LEVEL1 – LOTLEVEL2	Optional	WSH_DELIVERY_DETAIL S.LOT_NUMBER	
INVDETAIL.DISP OSITN	Optional	WSH_DELIVERY_DETAIL S.SUBLOT_NUMBER	
INVDETAIL.NOT ES1 - NOTES9	Optional	WSH_DELIVERY_DETAIL S.LOT_NUMBER	
INVDETAIL.LOT LEVEL1 – LOTLEVEL2	Optional	WSH_DELIVERY_DETAIL S.SUBLOT_NUMBER	
INVDETAIL.QUA NTITY (ITEM)INVDETAI L.NOTES1 - NOTES9	Optional	WSH_DELIVERY_DETAIL S.REQUESTED QUANTITY/WSH_DELIVE RY_DETAILS.REQUESTED _QUANTITY_UOM	
INVDETAIL.SERI ALNUMINVDET AIL.QUANTITY (ITEM)	Optional	WSH_DELIVERY_DETAIL S.SERIAL_NUMBERWSH_ DELIVERY_DETAILS.REQ UESTED QUANTITY/WSH_DELIVE RY_DETAILS.REQUESTED _QUANTITY_UOM	
INVDETAIL.USE RAREA	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
INVDetail.SERIALNUM	Optional	WSH_DELIVERY_DETAIL S.PREFERRED_GRADE, WSH_DELIVERY_DETAIL S.TO_SERIAL_NUMBER, WSH_DELIVERY_DETAIL S.SERIAL_NUMBER	
INVDetail.USE RAREA	Optional	WSH_DELIVERY_DETAIL S.TO_SERIAL_NUMBER, WSH_DELIVERY_DETAIL S.PREFERRED_GRADE	
CHARGE.	Optional		
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	
CHARGE.OPERAMT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.CURRENCY_CODE WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	
CHARGE.AMOUNT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	
CHARGE.OPERAMT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.CURRENCY_CODE	
CHARGE.CHARGE TYPECHARGE. AMOUNT (EXTENDED) (T)	Optional	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	

