# Oracle® Retail Enterprise Inventory Cloud Service

## Administration Guide

ORACLE®

Oracle Retail Enterprise Inventory Cloud Service Administration Guide, Release 19.11

F89495-01

# Contents

# 2    Data Seeding

# 3    Reporting

# 4    Internationalization

# 5   Batches

# 6 Batch - Data Archive

# 7 Technical Maintenance Screens

# 8 Integration

# 9   Configuration

# 10   Operational Issues Screens

# A   Appendix: Run Importer Separately from SIOCS Job Admin

# B   Appendix: BDI Transmitter CLI Procedure - Load External Data File

# C   Appendix: BDI Security Roles and Groups

# D   Appendix: Report Formats

# E   Appendix: Batch File Layout Specifications

## F    Appendix: Auto-Authorized Third-Party Stock Count Process Overview

## G    Appendix: Unit and Amount Stock Counts Export

## H    Appendix: UPC Barcode

## I    Appendix: EICS Provided URLs

## J    Appendix: Data Seeding Process Flows

## K    Appendix: Pricing Import with On-premise RPM

## L    Appendix: Open Transaction Seeding via BDI

## M    Appendix: Third Party Pricing Integration

## N    Appendix: Third-Party RFID Integration

## O    Appendix: Item Ticket

# P    Appendix: SFTP File Directory and Filename Format

# List of Examples

# List of Figures

# List of Tables

**ORACLE**

# Send Us Your Comments

Oracle Retail Enterprise Inventory Cloud Service Administration Guide, Release 19.11

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

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

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

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

> **Note:**
>
> Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Help Center (OHC) website. It contains the most current Documentation Library plus all documents revised or released recently.

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

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

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

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

# Preface

This document describes the administration tasks for Oracle Retail Enterprise Inventory Cloud Service.

## Audience

This document is intended for administrators.

## Documentation Accessibility

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

**Access to Oracle Support**

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

## Related Documents

For more information, see the following documents in the Oracle Retail Store Inventory Operations Cloud Services Release 19.11 documentation set:

- *Oracle Retail Store Inventory Operations Cloud Services Release Readiness Guide*
- *Oracle Retail Store Inventory Operations Cloud Services Implementation Guide*
- *Oracle Retail Store Inventory Operations Cloud Services Data Model*
- *Oracle Retail Enterprise Inventory Cloud Service Security Guide*
- *Oracle Retail Enterprise Inventory Cloud Service User Guide*
- *Oracle Retail Store Operations Cloud Service User Guide*
- *Oracle Retail Store Operations Cloud Service Mobile Guide*

## Improved Process for Oracle Retail Documentation Corrections

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

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

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

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number F123456-02 is an updated version of a document with part number F123456-01.

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

# Oracle Retail Documentation at the Oracle Help Center

Oracle Retail product documentation is available on the following website:

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

(Data Model documents are not available through Oracle Help Center. You can obtain them through My Oracle Support.)

# Conventions

The following text conventions are used in this document:

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

# 1
# Technical Architecture

This chapter describes the overall software architecture, offering a high-level discussion of the general structure of the system.

Overall there is no change in deployment from the previous release period.

There are underlying version updates to the technical stack (DB 19c, Web Logic 12.2.1.4, updated versions of UI libraries, Fusion middle ware libraries and so on.)

## Multiple Products

EICS (Enterprise Inventory Cloud Service) and SOCS (Store Operations Cloud Service) are two separately licensed products.

EICS includes:

- EICS Browser Client
- EICS Web Services
- EICS Server Tier
- EICS Database tier with data access code, batches, reports

SOCS includes:

- Oracle MAF Client

In order to use SOCS, EICS needs to be deployed.