Field Name	Required?	Transaction Table.Column Name	Description
CHARGE.CHGLI NENUMCHARGE .CHARGETYPE	Optional	WSH_FREIGHT_COSTS.FR EIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	
CHARGE.DESCRI PTNCHARGE.CH GLINENUM	Optional		
CHARGE.USERA REA CHARGE.DESCRI PTN	Optional	WSH_FREIGHT_COSTS descriptive flexfields	
CHARGE.USERA REA	Optional	WSH_FREIGHT_COSTS descriptive flexfields	
DISTRIBUTION.	Optional		
DISTRIBUTION.B USNAREADISTRI BUTION.	Optional		
DISTRIBUTION.C OSTCENTERDIST RIBUTION.BUSN AREA	Optional		
DISTRIBUTION.D EPARTMENTDIS TRIBUTION.COS TCENTER	Optional		
DISTRIBUTION.D IVISIONDISTRIB UTION.DEPART MENT	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
DISTRIBUTION.ELEMENT1 - ELEMENT 999DISTRIBUTION.DIVISION	Optional		
DISTRIBUTION.FUNDDISTRIBUTION.ELEMENT1 - ELEMENT 999	Optional		
DISTRIBUTION.GEOGRAPHYDISTRIBUTION.FUND	Optional		
DISTRIBUTION.GLENTITYSDISTRIBUTION.GEOGRAPHY	Optional		
DISTRIBUTION.GLENNOMACCTDISTRIBUTION.GLENTITYS	Optional		
DISTRIBUTION.OPERAMT (EXTENDED) (T)DISTRIBUTION.GLENNOMACCT	Optional		
DISTRIBUTION.PROFITCTRDISTRIBUTION.OPERAMT (EXTENDED) (T)	Optional		
DISTRIBUTION.PROJECTDISTRIBUTION.PROFITCTR	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
DISTRIBUTION.UNITDISTRIBUTION.PROJECT	Optional		
DISTRIBUTION.WAREHOUSEDISTRIBUTION.UNIT	Optional		
DISTRIBUTION.UNITSERAREADISTRIBUTION.WAREHOUSE	Optional		
DISTRIBUTION.UNITSERAREA	Optional		
SHIPUNIT.	Optional		
SHIPUNIT.CARRIER	Required	WSH_DELIVERY_DETAIL S.CARRIER_ID converted to external value	
SHIPUNIT.SHIPPERID	Required	WSH_DELIVERY_DETAIL S.SHIP_FROM_LOCATION_ID converted to external value.	
SHIPUNIT.CARRIER	Required	WSH_DELIVERY_DETAIL S.CARRIER_ID converted to external value	
SHIPUNIT.TRACKINGID	Required	WSH_DELIVERY_DETAIL S.TRACKING_NUMBER	
SHIPUNIT.SHIPPERID	Required	WSH_DELIVERY_DETAIL S.SHIP_FROM_LOCATION_ID converted to external value.	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.AMOUNT (DECLAREVAL)SHIPUNIT.TRACKINGID	Required	WSH_DELIVERY_DETAIL S.TRACKING_NUMBER	
SHIPUNIT.AMOUNT (FREIGHT) (T)SHIPUNIT.AMOUNT (DECLAREVAL)	Optional		
SHIPUNIT.AMOUNT (ITEM) (T)SHIPUNIT.AMOUNT (FREIGHT) (T)	Optional		
SHIPUNIT.AMOUNT (TOTAL) (T)SHIPUNIT.AMOUNT (ITEM) (T)	Optional		
SHIPUNIT.CARRSRVLVLSHIPUNIT.AMOUNT (TOTAL) (T)	Optional		
SHIPUNIT.CONTAINRIDSHIPUNIT.CARRSRVLVL	Optional	WSH_DELIVERY_DETAIL S.DELIVERY_DETAIL_ID	
SHIPUNIT.CONTNRGRPIDSHIPUNIT.CONTAINRID	Optional	WSH_DELIVERY_DETAIL S.MASTER_CONTAINER_ ITEM_ID converted to external valueWSH_DELIVERY_DE TAILS.DELIVERY_DETAIL_ _ID	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.CON TNRSEALSHIPUN T.CONTNRGRPI D	Optional	WSH_DELIVERY_DETAIL S.SEAL_CODEWSH_DELI VERY_DETAILS.MASTER_ CONTAINER_ITEM_ID converted to external value	
SHIPUNIT.CON RTYPESHIPUNIT. CONTNRSEAL	Optional	WSH_DELIVERY_DETAIL S.CONTAINER_TYPE_CO DE WSH_DELIVERY_DETAIL S.SEAL_CODE	
SHIPUNIT.COUN TRYDSTSHIPUNI T.CONTRTYPE	Optional	WSH_DELIVERY_DETAIL S.CONTAINER_TYPE_CO DE	
SHIPUNIT.COUN TRYORGSHIPUNI T.COUNTRYDST	Optional	WSH_DELIVERY_DETAIL S.COUNTRY_OF_ORIGIN	
SHIPUNIT.DATE TIME (DELIVACT)SHIP UNIT.COUNTRY ORG	Optional	WSH_TRIP_STOPS.ACTU AL_ARRIVAL_DATEWSH _DELIVERY_DETAILS.CO UNTRY_OF_ORIGIN	
SHIPUNIT.DATE TIME (DELIVSCHED)S HIPUNIT.DATETI ME (DELIVACT)	Optional	WSH_DELIVERY_DETAIL S.DATE_SCHEDULEDWS H_TRIP_STOPS.ACTUAL_ ARRIVAL_DATE	
SHIPUNIT.DATE TIME (LOADING)SHIP UNIT.DATETIME (DELIVSCHED)	Optional	WSH_DELIVERY_DETAIL S.DATE_SCHEDULED	
SHIPUNIT.DATE TIME (SHIP)SHIPUNIT. DATETIME (LOADING)	Optional	WSH_TRIP_STOPS.ACTU AL_DEPARTURE_DATE	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.DESCR IPTNSHIPUNIT.D ATETIME (SHIP)	Optional	WSH_TRIP_STOPS.ACTU AL_DEPARTURE_DATE	
SHIPUNIT.EXPO RTLICSHIPUNIT. DESCRIPTN	Optional		
SHIPUNIT.FRGH TCLSSHIPUNIT.E XPORTLIC	Optional	WSH_DELIVERY_DETAIL S.FREIGHT_CLASS_CAT_I D converted to external value	
SHIPUNIT.FRCH TITEMSHIPUNIT. FRGHTCLS	Optional	WSH_DELIVERY_DETAIL S.FREIGHT_CLASS_CAT_I D converted to external value	
SHIPUNIT.FRGT HTERMSSHIPUN IT.FRCHTITEM	Optional	WSH_DELIVERY_DETAIL S.FREIGHT_TERMS_COD E converted to external value	
SHIPUNIT.HAZR DMATLSHIPUNI T.FRGTHTERMS	Optional	WSH_DELIVERY_DETAIL S.HAZARD_CLASS_ID converted to external valueWSH_DELIVERY_DE TAILS.FREIGHT_TERMS_ CODE converted to external value	
SHIPUNIT.IMPO RTLICSHIPUNIT. HAZRDMATL	Optional	WSH_DELIVERY_DETAIL S.HAZARD_CLASS_ID converted to external value	
SHIPUNIT.LOAD POINTSHIPUNIT. IMPORTLIC	Optional		
SHIPUNIT.NOTE S1 - NOTES9SHIPUNI T.LOADPOINT	Optional	WSH_DELIVERY_DETAIL S.PACKING_INSTRUCTIO NS	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.OPER AMT (FREIGHT) (T)SHIPUNIT.NO TES1 - NOTES9	Optional	WSH_DELIVERY_DETAIL S.PACKING_INSTRUCTIO NS	
SHIPUNIT.PACKI NGSHIPUNIT.OP ERAMT (FREIGHT) (T)	Optional		
SHIPUNIT.PACKI NGDESCSHIPUN IT.PACKING	Optional		
SHIPUNIT.QUAN TITY (ESTWEIGHT)SHI PUNIT.PACKING DESC	Optional		
SHIPUNIT.QUAN TITY (HEIGHT)SHIPU NIT.QUANTITY (ESTWEIGHT)	Optional	Populate from MTL_SYSTEM_ITEMS	
SHIPUNIT.QUAN TITY (LENGTH)SHIPU NIT.QUANTITY (HEIGHT)	Optional	Populate from MTL_SYSTEM_ITEMS Populate from MTL_SYSTEM_ITEMS	
SHIPUNIT.QUAN ITY (LOADINGWT)S HIPUNIT.QUANT ITY (LENGTH)	Optional	Calculate WSH_DELIVERY_DETAIL S.GROSS WEIGHT - WSH_DELIVERY_DETAIL S.NET WEIGHT Populate from MTL_SYSTEM_ITEMS	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.QUAN TITY (NETWEIGHT)SH IPUNIT.QUANIT TY (LOADINGWT)	Optional	WSH_DELIVERY_DETAIL S.NET_WEIGHT + WSH_DELIVERY_DETAIL S.WEIGHT_UOM_CODE for delivery lines = container item, Calculate WSH_DELIVERY_DETAIL S.GROSS WEIGHT - WSH_DELIVERY_DETAIL S.NET WEIGHT	
SHIPUNIT.QUAN TITY (TOTWEIGHT)SH IPUNIT.QUANTI TY (NETWEIGHT)	Optional	WSH_DELIVERY_DETAIL S.GROSS_WEIGHT + WSH_DELIVERY_DETAIL S.WEIGHT_UOM_CODEW SH_DELIVERY_DETAILS. NET_WEIGHT + WSH_DELIVERY_DETAIL S.WEIGHT_UOM_CODE for delivery lines = container item	
SHIPUNIT.QUAN TITY (VOLUME)SHIPU NIT.QUANTITY (TOTWEIGHT)	Optional	WSH_DELIVERY_DETAIL S.VOLUME + WSH_DELIVERY_DETAIL S.VOLUME_UOM_CODE for delivery lines = container itemWSH_DELIVERY_DET AILS.GROSS_WEIGHT + WSH_DELIVERY_DETAIL S.WEIGHT_UOM_CODE	
SHIPUNIT.QUAN TITY (WIDTH)	Optional	WSH_DELIVERY_DETAIL S.VOLUME + WSH_DELIVERY_DETAIL S.VOLUME_UOM_CODE for delivery lines = container item	
SHIPUNIT.QUAN TITY (VOLUME)	Optional	WSH_DELIVERY_DETAIL S.GROSS_WEIGHT + WSH_DELIVERY_DETAIL S.WEIGHT_UOM_CODE	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.SHIPMATLIDSHIPUNIT. QUANTITY (WIDTH)	Optional	WSH_DELIVERY_DETAIL S.INVENTORY_ITEM_ID converted to Item Number by joining to mtl_system_items (this delivery detail id is the delivery detail id of the container record)	
SHIPUNIT.SHIPNOTESSHIPUNIT.S HIPMATLID	Optional	WSH_DELIVERY_DETAIL S.SHIPPING_INSTRUCTIONS NSWSH_DELIVERY_DETAILS.INVENTORY_ITEM_ ID converted to Item Number by joining to mtl_system_items (this delivery detail id is the delivery detail id of the container record)	
SHIPUNIT.SHIPPINGNOTESSHIPUNIT.S HIPNOTES	Optional	WSH_DELIVERY_DETAIL S.DELIVER_TO_LOCATION_IDWSH_DELIVERY_DETAILS.SHIPPING_INSTRUCTIONS	
SHIPUNIT.SHIPSRVLSHIPUNIT. .SHIPPOINT	Optional	WSH_DELIVERY_DETAIL S.SHIP_METHOD_CODE WSH_DELIVERY_DETAIL S.DELIVER_TO_LOCATION_ID	
SHIPUNIT.SHIPMTLDESCSHIPUNIT. T.SHIPSRVLVL	Optional	WSH_DELIVERY_DETAILS .ITEM_DESCRIPTIONWSH_DELIVERY_DETAILS.S HIP_METHOD_CODE	
SHIPUNIT.SHIPUNITSEQSHIPUNIT. T.SHIPMTLDESC	Optional	WSH_DELIVERY_DETAILS .ITEM_DESCRIPTION	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.SHPU NITSTSSHIPUNIT .SHPUNITSEQ	Optional		
SHIPUNIT.SHPU NITTOTSHIPUNI T.SHPUNITSTS	Optional		
SHIPUNIT.SPECI ALHNDSHIPUNI T.SHPUNITTOT	Optional		
SHIPUNIT.STAGE POINTSHIPUNIT. SPECIALHND	Optional		
SHIPUNIT.TEMP RATURE (ACTUAL)SHIPU NIT.STAGEPOIN T	Optional		
SHIPUNIT.TEMP RATURE (DELIVERY)SHIP UNIT.TEMPRATU RE (ACTUAL)	Optional		
SHIPUNIT.TEMP RATURE (LOADING)SHIP UNIT.TEMPRATU RE (DELIVERY)	Optional		
SHIPUNIT.TRAC KINGIDTYPESHI PUNIT.TEMPRAT URE (LOADING)	Optional		
SHIPUNIT.TRAN SMETHDSHIPUN IT.TRACKINGIDT YPE	Optional	WSH_DELIVERY_DETAIL S.SHIP_METHOD_CODE Write Procedure to Populate this column.	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.TRAN STERMSSHIPUNI T.TRANSMETHD	Optional	WSH_NEW_DELIVERIES. FOB_CODEWSH_DELIVE RY_DETAILS.SHIP_METH OD_CODE	
SHIPUNIT.USER AREASHIPUNIT. TRANSTERMS	Optional	WSH_DELIVERY_DETAIL S.DELIVERY_DETAIL_ID (To be called as DELIVERY_DETAIL_NUM BER) WSH_DELIVERY_DETAIL S.PACKING_INSTRUCTIO NS WSH_DELIVERY_DETAIL S.DESSCRIPTIVE_FLEXFIEL DS WSH_DELIVERY_DETAIL S.LOT_NUMBER WSH_DELIVERY_DETAIL S.SUBLOT_NUMBER WSH_DELIVERY_DETAIL S.REVISION WSH_DELIVERY_DETAIL S.SERIAL NUMBER WSH_DELIVERY_DETAIL S.TO_SERIAL NUMBER WSH_NEW_DELIVERIES. FOB_CODE	