# Logical Model

**Figure 1-1    Logical Model**
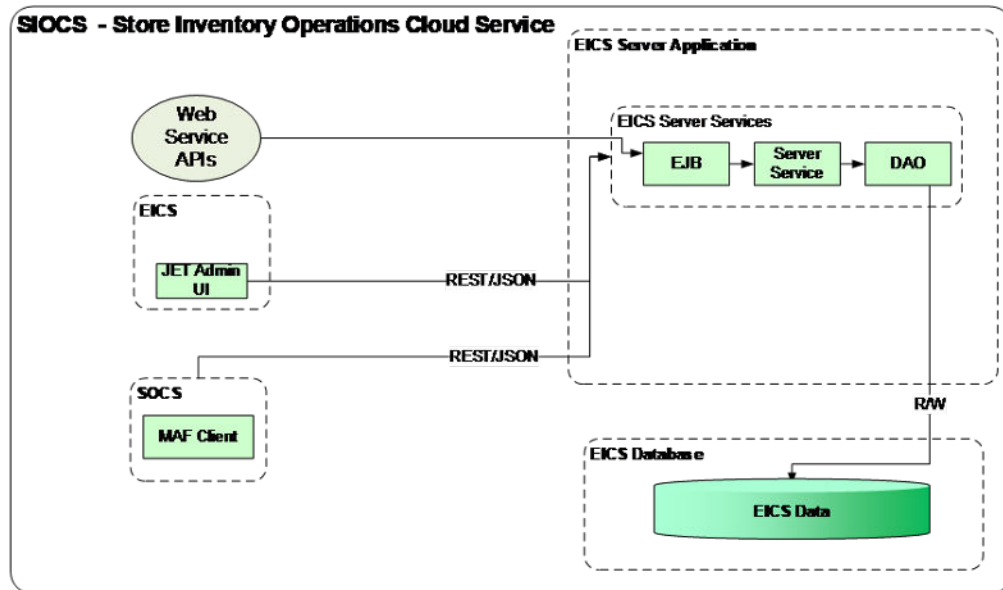
# SaaS Deployment

**Figure 1-2　SaaS Deployment**
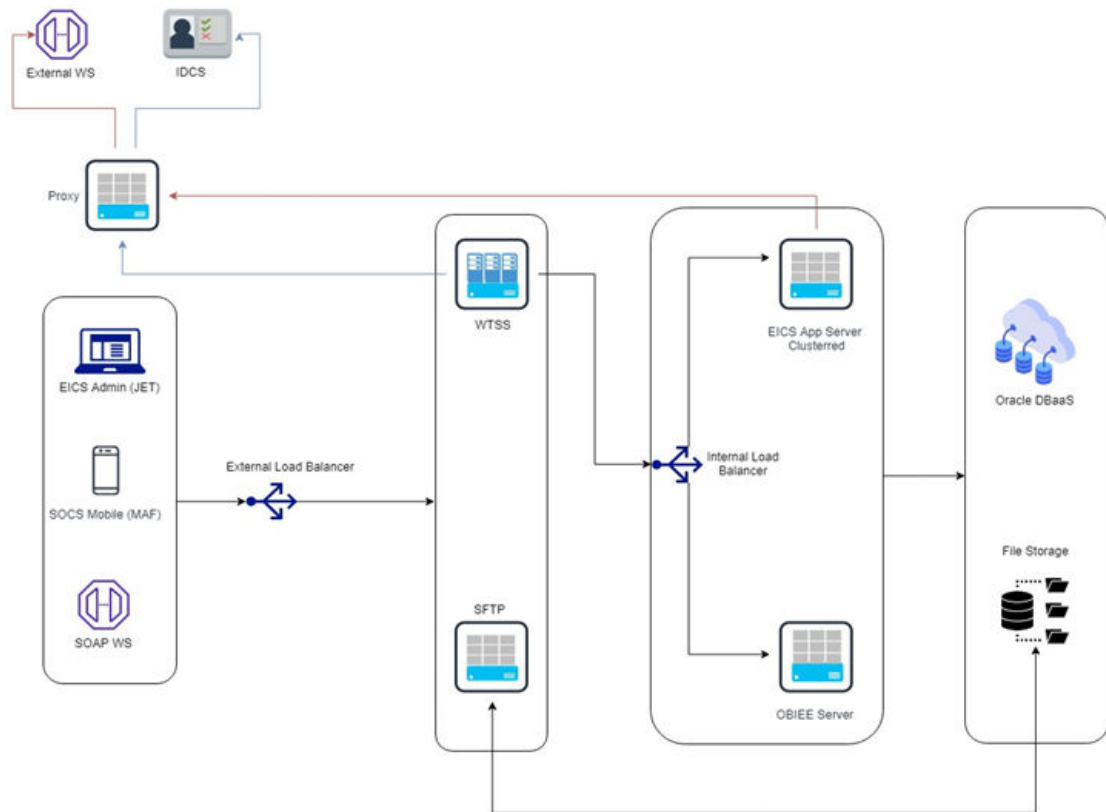


# Cloud Deployment

## EICS Client

Oracle JET based browser application that allows the user to perform a wide range of administrative functions.

## SOCS Mobile Client

The mobile client provides all day-to-day transactional workflows within an Oracle Mobile Application Framework (MAF) platform. MAF is a hybrid-mobile platform that supports both iOS and Android devices. For more details, please see *Oracle Retail Store Operations Cloud Service Mobile Guide*.

# Web Services

There is no GUI for the SOAP web services APIs that are provided by EICS. These APIs allow customers to create or develop applications or add-ons that can replicate some or all of the steps of a transaction workflow.

# WTSS / IDCS or OCI IAM

WTSS: Web Traffic Security Service

Integration Cloud Services uses Oracle Identity Cloud Service (IDCS) as its identity provider (IDP) or Oracle Cloud Infrastructure Identity and Access Management (OCI IAM) as its identify provider (IDP).

# EICS Application Server(s)

Server deployed as a J2EE application inside the WebLogic Application Server.

# OBIEE Server

OBIEE / BIPublisher is used as reporting engine.

# SFTP Server

The Secure File Transfer Protocol server is responsible for the moving of files from outside the internet firewall to within the internet firewall in a secure manner. This is used with File Storage Service.

# Oracle DB Server (DBaaS)

Contains EICS schema. Uses JDBC to access data from the database.

WebLogic application server provides a connection pool to use database resources in an efficient fashion.

PL/SQL stored procedures are also used for high volume batch processing.

# Client-Server Communication

Client(s) use ReST service calls to access the server.

External systems may use SOAP service calls to access the server.

All transactions are container managed.

Performance is sensitive to network latency (hence compression from client to server).

# Integration

Oracle Retail Integration Cloud Server (RICS) is used for integration between multiple systems within the cloud as well as deployment of public web services by EICS and other systems.

## Oracle Retail Integration Bus (RIB)

The RIB is a near-real time, message-based communication queue. Payloads are delivered in an asynchronous fashion between multiple systems on the enterprise in a non-blocking (fire and forget) manner. This broadcast of notifications is subscribed to by each application interested in an event notification.

## SOAP Web Services

Numerous APIs into EICS Server are exposed through operations in a wide array of SOAP web services available to external applications.

## File-Based

EICS receives information through several file-based scenarios in order to receive large volumes of data for more efficient processing. These include sales records, stock counts performed by a third party, and price change from a price management system.

# Deployment with Other Products

EICS is deployed in three ways:

- It is deployed with the rest of Retail assets such as Retail Merchandising Foundation Cloud Service and Retail Integration Cloud Service.

- It is deployed with POS/Xstore typically with or without RIB.

- It is deployed as a stand-alone or point deployment.

EICS and SOCS have a distributed deployment model with browser and mobile devices running at stores, connecting with server and database hosted at corporate. The central server deployment allows real-time inventory queries for stock-on-hand positions across the enterprise but requires a fairly robust network connection between store and corporate environments.

# Deployment - Performance: Bandwidth, Scaling

## Bandwidth Requirements for Browser Clients

Installations with less than 128 KB bandwidth available between the device containing the browser or the mobile application and the data center are not recommended or supported. Limiting the client to less than 128 KB total available bandwidth causes unpredictable network utilization spikes, and the performance of the client degrades below requirements established for the product.