Field Name	Required?	Transaction Table.Column Name	Description
SHIPUNIT.USER AREA	Optional	WSH_DELIVERY_DETAIL S.DELIVERY_DETAIL_ID (To be called as DELIVERY_DETAIL_NUM BER) WSH_DELIVERY_DETAIL S.PACKING_INSTRUCTIO NS WSH_DELIVERY_DETAIL S.DESRIPTIVE_FLEXFIEL DS WSH_DELIVERY_DETAIL S.LOT_NUMBER WSH_DELIVERY_DETAIL S.SUBLOT_NUMBER WSH_DELIVERY_DETAIL S.REVISION WSH_DELIVERY_DETAIL S.SERIAL NUMBER WSH_DELIVERY_DETAIL S.TO_SERIAL NUMBER	
INVITEM.	Optional	If the Ship Unit is a container, these are the details about the item(s) in the container.	
INVITEM.ITEM INVITEM.	Required	If the Ship Unit is a container, these are the details about the item(s) in the container.	
INVITEM.QUAN TITY (ITEM) (REQUIRED)	Required		
INVITEM.ITEM	Required		
INVITEM.AMOU NT (EXTENDED) (T)INVITEM.QUA NTITY (ITEM)	Required		

Field Name	Required?	Transaction Table.Column Name	Description
INVITEM.AMOUNT (ITEM) (T)INVITEM.AMOUNT (EXTENDED) (T)	Optional		
INVITEM.COUNT RY ORGINVITEM.A MOUNT (ITEM) (T)	Optional		
INVITEM.FRGT CLSINVITEM.CO UNTRY ORG	Optional		
INVITEM.FRGT TEMINVITEM.FR GHTCLS	Optional		
INVITEM.HAZRD MATLINVITEM.F RGHTITEM	Optional		
INVITEM.ITEMD ESCINVITEM.HA ZRDMATL	Optional		
INVITEM.ITEMR VINVITEM.ITEM DESC	Optional		
INVITEM.ITEMR VXINVITEM.ITE MRV	Optional		
INVITEM.LOTLE VEL1 - LOTLEVEL2INVI TEM.ITEMRVX	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
INVITEM.NOTES 1 - NOTES 9INVITEM.LOTLE VEL1 - LOTLEVEL2	Optional		
INVITEM.OWNR SHPCDEINVITE M.NOTES 1 - NOTES 9	Optional		
INVITEM.QUAN TITY (VOLUME)INVIT EM.OWNRSHPC DE	Optional		
INVITEM.QUAN TITY (WEIGHT)INVITE M.QUANTITY (VOLUME)	Optional		
INVITEM.SHIPN OTESINVITEM.Q UANTITY (WEIGHT)	Optional		
INVITEM.UPCIN VITEM.SHIPNOT ES	Optional		
INVITEM.USERA REAINVITEM.UP C	Optional		
INVITEM.USERA REA	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
CONTAINER.	Optional		This is used to send intermediate packaging info. We do not have the ability to send intermediate packaging within the shipping unit, so we will not use this segment.
CONTAINER.CO NTAINRIDCONT AINER.	Optional	WSH_DELIVERY_DETAIL S.CONTAINER_NAME	This is used to send intermediate packaging info. We do not have the ability to send intermediate packaging within the shipping unit, so we will not use this segment.
CONTAINER.CO NTNRSEALCONT AINER.CONTAIN RID	Optional	WSH_DELIVERY_DETAIL S.SEAL_CODEWSH_DELI VERY_DETAILS.CONTAI NER_NAME	
CONTAINER.CO NTNRTYPECONT AINER.CONTNR SEAL	Optional	WSH_DELIVERY_DETAIL S.CONTAINER_TYPE_CO DE WSH_DELIVERY_DETAIL S.SEAL_CODE	
CONTAINER.DES CRIPTNCONTAI NER.CONTNRTY PE	Optional	WSH_DELIVERY_DETAIL S.CONTAINER_TYPE_CO DE	
CONTAINER.NO TES 1 - NOTES 9CONTAINER.DE SCRIPTN	Optional		
CONTAINER.PA CKINGCONTAIN ER.NOTES 1 - NOTES 9	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
CONTAINER.PACKINGDESCCONTAINER.PACKING	Optional		
CONTAINER.PACKINGDESCCONTAINER.PACKINGDESC	Optional		
CONTAINER.SHIPMENTIDCONTAINER.PARCNTNRID	Optional	WSH_DELIVERY_DETAIL S.INVENTORY_ITEM_ID converted to ITEM NUMBER by joining to mtl_system_items	
CONTAINER.SHIPMENTIDCONTAINER.SHIPMENTID	Optional	WSH_DELIVERY_DETAIL S.ITEM_DESCRIPTION WSH_DELIVERY_DETAIL S.INVENTORY_ITEM_ID converted to ITEM NUMBER by joining to mtl_system_items	