## Network Latency Constraints

EICS is also sensitive to the network latency between the browser or mobile device and the data center. Oracle Retail does not recommend or support installations with more than 100 ms total round-trip network latency between the client device and the data center. Latency beyond the 100 ms limit causes unpredictable network utilization spikes, and the

performance of the client degrades below requirements established for the product. The 100 ms limitation provides reasonable, predictable performance and network utilization for transactions.

## Application Server Clustering

EICS supports application server clustering with a load balancer for Oracle WebLogic.

The EICS Server can run inside an application server cluster, but the load must be distributed among the nodes in the cluster.

It is also possible to configure a single application server instance with multiple JVMs inside the application server instance. When doing this, an upper limit of 1.5 to 2 JVMs per server CPU is recommended.

Clustering for Oracle WebLogic Server is also supported through a load balancer.

## Database Clustering with RAC

EICS supports a RAC-enabled database for performance.

# 2

# Data Seeding

This chapter contains the following:

- What is Data Seeding?
- Overview of Data Seeding Process
- Pre-requisites
- Configure EICS for Data Seeding
- Run BDI Data Seeding

## What is Data Seeding?

EICS needs merchandising foundation data (stores, items, initial inventory positions, suppliers, and so on) in order to function.

Initial inventory data seeding is applicable for new or fresh full EICS installation and should only be run once. Once the initial set of data is seeded into EICS, subsequent inventory changes are communicated via Oracle Retail Integration Cloud Service (RICS - RIB).

EICS data seeding uses Oracle Retail Integration Cloud Service (RICS) BDI process to seed data from MFCS into the EICS. There are 35 processes that get extracted from MFCS into EICS. These processes have been grouped into nine (9) groups based on dependency and functionality. See Data Seeding Process Flows for details.

Data seeding can be used for a variety of use cases:

- Loading just the Foundation data from Merchandising
- Loading all stores data
- Loading a single store data

## Overview of Data Seeding Process

1. An **Extractor** extracts data from a data producing system into an outgoing intermediate integration interface table.
2. A **Downloader** and **transporter** services data to receiver service.
3. An **Uploader** uploads the data to an incoming intermediate integration interface table.
4. An **Importer** imports the data from the interface tables into EICS application tables.

For additional detail on BDI Data Integration Topologies, please see *Oracle Retail Bulk Data Integration Implementation Guide*.

## Pre-requisites

Prior to running data seeding, the following requirements must be met:

**Sourcing System** must be installed:

The sourcing system is the merchandising foundation data sourcing system, such as MFCS.

**BDI Transport System** must be installed:

BDI Process Flow: Mechanism to run multiple dependent batch jobs.

- BDI-RMS Batch Job Admin: Provides management of BDI for RMS batch jobs.

- BDI-SIM Batch Job Admin: Provides management of BDI for EICS batch jobs.

**Target System** must be installed:

EICS is the receiver target system, proper security settings and system configuration for data seeding must be set. See Configure EICS for Data Seeding section for details.

# Configure EICS for Data Seeding

- Security Settings
- Configuration Parameters
- Steps to Enable Data Seeding in SIOCS UI

## Security Settings

There are two types of security assignments. IDCS Integration Users Role and Application Permissions. Integration User Role is managed within IDCS or OCI IAM. EICS Application Permissions are managed via SIOCS. See *Oracle Retail Enterprise Inventory Cloud Service Security Guide* "User Provisioning" section for details.

The following table shows user security relevant to Data Seeding:

| Task | IDCS or OCI IAM Application Role | Application Permission Name | Description |
|---|---|---|---|
| BDI Process Flow Integration to access EICS Importer Service | integration_users | N/A | The Application integration user is meant for non-EICS application accessing EICS provided integration resources. The integration app user only needs to be assigned to enterprise security group.<br><br>Notes:<br><br>The application integration user does not require access to the SIOCS client applications, and therefore does not require store assignments or role assignments (permissions). |
| View EICS Import Job Activities | sysop_users | Access Job Admin | Allows access to SIOCS batch job screen. |

| Task | IDCS or OCI IAM Application Role | Application Permission Name | Description |
|---|---|---|---|
| Manual Run Data Seed Importer | sysop_users | Access Data Seed | Allows SIOCS user to manually run data seeding importer jobs via SIOCS batch job admin. |

> **Note:**
>
> sysop_users IDCS or OCI IAM Application Role should only be assigned to Oracle Cloud Administrator users.

## Configuration Parameters

The EICS Data Seeding Importer process in EICS imports data from the staged or intermediate tables into EICS application tables. The configuration parameters determine how the importer process operates in EICS.

The data seeding system configuration flags are disabled upon EICS installation because the full set of data should only be loaded once for foundation inventory data and store inventory data.

The following table shows the configuration parameters and their default values.

| Configuration Name | Description |
|---|---|
| Data Seed Foundation Data | The foundation data can be used for initial bulk load of EICS foundation data (such as Supplier, Merchandise Hierarchy, Item and others) from data producing system during initial setup of EICS system. This option should be disabled after EICS data seeding is completed. Default value is set to **No**. |
| | If you set this parameter to **Yes**, then data from sourcing system is automatically loaded into EICS application tables during BDI Process Flow process. If you leave this parameter to No, the process loads the data from sourcing system into EICS BDI inbound interface tables, then you must run Importer Job to load data from interface tables into application tables manually via SIOCS Batch Job Admin Console, see Appendix: Run Importer Separately from SIOCS Job Admin for details. |
| Data Seed Store Data | The seed store data can be used for initial bulk data load of EICS store inventory data (such as Store Item, and Store Item Stock) from a data producing system. Default value is set to **No**. |
| | If you set this parameter to **Yes**, then store data from sourcing system are automatically loaded into EICS application tables during BDI Process Flow process. If you leave this parameter to No, the process loads the store data from sourcing system into EICS BDI inbound interface tables, then you must run Importer Job to import store data from interface tables into application tables manually via SIOCS Batch Job Admin Console, see Appendix: Run Importer Separately from SIOCS Job Admin for details. |

| Configuration Name | Description |
|---|---|
| Data Seed By Store | Determines whether the store data are automatically loaded for a single store only when **Data Seed Store Data** is set to Yes. By default, this parameter is set to No, then all stores and inventory are seeded. To rollout a single store, enable this configuration allows only a single store's inventory data to be seeded. Default value is set to **No**. |
| | If you set this parameter to **Yes**, then you must run Importer Job for store data manually via SIOCS Batch Job Admin Console, see Appendix: Run Importer Separately from SIOCS Job Admin for details. |
| Data Seed Chunk Limit | Controls the data seeding transaction size limit. Default value is set to 10000. |
| Data Seed On Error Continue | This flag determines how the error should be handled during data seeding import process. If the flag is turned on (true), the process will skip the current chunk of data, and continue to process next chunk of data. If the flag is turned off (false), the data loading process will be terminated when an error occurs. Default value is set to **No**. |