Field Name	Required?	Transaction Table.Column Name	Description
CONTAINER.USE RAREACONTAIN ER.SHPMTLDESC	Optional	WSH_DELIVERY_ASSIGN MENTS.DELIVERY_DET AIL_ID (To be called as DELIVERY_DETAIL_NUM BER) WSH_DELIVERY_ASSIGN MENTS.PARENT_DELIVE RY_DETAIL_ID (To be called as PARENT_DELIVERY_DET AIL_NUMBER) HEIGHT – From mtl_system_items LENGTH - From mtl_system_items WEIGHT- WSH_DELIVERY_DETAIL S. GROSS_WEIGHT WSH_DELIVERY_DETAIL S. NET_WEIGHT VOLUME – WSH_DELIVERY_DETAIL S. VOLUME WIDTH- From mtl_system_items LOT- WSH_DELIVERY_DETAIL S. LOT_NUMBER SUBLOT- WSH_DELIVERY_DETAIL S. SUBLOT_NUMBER WSH_DELIVERY_DETAIL S. REVISION WSH_DELIVERY_DETAIL S. SERIAL NUMBER WSH_DELIVERY_DETAIL S. TO_SERIAL NUMBER WSH_DELIVERY_DETAIL S. SHIP_FROM_LOCATION_ ID converted to code (To be called as SHIP_FROM_LOCATION LOCATION) WSH_DELIVERY_DETAIL S. DELIVER_TO_LOCATION _ID converted to code (To	

Field Name	Required?	Transaction Table.Column Name	Description
		be called as DELIVER_TO_LOCATION) WSH_DELIVERY_DETAIL S. TRACKING_NUMBER WSH_DELIVERY_DETAIL S. COUNTRY_OF_ORIGIN WSH_TRIP_STOPS.ACTU AL_ ARRIVAL_DATE WSH_DELIVERY_DETAIL S. DATE_SCHEDULED WSH_TRIP_STOPS.ACTU AL_ DEPARTURE_DATE WSH_DELIVERY_DETAIL S. SHIP_METHOD_CODE WSH_DELIVE	

Field Name	Required?	Transaction Table.Column Name	Description
CONTAINER.USE RAREA		WSH_DELIVERY_ASSIGN MENTS.DELIVERY_DETA IL_ID (To be called as DELIVERY_DETAIL_NUM BER) WSH_DELIVERY_ASSIGN MENTS.PARENT_DELIVE RY_DETAIL_ID (To be called as PARENT_DELIVERY_DET AIL_NUMBER) HEIGHT - From mtl_system_items LENGTH - From mtl_system_items WEIGHT- WSH_DELIVERY_DETAIL S. GROSS_WEIGHT WSH_DELIVERY_DETAIL S. NET_WEIGHT VOLUME - WSH_DELIVERY_DETAIL S. VOLUME WIDTH- From mtl_system_items LOT- WSH_DELIVERY_DETAIL S. LOT_NUMBER SUBLOT- WSH_DELIVERY_DETAIL S. SUBLOT_NUMBER WSH_DELIVERY_DETAIL S. REVISION WSH_DELIVERY_DETAIL S. SERIAL NUMBER WSH_DELIVERY_DETAIL	

Field Name	Required?	Transaction Table.Column Name	Description
		S. TO_SERIAL NUMBER	
		WSH_DELIVERY_DETAIL S. SHIP_FROM_LOCATION_ ID converted to code (To be called as SHIP_FROM_LOCATION LOCATION)	
		WSH_DELIVERY_DETAIL S. DELIVER_TO_LOCATION _ID converted to code (To be called as DELIVER_TO_LOCATION)	
		WSH_DELIVERY_DETAIL S. TRACKING_NUMBER	
		WSH_DELIVERY_DETAIL S. COUNTRY_OF_ORIGIN	
		WSH_TRIP_STOPS.ACTU AL_ ARRIVAL_DATE	
		WSH_DELIVERY_DETAIL S. DATE_SCHEDULED	
		WSH_TRIP_STOPS.ACTU AL_ DEPARTURE_DATE	
		WSH_DELIVERY_DETAIL S. SHIP_METHOD_CODE	
		WSH_DELIVERY_DETAIL S. FREIGHT_CLASS_CAT_ID converted to external value (To be called as FREIGHT_CLASS_CATEG ORY)	
		WSH_DELIVERY_DETAIL S. FREIGHT_TERMS_CODE converted to external value	

Field Name	Required?	Transaction Table.Column Name	Description
		WSH_DELIVERY_DETAIL S. HAZARD_CLASS_ID converted to external value (To be called as HAZARD_CLASS or as HAZARD_CLASS_CODE)	
		WSH_DELIVERY_DETAIL S. FOB_CODE	
		WSH_DELIVERY_DETAIL S. SHIPPING_INSTRUCTION S	
		WSH_DELIVERY_DETAIL S. PACKING_INSTRUCTION S	
		WSH_DELIVERY_DETAIL S. Descriptive Flexfields	
CHARGE.	Required	This CHARGE section is for a CONTAINER.	
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external valueThis CHARGE section is for a CONTAINER.	
CHARGE.OPERATIONAL_AMOUNT (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.CURRENCY_CODEWSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	
CHARGE.OPERATIONAL_AMOUNT (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	