To change configuration values:

1. Login to Oracle Enterprise Inventory Cloud Service page as EICS System Ops user.

> **Note:**
>
> Full details about the configuration screens can be found in the *Oracle Retail Enterprise Inventory Cloud Service User Guide*.

2. On Main Menu, select Admin - Configuration - System Administration.

3. Search for data seed in option field, change value and save the change.

**Figure 2-1    Data Seed Settings**
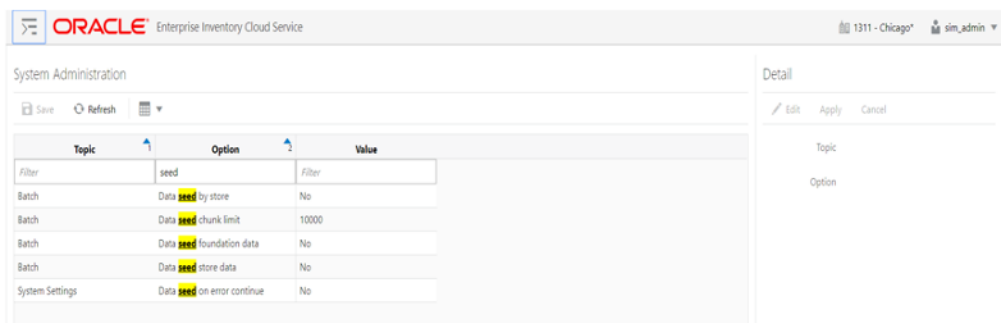


## Steps to Enable Data Seeding in SIOCS UI

To seed initial inventory foundation data from sourcing system directly into destination application tables, an application implementation consultant must perform the following configuration steps:

1. Use SIOCS System Administration Console to set **data seeding foundation** to **Yes**.

2. Use SIOCS System Administration Console to set **data seeding store** to **Yes**.

## Steps to Manual Control Importers

Optionally, if you want to manually control importer steps, perform the following steps:

1. Use SIOCS System Administration Console to set **data seeding foundation** to **No**.

2. Use SIOCS System Administration Console to set **data seeding store** to No.

3. Start BDI Process Flow (see BDI Process Flow Admin Console section for details).

4. Once data is loaded into inbound interface table, if you want to import interface data into application tables, set **data seeding foundation** to **Yes**, set **data seeding store** to **Yes**.

5. Use SIOCS Batch Job Admin Data Seeding jobs to manually start data seeding importer job.

## Optional Steps Prior Initial Seeding

For initial data seeding into EICS, if slowness is observed due to validating data integrity constraints, sysop user can disable the data seeding constraints prior to running initial data seeding (Note: only certain predefined database constraints can be disabled via Data Seeding Console). The database constraints must be enabled after initial data seeding is completed.

To disable data seeding constraints:

1. Login Oracle Enterprise Inventory Cloud Service page as EICS System Ops user.

2. On Main Menu, select **Admin/ Technical Maintenance/Job Admin**.

3. On Job Admin Screen, click **Data Seed**, this will bring you to the Data Seed screen.

4. Select Job Name **Database Constraints**.

5. From Seed Option drop down list, select **Disable Constraint**.

6. Click **Start Job**.

Notes: See Post Data Seeding steps for enable data seeding constraints

# Run BDI Data Seeding

You only need to run one- time full data seeding to initially seed functional inventory data into EICS.

The data seeding processes can be started via BDI Process Flow Admin Console.

## BDI Process Flow Admin Console

To start the process flow:

1. Login to BDI Process Flow Console.

2. Click **Manage Process Flow**.

3. Choose group **SIM** from **Select Group** drop-down list.

> **Note:**
>
> For data seeding process flow list, see Appendix: Data Seeding Process Flows that shows a list of Data Seeding Process Flows from Merchandising to EICS with groups. There are no dependencies within the same data seeding process group.

**Figure 2-2    BDI Process Flow Console**
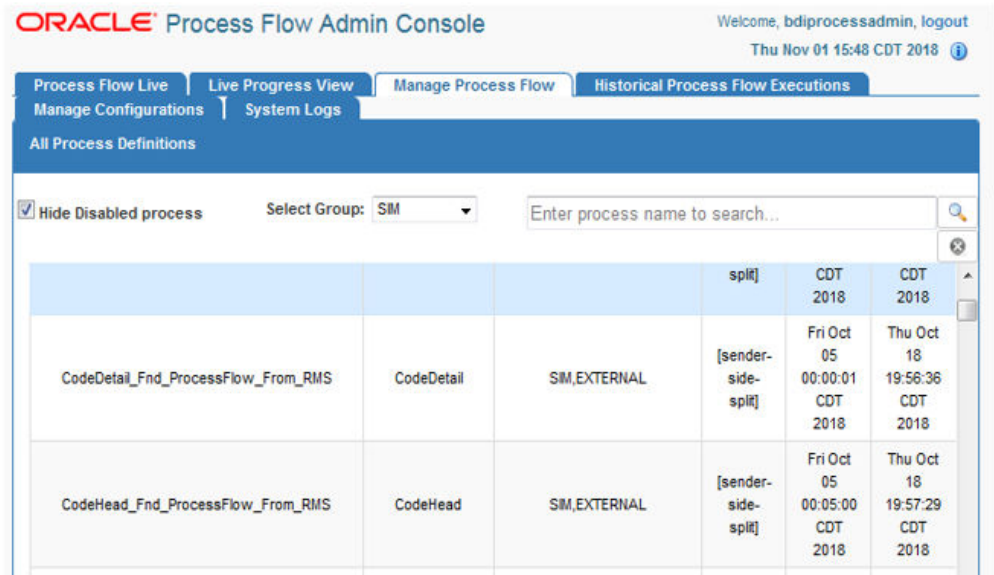


To start the process:

1.  Select the process name.
2.  Navigate to the lower panel and click L**aunch Process Flow**.
3.  Click **Run**.
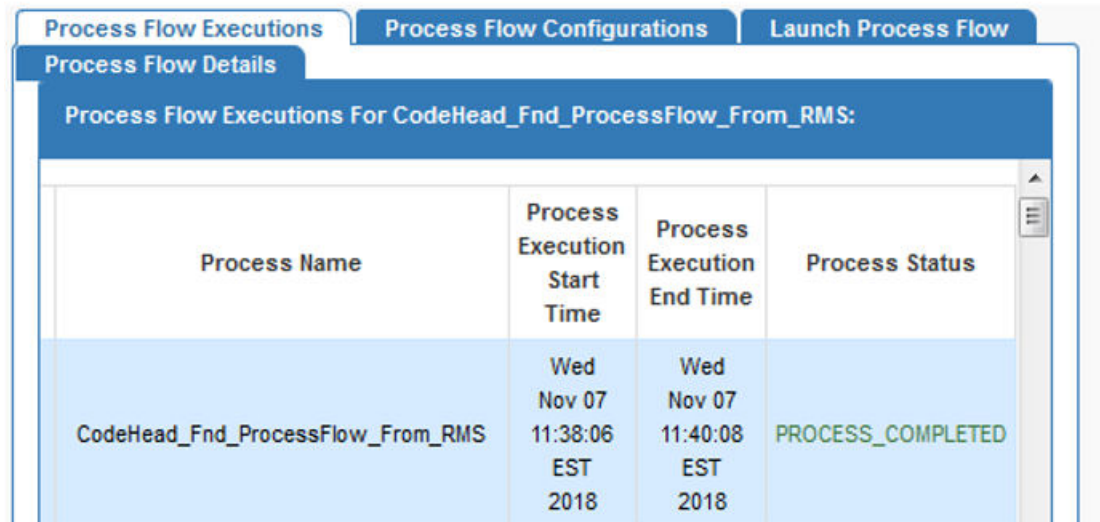
**Figure 2-3    Launch Process**

Chapter 2
Run BDI Data Seeding
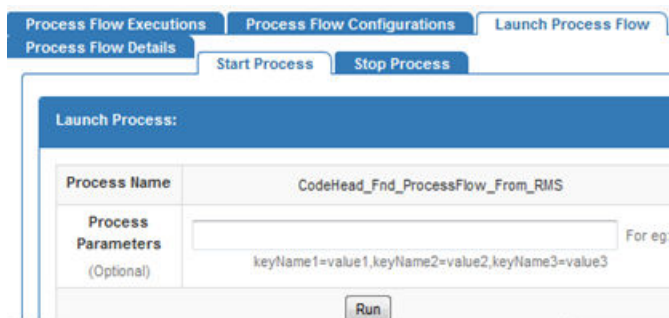
To view BDI Process Flow details, navigate to **Process Flow Execution** tab to track the status of process flow.

**Figure 2-4    Process Flow Details**



View Process Flow component details:

- Extractor
- Downloader
- Uploader/Receiver
- Importer

# Extractor

To view extract data from source app, launch edge app batch job admin. The following shows the steps to view extractor from RMS:

1. Launch RMS Batch Job Admin console.

2. Login as BDI Job Admin user.

ORACLE®

2-7

**Figure 2-5    RMS Batch Job**
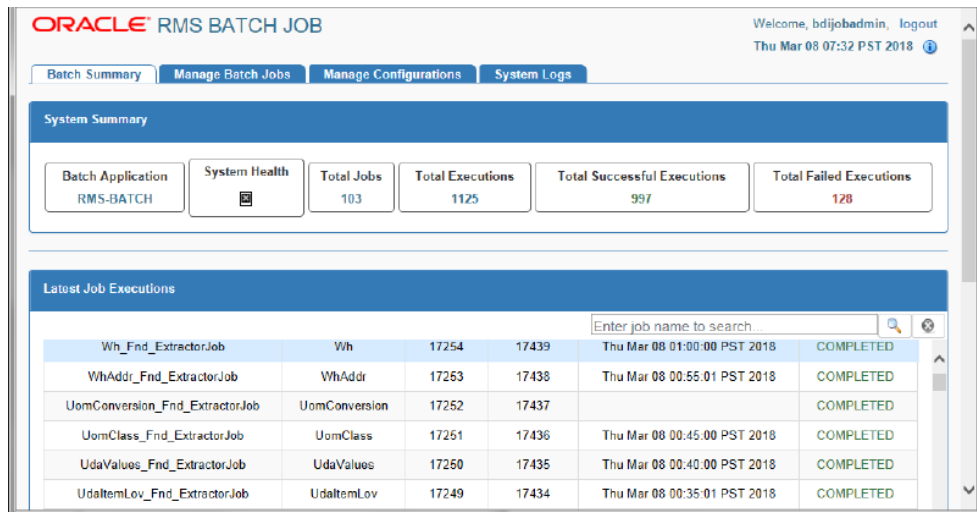


For details, please see *Oracle Retail Bulk Data Integration Implementation Guide*.

# Downloader

To view BDI Downloader job details:

1.  Launch BDI for edge app batch admin console.

    Follow these steps to view download from RMS.