Field Name	Required?	Transaction Table.Column Name	Description
CHARGE.OPERA MT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.C URRENCY_CODE	
CHARGE.CHARG ETYPECHARGE. AMOUNT (EXTENDED) (T)	Optional	WSH_FREIGHT_COSTS.FR EIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	
CHARGE.CHGLI NENUMCHARGE .CHARGETYPE	Optional	WSH_FREIGHT_COSTS.FR EIGHT_COST_TYPE_ID converted to FREIGHT_COST_TYPE	
CHARGE.DESCR PTNCHARGE.CH GLINENUM	Optional		
CHARGE.USERA REA CHARGE.DESCR PTN	Optional	WSH_FREIGHT_COSTS descriptive flexfields	
CHARGE.USERA REA	Optional	WSH_FREIGHT_COSTS descriptive flexfields	
INVITEM.	Optional		
INVITEM.ITEM INVITEM.	Required		
INVITEM.QUAN TITY (ITEM)	Required		
INVITEM.ITEM	Required		
INVITEM.AMOU NT (EXTENDED) (T)	Required		
INVITEM.QUAN TITY (ITEM)	Required		

Field Name	Required?	Transaction Table.Column Name	Description
INVITEM.AMOU NT (ITEM) (T)	Optional		
INVITEM.AMOU NT (EXTENDED) (T)	Optional		
INVITEM.COUNT RY ORG	Optional		
INVITEM.AMOU NT (ITEM) (T)	Optional		
INVITEM.FRGHT CLS	Optional		
INVITEM.COUNT RY ORG	Optional		
INVITEM.FRGHTI TEM	Optional		
INVITEM.FRGHT CLS	Optional		
INVITEM.HAZRD MATL	Optional		
INVITEM.FRGHTI TEM	Optional		
INVITEM.ITEMD ESC	Optional		
INVITEM.HAZRD MATL	Optional		
INVITEM.ITEMR V	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
INVITEM.ITEMD ESC	Optional		
INVITEM.ITEMR VX	Optional		
INVITEM.ITEMR V	Optional		
INVITEM.LOTLE VEL1 - LOTLEVEL2	Optional		
INVITEM.ITEMR VX	Optional		
INVITEM.NOTES 1 - NOTES 9	Optional		
INVITEM.LOTLE VEL1 - LOTLEVEL2	Optional		
INVITEM.OWNR SHPCDE	Optional		
INVITEM.NOTES 1 - NOTES 9	Optional		
INVITEM.QUAN TITY (VOLUME)	Optional		
INVITEM.OWNR SHPCDE	Optional		
INVITEM.QUAN TITY (WEIGHT)	Optional		
INVITEM.QUAN TITY (VOLUME)	Optional		

Field Name	Required?	Transaction Table.Column Name	Description
INVITEM.SHIPNOTES	Optional		
INVITEM.QUANTITY (WEIGHT)	Optional		
INVITEM.UPC	Optional		
INVITEM.SHIPNOTES	Optional		
INVITEM.USERAREA	Optional		
INVITEM.UPC	Optional		
INVITEM.USERAREA	Optional		
CHARGE.	Optional	This CHARGE section is for SHIPUNIT.	
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value This CHARGE section is for SHIPUNIT.	
CHARGE.OPERAMOUNT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	
CHARGE.CHARGEID	Required	WSH_FREIGHT_COSTS.CURRENCY_CODE WSH_FREIGHT_COSTS.FREIGHT_COST_TYPE_ID converted to external value	
CHARGE.AMOUNT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.TOTAL_AMOUNT	

Field Name	Required?	Transaction Table.Column Name	Description
CHARGE.OPERA MT (EXTENDED) (T)	Required	WSH_FREIGHT_COSTS.C URRENCY_CODE	
CHARGE.CHARG ETYPE	Optional	WSH_FREIGHT_COSTS.FR EIGHT_COST_TYPE_ID	
CHARGE.AMOU NT (EXTENDED) (T)	Optional	converted to FREIGHT_COST_TYPE	
CHARGE.CHGLI NENUM	Optional	WSH_FREIGHT_COSTS.FR EIGHT_COST_TYPE_ID	
CHARGE.CHARG ETYPE	Optional	converted to FREIGHT_COST_TYPE	
CHARGE.DESCRI PTN	Optional		
CHARGE.CHGLI NENUM	Optional		
CHARGE.DESCRI PTN	Optional		
CHARGE.USERA REA	Optional	WSH_FREIGHT_COSTS descriptive flexfields	
CHARGE.USERA REA	Optional	WSH_FREIGHT_COSTS descriptive flexfields	

```

<?xml version="1.0" encoding = "UTF-8"?><SHOW_SHIPMENT_005>
  <CNTROLAREA>
    <BSR>
      <VERB>SHOW</VERB>
      <NOUN>SHIPMENT</NOUN>
      <REVISION>001</REVISION>
    </BSR>
    <SENDER>
      <LOGICALID>ORACLE</LOGICALID>
      <COMPONENT>ITM Application</COMPONENT>
      <TASK>SSNO</TASK>
      <REFERENCEID>13784~WSH_EXPORT_COMPLIANCE~61</REFERENCEID>
      <CONFIRMATION>Y</CONFIRMATION>
      <LANGUAGE>US</LANGUAGE>
      <CODEPAGE>UTF8</CODEPAGE>
      <AUTHID>vsathiya-venkat</AUTHID>
    </SENDER>
    <DATETIME qualifier="CREATION">
      <YEAR>2004</YEAR>
      <MONTH>06</MONTH>
      <DAY>18</DAY>
      <HOUR>08</HOUR>
      <MINUTE>31</MINUTE>
      <SECOND>40</SECOND>
      <SUBSECOND>0000</SUBSECOND>
      <TIMEZONE>+00:00</TIMEZONE>
    </DATETIME>
  </CNTROLAREA>
  <DATAAREA>
    <SHOW_SHIPMENT>
      <SHIPMENT>
        <DATETIME qualifier="DOCUMENT">
          <YEAR>2004</YEAR>
          <MONTH>06</MONTH>
          <DAY>18</DAY>
          <HOUR>08</HOUR>
          <MINUTE>31</MINUTE>
          <SECOND>35</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+00:00</TIMEZONE>
        </DATETIME>
        <DATETIME qualifier="CREATION">
          <YEAR>2004</YEAR>
          <MONTH>06</MONTH>
          <DAY>18</DAY>
          <HOUR>08</HOUR>
          <MINUTE>27</MINUTE>
          <SECOND>01</SECOND>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+00:00</TIMEZONE>
        </DATETIME>
        <DATETIME qualifier="ACTUAL_ARRIVAL">
          <YEAR/>
          <MONTH/>
          <DAY/>
          <HOUR/>
          <MINUTE/>
          <SECOND/>
          <SUBSECOND>0000</SUBSECOND>
          <TIMEZONE>+00:00</TIMEZONE>
        </DATETIME>
      </SHIPMENT>
    </SHOW_SHIPMENT>
  </DATAAREA>

```

```

<DATETIME qualifier="ULTIMATE_DROPOFF">
  <YEAR>2004</YEAR>
  <MONTH>06</MONTH>
  <DAY>18</DAY>
  <HOUR>11</HOUR>
  <MINUTE>59</MINUTE>
  <SECOND>00</SECOND>
  <SUBSECOND>0000</SUBSECOND>
  <TIMEZONE>+00:00</TIMEZONE>
</DATETIME>
<DATETIME qualifier="ACTUAL_DEPARTURE">
  <YEAR/>
  <MONTH/>
  <DAY/>
  <HOUR/>
  <MINUTE/>
  <SECOND/>
  <SUBSECOND>0000</SUBSECOND>
  <TIMEZONE>+00:00</TIMEZONE>
</DATETIME>
<DATETIME qualifier="INITIAL_PICKUP">
  <YEAR>2004</YEAR>
  <MONTH>06</MONTH>
  <DAY>18</DAY>
  <HOUR>08</HOUR>
  <MINUTE>25</MINUTE>
  <SECOND>16</SECOND>
  <SUBSECOND>0000</SUBSECOND>
  <TIMEZONE>+00:00</TIMEZONE>
</DATETIME>
<QUANTITY qualifier="ESTIMATED_WEIGHT">
  <VALUE>3</VALUE>
  <NUMOFDEC/>
  <SIGN/>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="LOADING_WEIGHT">
  <VALUE>0</VALUE>
  <NUMOFDEC/>
  <SIGN/>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="NET_WEIGHT">
  <VALUE>3</VALUE>
  <NUMOFDEC/>
  <SIGN/>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="SHIP_UNIT">
  <VALUE/>
  <NUMOFDEC/>
  <SIGN/>
  <UOM/>
</QUANTITY>
<QUANTITY qualifier="TOTAL_WEIGHT">
  <VALUE>3</VALUE>
  <NUMOFDEC/>
  <SIGN/>
  <UOM>Lbs</UOM>
</QUANTITY>
<QUANTITY qualifier="VOLUME">

```

```

<VALUE>3</VALUE>
    <NUMOFDEC/>
    <SIGN/>
    <UOM>FT3</UOM>
</QUANTITY>
<AMOUNT qualifier="FREIGHT">
    <VALUE/>
    <NUMOFDESC/>
    <SIGN/>
    <CURRENCY/>
    <DRCR/>
</AMOUNT>
<DOCUMENTID>918</DOCUMENTID>
<SHIPPERID>Beijing</SHIPPERID>
<SYNCID>A</SYNCID>
<CARRIER/>
<CARRSRVLVL/>
<COUNTRYDST>CN</COUNTRYDST>
<DESCRIPTN/>
<DOCTYPE>SS</DOCTYPE>
<DOCUMENTRV>001</DOCUMENTRV>
<SHIPPOINT>Atlanta (OPS)</SHIPPOINT>
<SHIPSRLVL/>
<TRANSMETHD/>
<TRANSTERMS/>
<WAREHOUSE>IM1</WAREHOUSE>
<USERAREA>
    <ORACLE.SHOW_SHIPMENT_005.SHIPMENT.USERAREA>
        <ORACLE.DELIVERY_NAME>63771</ORACLE.DELIVERY_NAME>
        <ORACLE.LOADING_SEQUENCE/>
        <ORACLE.VEHICLE_NUM_PREFIX/>
        <ORACLE.VEHICLE_NUMBER/>
        <ORACLE.ROUTE_ID/>
        <ORACLE.ROUTING_INSTRUCTIONS/>
        <ORACLE.DEPARTURE_SEAL_CODE/>
        <ORACLE.ORIG_DOCID/>
        <ORACLE.CUSTOMER_NAME>Computer Service and
Rentals</ORACLE.CUSTOMER_NAME>
        <ORACLE.CUSTOMER_NUMBER>1006</ORACLE.CUSTOMER_NUMBER>
        <ORACLE.WAYBILL/>
        <ORACLE.DESCRPTIVE_FLEXFIELD>
            <ORACLE.ATTRIBUTE_CATEGORY/>
            <ORACLE.ATTRIBUTE1/>
            <ORACLE.ATTRIBUTE2/>
            <ORACLE.ATTRIBUTE3/>
            <ORACLE.ATTRIBUTE4/>
            <ORACLE.ATTRIBUTE5/>
            <ORACLE.ATTRIBUTE6/>
            <ORACLE.ATTRIBUTE7/>
            <ORACLE.ATTRIBUTE8/>
            <ORACLE.ATTRIBUTE9/>
            <ORACLE.ATTRIBUTE10/>
            <ORACLE.ATTRIBUTE11/>
            <ORACLE.ATTRIBUTE12/>
            <ORACLE.ATTRIBUTE13/>
            <ORACLE.ATTRIBUTE14/>
            <ORACLE.ATTRIBUTE15/>
        </ORACLE.DESCRPTIVE_FLEXFIELD>
    </ORACLE.SHOW_SHIPMENT_005.SHIPMENT.USERAREA>
</USERAREA>
<PARTNER>