2.  Launch BDI RMS Batch Job Admin console.

3.  Login as BDI RMS Job Admin user.

**Figure 2-6    BDI RMS Batch Job**
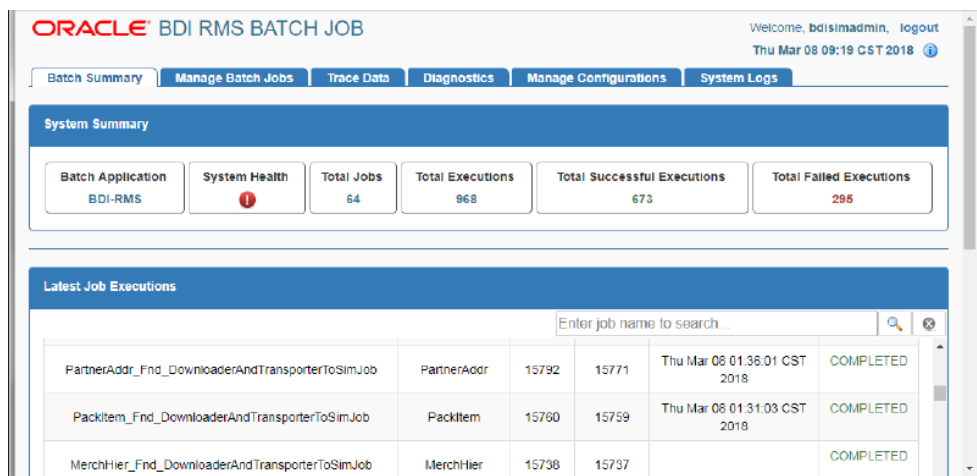


For details, please see *Oracle Retail Bulk Data Integration Implementation Guide*.

# Uploader/Receiver

To view BDI SIM uploader details:

1. Launch BDI SIM Edge Batch Job Admin console.

2. Login as SIM BDI Job Admin user.

**Figure 2-7    BDI SIM Batch Job**



For details, please see *Oracle Retail Bulk Data Integration Implementation Guide*.

# Importer

To View EICS Importer details:

1. Launch SIOCS application, login SIOCS as System Ops user.

2. On Main Menu, select Admin - Technical Maintenance - Job Admin.

**Figure 2-8    SIOCS Job Admin**
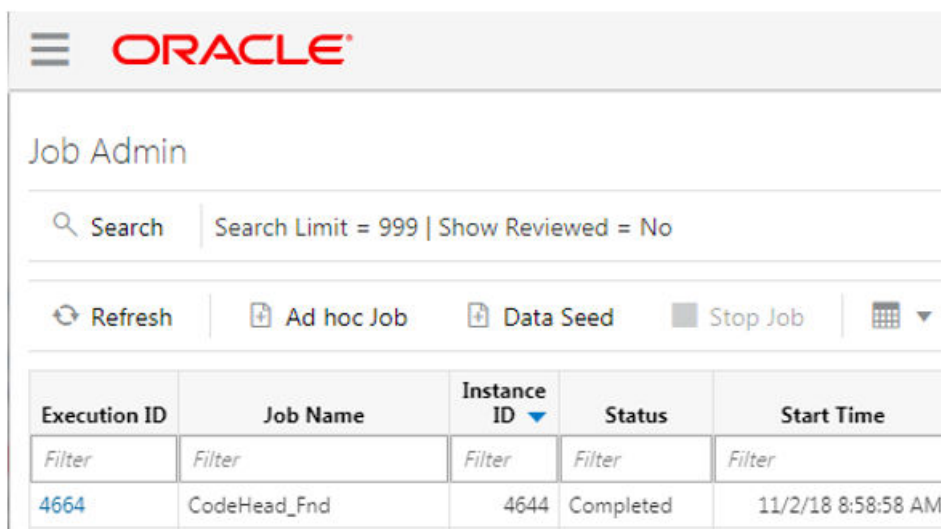


3. To view detail record, select the record, then click hyperlink on the execution id. This will bring you to the **Job Execution Detail** screen.

**Figure 2-9    Job Execution**
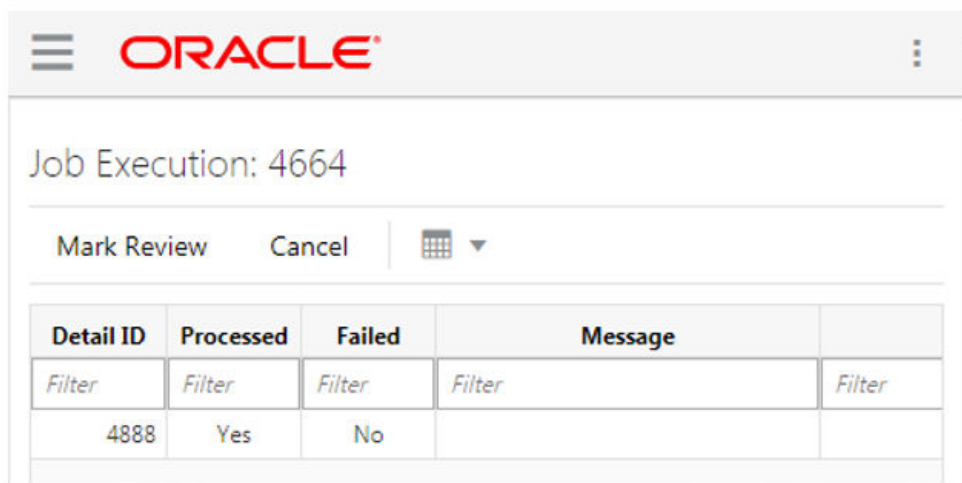


## Import Staging Data to Application Tables

If you set the EICS seed options to enabled, the data from external source system is automatically imported into the application table, and you can skip the following steps.

If you set the seed options to disabled, and want to import data from staging table to applications through manual steps, please refer to the following:

| Task | Step |
| --- | --- |
| Enable Data Seed Flags | Login SIOCS as System Ops user. |
| | On Main Menu, select Admin - Configuration - System Administration. |
| | Search for data seed on option field, change to desired value and save the change. |
| | Enabled Data Seed Foundation Data: true |
| | Enable Data Seed Store Data: true |
| Import data from stage tables to application tables | See Appendix: Run Importer Separately from SIOCS Job Admin. |

## Re-running EICS Importer Data Loads

After correcting the errors during process flow, you may re-run the initial data seeding process flow.