```

```

<NAME>ITMTESTORG1</NAME>
  <PARTNRTYPE>SHIP_FROM</PARTNRTYPE>
  <PARTNERID>IM1</PARTNERID>
  <CURRENCY/>
  <DUNS_NUMBER>0</DUNS_NUMBER>
  <PARTNIDX>0</PARTNIDX>
  <USERAREA>
    <ORACLE.SHOW_SHIPMENT_005.SHIPMENT.PARTNER.USERAREA>
      <ORACLE.INTMED_SHIP_TO_LOCATION/>
      <ORACLE.POOLED_SHIP_TO_LOCATION/>
    </ORACLE.SHOW_SHIPMENT_005.SHIPMENT.PARTNER.USERAREA>
  </USERAREA>
  <ADDRESS>
    <ADDRLINE index="1">Jia 1</ADDRLINE>
    <ADDRLINE index="2">Yuantanbeiju Dongkou</ADDRLINE>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRTYPE/>
    <CITY>Xicheng</CITY>
    <COUNTRY>CN</COUNTRY>
    <COUNTY/>
    <POSTALCODE>100037</POSTALCODE>
    <REGION/>
    <STATEPROVN/>
  </ADDRESS>
  <CONTACT>
    <FAX>(01) 8512393</FAX>
    <TELEPHONE>(01) 8512211</TELEPHONE>
    <URL/>
  </CONTACT>
  <DOCUMNTREF>
    <DOCTYPE>SHIPMENTREQUEST</DOCTYPE>
    <DOCUMENTID/>
    <PARTNRID>IM1</PARTNRID>
  </DOCUMNTREF>
</PARTNER>
<PARTNER>
  <NAME>Computer Service and Rentals</NAME>
  <PARTNRTYPE>SHIP_TO</PARTNRTYPE>
  <PARTNERID>Atlanta (OPS)</PARTNERID>
  <CURRENCY/>
  <DUNS_NUMBER/>
  <PARTNIDX/>
  <USERAREA>
    <ORACLE.SHOW_SHIPMENT_005.SHIPMENT.PARTNER.USERAREA>
      <ORACLE.INTMED_SHIP_TO_LOCATION/>
      <ORACLE.POOLED_SHIP_TO_LOCATION/>
    </ORACLE.SHOW_SHIPMENT_005.SHIPMENT.PARTNER.USERAREA>
  </USERAREA>
  <ADDRESS>
    <ADDRLINE index="1">1156 River Street</ADDRLINE>
    <ADDRLINE index="2"/>
    <ADDRLINE index="3"/>
    <ADDRLINE index="4"/>
    <ADDRTYPE/>
    <CITY>Atlanta</CITY>
    <COUNTRY>US</COUNTRY>
    <COUNTY>Fulton</COUNTY>
    <POSTALCODE>30301</POSTALCODE>
    <REGION/>
    <STATEPROVN>GA</STATEPROVN>
  </ADDRESS>

```

```

</ADDRESS>
    <CONTACT>
        <FAX>404-876-8526</FAX>
        <TELEPHONE>404-876-0987</TELEPHONE>
        <URL/>
    </CONTACT>
    <DOCUMNTREF>
        <DOCTYPE>SHIPMENTREQUEST</DOCTYPE>
        <DOCUMENTID/>
        <PARTNRID>Atlanta (OPS)</PARTNRID>
    </DOCUMNTREF>
</PARTNER>
<SHIPITEM>
    <QUANTITY qualifier="ITEM">
        <VALUE>3</VALUE>
        <NUMOFDEC/>
        <SIGN/>
        <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="BACKORDERED">
        <VALUE>0</VALUE>
        <NUMOFDEC/>
        <SIGN/>
        <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="ORDERED">
        <VALUE>3</VALUE>
        <NUMOFDEC/>
        <SIGN/>
        <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="SHIPPED">
        <VALUE>3</VALUE>
        <NUMOFDEC/>
        <SIGN/>
        <UOM>Ea</UOM>
    </QUANTITY>
    <QUANTITY qualifier="VOLUME">
        <VALUE>3</VALUE>
        <NUMOFDEC/>
        <SIGN/>
        <UOM>FT3</UOM>
    </QUANTITY>
    <QUANTITY qualifier="WEIGHT">
        <VALUE>3</VALUE>
        <NUMOFDEC/>
        <SIGN/>
        <UOM>Lbs</UOM>
    </QUANTITY>
    <ITEM>WSH-Standard</ITEM>
    <COUNTRYORG/>
    <FRGHTCLS/>
    <HAZRDMATL/>
    <ITEMDESC>WSH-Standard</ITEMDESC>
    <ITEMRV/>
    <ITEMX/>
    <NOTES/>
    <OWNRSHPCDE>Destination</OWNRSHPCDE>
    <PARTIALSHP>0</PARTIALSHP>
    <SHIPNOTES/>
    <USERAREA>

```



```

<ORACLE.SHOW_SHIPMENT_005.SHIPMENT.SHIPITEM.USERAREA>

<ORACLE.DELIVERY_DETAIL_NUMBER>225272</ORACLE.DELIVERY_DETAIL_NUMBER>
  <ORACLE.PARENT_DELIVERY_DETAIL_NUMBER/>

<ORACLE.SHIP_TOLERANCE_ABOVE>50</ORACLE.SHIP_TOLERANCE_ABOVE>

<ORACLE.SHIP_TOLERANCE_BELOW>50</ORACLE.SHIP_TOLERANCE_BELOW>
  <ORACLE.LOAD_SEQUENCE_NUMBER/>
  <ORACLE.SHIP_SET_NUMBER/>
  <ORACLE.SUBINVENTORY>Stores</ORACLE.SUBINVENTORY>
  <ORACLE.TOP_MODEL_LINE_NUMBER/>
  <ORACLE.SHIP_MODEL_COMPLETE_FLAG/>
  <ORACLE.COMMODITY_CODE_CATEGORY/>
  <ORACLE.SHIPMENT_PRIORITY_CODE/>

<ORACLE.DELIVERY_TO_LOCATION>1093</ORACLE.DELIVERY_TO_LOCATION>
  <ORACLE.CUSTOMER_NAME>Computer Service and
Rentals</ORACLE.CUSTOMER_NAME>

<ORACLE.CUSTOMER_NUMBER>1006</ORACLE.CUSTOMER_NUMBER>
  <ORACLE.DESCRPTIVE_FLEXFIELD>
    <ORACLE.ATTRIBUTE_CATEGORY/>
    <ORACLE.ATTRIBUTE1/>
    <ORACLE.ATTRIBUTE2/>
    <ORACLE.ATTRIBUTE3/>
    <ORACLE.ATTRIBUTE4/>
    <ORACLE.ATTRIBUTE5/>
    <ORACLE.ATTRIBUTE6/>
    <ORACLE.ATTRIBUTE7/>
    <ORACLE.ATTRIBUTE8/>
    <ORACLE.ATTRIBUTE9/>
    <ORACLE.ATTRIBUTE10/>
    <ORACLE.ATTRIBUTE11/>
    <ORACLE.ATTRIBUTE12/>
    <ORACLE.ATTRIBUTE13/>
    <ORACLE.ATTRIBUTE14/>
    <ORACLE.ATTRIBUTE15/>
  </ORACLE.DESCRPTIVE_FLEXFIELD>
  </ORACLE.SHOW_SHIPMENT_005.SHIPMENT.SHIPITEM.USERAREA>
</USERAREA>
<DOCUMNTREF>
  <DOCTYPE>PURCHASEORDER</DOCTYPE>
  <DOCUMENTID/>
  <PARTNRID>1093</PARTNRID>
</DOCUMNTREF>
<INVDETAIL>
  <QUANTITY qualifier="ITEM">
    <VALUE>3</VALUE>
    <NUMOFDEC/>
    <SIGN/>
    <UOM>Ea</UOM>
  </QUANTITY>
  <DESCRIPTN>WSH-Standard</DESCRIPTN>
  <LOTLEVEL index="1"/>
  <LOTLEVEL index="2"/>
  <SERIALNUM/>
  <USERAREA>

<ORACLE.SHOW_SHIPMENT_005.SHIPMENT.SHIPITEM.INVDETAIL.USERAREA>
  <ORACLE.TO_SERIAL_NUMBER/>

```

```
<ORACLE.PREFERRED_GRADE/>

</ORACLE.SHOW_SHIPMENT_005.SHIPMENT.SHIPITEM.INVDETAIL.USERAREA>
    </USERAREA>
    </INVDETAIL>
    </SHIPITEM>
    </SHIPMENT>
    </SHOW_SHIPMENT>
    </DATAAREA>
</SHOW_SHIPMENT_005>
```

Oracle Shipping Execution EDI Transactions

This chapter covers the following topics:

- DSNO

DSNO

Departure Ship Notice Outbound 856

DSNO is a notice used by the supplier to provide advance notification of a shipment to the buyer or trading partner that the material has been shipped. It is extremely flexible, and permits the seller to transmit shipment information in varying levels of detail. It is also referred to as ASC X12 856 transaction EDI.

Interface Trip Stop (ITS): DSNO is generated as a part of Interface Trip Stop process. DSNO can also be generated after Ship Confirm using the concurrent program Departure Ship Notice Outbound.

The Outbound Execution engine is responsible for extracting data related to a transaction from Oracle Applications. A transaction is referred to as a Business Document or a message i.e. purchase order, invoices, shipment notices, etc. The Extracted data can be staged in a staging table, fed directly to the EDI Flat File process to create an Outbound EDI File, or sent to the XML Engine for producing XML Documents.

A given transaction can have multiple representation for the outside world i.e. a purchase order document can be an XML document for one customer and EDI Flat File for another customer. The data related to different maps for the same transaction is stored in the EDI repository. The triggering process will identify which map to process and then pass on the value to the engine.

The input to the Execution Engine is the Map Identifier for the message and the transaction type. The map identifier stores the mapping information between the Oracle Applications tables and the Flat File or XML or Staging table.

The Execution engine extracts the data from EC messaging views, defined on top of

business views for each transaction. In addition to the data obtained from the view, the engine executes stored procedures, functions, and custom code to pull out the remaining business data. RFID information can also be provided as part of the DSNO for Oracle Warehouse Management enabled organizations. For more information, see the *Oracle E-Commerce Gateway User and Implementation Guide*.

Oracle Order Management Error and Information Messages

This appendix covers the following topics:

- Process_PO and Acknowledge_PO
- Show_SalesOrder
- Cancel_PO
- Change SO
- Inbound Change PO Request

Process_PO and Acknowledge_PO

1. The error out message if it is not determined whether document delivery is required for that TP.

OE_OI_TP_NOT_FOUND - Missing Primary SOLD_TO site usage for the customer id &CUST_ID

2. The error out message if the outbound acknowledgment data is not generated correctly.

OE_OI_ACK_DATA_NOT_FOUND - Error in getting data from &TABLE for sending the acknowledgment

3. The error out message if a particular type of EDI location is not found.

OE_OI_EDI_LOC_NOT_FOUND - &EDI_LOCATION not found for the site usage of &SITE_USAGE

4. The error out message if the org cannot be determined from the address.

OE_OI_ORG_NOT_FOUND - &SITE_USAGE site usage is not found for address id &ADDRESS_ID

5. The error out message if the customer site cannot be determined when generating acknowledgment data.

Process_PO and Acknowledge Purchase Order

OE_OI_CUST_SITE_NOT_FOUND - Could not determine account/party site id for customer id &CUST_ID

6. The error out message if the customer cannot be identified before raising the acknowledgment event.

OE_OI_CUST_NOT_FOUND - No Customer Found & OPT_TABLE for document id &DOC_ID

Show_SalesOrder

No seeded messaged for this transaction exist.

Cancel_PO

1. Error message if invalid quantity is sent on the inbound Cancel_PO XML.

OE_OI_INVALID_QTY_3A9 - Attempt to cancel Invalid Quantity

2. Error message if we cannot determine whether document delivery is required for that trading partner.

OE_OI_TP_NOT_FOUND - Missing Primary SOLD_TO site usage for the customer id &CUST_ID

3. Error message if we cannot generate the outbound acknowledgment data correctly.

OE_OI_ACK_DATA_NOT_FOUND - Error in getting data from &TABLE for sending the acknowledgment

4. Error message if a particular type of EDI location is not found.

OE_OI_EDI_LOC_NOT_FOUND - &EDI_LOCATION not found for the site usage of &SITE_USAGE

5. Error message if the org cannot be determined from the address.

OE_OI_ORG_NOT_FOUND - & SITE_USAGE site usage is not found for address id &ADDRESS_ID

6. Error message if the customer site cannot be determined when generating acknowledgment data.

OE_OI_CUST_SITE_NOT_FOUND - Could not determine account/party site id for customer id &CUST_ID

7. Error message if the customer cannot be identified before raising the acknowledgment event.

OE_OI_CUST_NOT_FOUND - No Customer Found &OPT_TABLE for document id &DOC_ID

Note: Integration with the Collaboration History Module (CLN)

Order Management provides integration with Collaboration History Module (CLN).

The Collaboration History module records the transaction history when the trading partners exchange messages. To enable this Order Management has incorporated the API calls during the processing of the XML messages.

Refer to the Collaboration History Module documentation for more details.

Change SO

Following message will be raised when status of 3A8 response is 'ACCEPT' but 3A8 does not match with the changes proposed on the original 3A7.

1. **Message_Name: ONT_3A8_RESPONSE_COL_MISMATCH**

3A8 Request Purchase Order Change response does not match with changes proposed on the original 3A7.

Message Type: Error

1. One or more rejected lines was received on a Change PO Response
Change PO - Buyer response contained one or more rejected lines which were not processed by Order Management
Change PO Response was rejected at header level.
Change PO - Buyer response contained a rejected header and was not processed by Order Management.
2. OM is ready to send Change SO
OM is ready to send Change SO message.
3. Setup validation done (Successful)
Change SO - Setup validation successful.
4. Setup validation done (Failed)
Change SO - Setup validation failed.
5. Change Sales Order message sent

Change SO message has been sent.

Inbound Change PO Request

1. When the Change PO message is processed and Order Import tables are populated with the data the message recorded using Collaboration History Module API is:

On success – Change PO - Data successfully entered into Open Interface tables

2. Order Import is going to be run immediately:

Change PO - Order Import will be run in Synchronous mode

3. Order Import will be run in Asynchronous mode.

Change PO - Order Import will be run in Asynchronous mode

4. Order Import has completed.

Change PO-Order import has completed (Sucessfully)

5. Order Import has completed.

Change PO-Order import has completed (Failure)