Important: Before executing additional runs of data seeding, please choose the appropriate re-seed options in SIOCS Job Admin Console.

> **✎ Note:**
>
> To import data for a specific store, please see Addition Notes: Importing Data for Single Store section

EICS system ops user login SIOCS, from Main Menu, navigate to Admin Technical Maintenance - Job Admin

On Job Admin Screen, click **Data Seed**. This will bring you to the Data Seed screen (for screen details, see Data Seed Importer Job), perform the following steps:

| Task | Step |
| --- | --- |
| Delete the previous seeded module data | Choose desired module from job drop-down list, and then select **Data Seed Option Deleted.** |
| | For example: |
| | Choose **Job Name Store** End. |
| | Select **Seed Option Delete Seeded**. |
| | If you enter the store id, then only the seed data for that store is deleted. |
| Re-import data from stage tables | Choose **Job Name Store** from job drop-down list, and then select **Data Seed Option Seed**. |
| (see Re-import data from Source System, Re-import data step) | For example: |
| | If you enter the store id, then only the seed data for that store is seeded. |
| | If you want to import all stores, leave the store id as blank. |
| Re-import data from Source System | Delete the previous seeded module data |
| | Go to BDI process flow, choose the flow to be re-run, start the BDI process flow. |

# BDI-Based Data Seeding Post-Load Processes

1. Enable Oracle Enterprise Inventory Cloud Service (EICS) constraints after data seeding.

   a. Login SIOCS as EICS System Ops user.

   b. On Main Menu, navigate to Admin - Technical Maintenance - Job Admin

   c. On Job Admin Screen, click **Data Seed**. This will bring you to the Data Seed screen.

   d. Select **Job Name** Database Constraints.

   e. From **Seed Option** drop-down list, select **Enable Constraint**, click **Start Job**.

2. Purge data from inbound data seeding stage tables.

   The inbound data seeding stage tables can be purged by batch job, the purge job purge group of data seeding interface tables, see Data Purging Batches - BDI Staging Data in the Batches chapter.

   Alternatively, if you want to purge individual interface table, then follow following steps:

   a. Login SIOCS as EICS System Ops user.

   b. On Main Menu, navigate to Admin Technical Maintenance - Job Admin.

   c. On Job Admin Screen, click **Data Seed**. This will bring you to the Data Seed screen.

   d. Select Each Importer Job Name (for example Code Head), then choose **Seed Option Delete In**, click **Start Job**.

3. Delete Starter Store

   The starter store (store id = 0) is for System Ops User to login SIOCS application prior seeding the stores into the system. After stores are seeded into SIOCS, the starter store needs to be removed, the operation only deletes starter store 0 if there are other non-starter stores in SIOCS database.

   To delete starter store, perform the following steps:

   a. Login SIOCS as EICS System Ops user.

   b. On Main Menu, navigate to Admin Technical Maintenance - Job Admin.

   c. On Job Admin Screen, click **Data Seed**. This will bring you to the Data Seed screen.

   d. Select Each Importer Job Name (Store), then choose **Delete Starter Store**, click **Start Job**.

4. Turn off EICS Data Seeding System Configuration Flags.

   a. Login SIOCS as EICS System Ops user.

   b. On Main Menu, navigate to Admin - Configuration - System Administration.

   c. Search for data seed on option field, change value and save the change.

   d. Set the following values:

      **Enabled Seed Foundation**: No

**Enable Seed Store Data**: No

# 3
# Reporting

EICS can produce reports for retails to view.

Reports are generated from within the functional areas of EICS and includes information about shipping documentation, delivery reports, pick detail reports and so on. EICS uses a report screen to preview the report by sending the request parameters as report name and required parameter for the corresponding listed reports.

This section covers the following:

- EICS Reporting Framework
- Report URL Location
- Previewing a Report
- EICS Operational Reports

## EICS Reporting Framework

The following section describes the reporting framework.

**Figure 3-1    Reporting Framework**
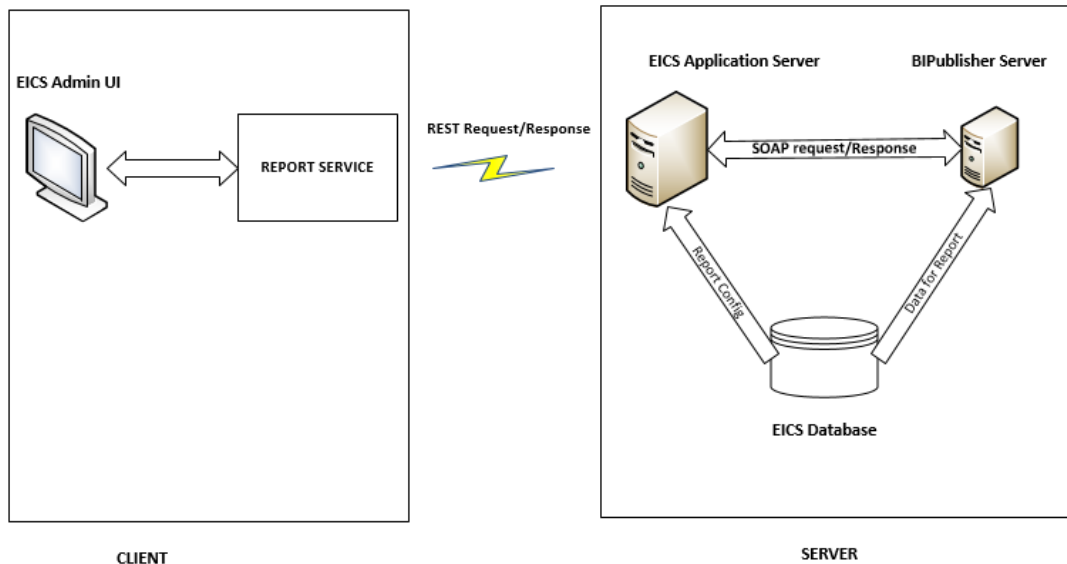


## Report Service

The report format type is predefined as pdf format in the client. The report service use REST service calls to access the EICS server.

## Security Considerations

Customers should create an IDCS or OCI IAM user and the user should be assigned the following BI IDCS or OCI IAM groups to access the report endpoints.

IDCS or OCI IAM groups required for production environment:

- BIConsumer
- BIContentAuthor
- BIAdministrator

IDCS or OCI IAM groups required for non-production environment:

- BIConsumer_preprod
- BIContentAuthor_ preprod
- BIAdministrator _ preprod

## BI Publisher Server

EICS uses BI Publisher as the printing engine. BI Publisher gets the report data from the database by sql and generates the report in pdf format. The following is the workflow for the process of Preview Reports:

1. A retailer makes a request on the JET Client to preview a report.
2. Format type of reports (pdf) which is added to the report detail parameters requested by retailers.
3. The client calls the reporting service to request the REST call.
4. The REST request hit the EICS server and fetch the report configuration from the EICS database to request the BI Publisher.
5. The BI Publisher server receives URLs or parameters, runs a query to retrieve data to be printed, and formats the data.
6. That data is then sent back to the UI as response and reports are download in the pdf format.

> **Note:**
>
> EICS tables (REPORT_TEMPLATE) refer to the report format. The EICS server fetches the report configuration detail from this table.

## Report URL Location

The URL Location for each report type:

**Table 3-1    Report URL Location**

| Type | URL Location |
|---|---|
| Customer Order Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrder Report/CustomerOrderReport.xdo |
| Customer Order Delivery BOL Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrderDeliveryBOLReport/CustomerOrderDeliveryBOLReport.xdo |
| Customer Order Bin Label Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrderB inLabelReport/CustomerOrderBinLabelReport.xdo |
| Customer Order Delivery Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrder DeliveryReport/CustomerOrderDeliveryReport.xdo |
| Customer Order Pick Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrderP ickReport/CustomerOrderPickReport.xdo |
| Customer Order Pick Discrepancy Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrderP ickDiscrepancyReport/CustomerOrderPickDiscrepancyReport.xdo |
| Customer Order Reverse Pick Report | <BIP_XMLPSERVER_URL>/SIOCS/CustomerOrderReversePickReport/CustomerOrderReversePickReport.xdo |
| Direct Delivery Report | <BIP_XMLPSERVER_URL>/SIOCS/DirectDeliveryR eport/DirectDeliveryReport.xdo |
| Direct Delivery AGSN Report | <BIP_XMLPSERVER_URL>/SIOCS/VendorDeliveryAGSNReport/VendorDeliveryAGSNReport.xdo |
| Direct Delivery Discrepant Item Report | <BIP_XMLPSERVER_URL>/SIOCS/DirectDeliveryDi screpantItemsReport/DirectDeliveryDiscrepantItemsRe port.xdo |
| Direct Delivery Label Report | <BIP_XMLPSERVER_URL>/SIOCS/VendorDeliveryLabel/VendorDeliveryLabel.xdo |
| Inventory Adjustment Report | <BIP_XMLPSERVER_URL>/SIOCS/InventoryAdjust mentReport/InventoryAdjustmentReport.xdo |
| InventoryAdjustmentAGSNRep ort | <BIP_XMLPSERVER_URL>/SIOCS/InventoryAdjustmentAGSNReport/InventoryAdjustmentAGSNReport.xdo |
| Item Basket Report | <BIP_XMLPSERVER_URL>/SIOCS/ItemBasketReport/ItemBasketReport.xdo |
| Item Detail Report | <BIP_XMLPSERVER_URL>/SIOCS/ItemDetailRepor t/ItemDetailReport.xdo |
| Purchase Order Report | <BIP_XMLPSERVER_URL>/SIOCS/PurchaseOrderR eport/PurchaseOrderReport.xdo |
| RFID History Report | <BIP_XMLPSERVER_URL>/SIOCS/RFIDHistoryReport/RFIDHistoryReport.xdo |
| RTV Report | <BIP_XMLPSERVER_URL>/SIOCS/RTVReport/RTV Report.xdo |
| RTV Shipment Report | <BIP_XMLPSERVER_URL>/SIOCS/VendorShipmentReport/VendorShipmentReport.xdo |
| RTV Shipment BOL Report | <BIP_XMLPSERVER_URL>/SIOCS/VendorShipmentBOLReport/VendorShipmentBOLReport.xdo |
| RTV Shipment Container Report | <BIP_XMLPSERVER_URL>/SIOCS/VendorShipmentCartonReport/VendorShipmentCartonReport.xdo |

**Table 3-1    (Cont.) Report URL Location**

| Type | URL Location |
| --- | --- |
| RTV Shipping Label Report | <BIP_XMLPSERVER_URL>/SIOCS/VendorShippingLabel/VendorShippingLabel.xdo |
| Scan List Report | <BIP_XMLPSERVER_URL>/SIOCS/ReplenishmentGapReport/ReplenishmentGapReport.xdo |
| Shelf Adjustment Report | <BIP_XMLPSERVER_URL>/SIOCS/ShelfAdjustment Report/ShelfAdjustmentReport.xdo |
| Shelf Replenishment Report | <BIP_XMLPSERVER_URL>/SIOCS/ShelfReplenishm entReport/ShelfReplenishmentReport.xdo |
| Stock Count All Location Report | <BIP_XMLPSERVER_URL>/SIOCS/StockCountAllLo cReport/StockCountAllLocReport.xdo |
| Stock Count Report | <BIP_XMLPSERVER_URL>/SIOCS/StockCountReport/StockCountReport.xdo |
| Stock Count Export Report | <BIP_XMLPSERVER_URL>/SIOCS/StockCountExpo rtReport/StockCountExportReport.xdo |
| Stock Count Rejected Item Report | <BIP_XMLPSERVER_URL>/SIOCS/StockCountRejec tedItemReport/StockCountRejectedItemReport.xdo |
| Store Order Report | <BIP_XMLPSERVER_URL>/SIOCS/StoreOrderReport/StoreOrderReport.xdo |
| Transfer Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferReport/TransferReport.xdo |
| Transfer Receiving Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferDelivery Report/TransferDeliveryReport.xd |
| Transfer Receiving AGSN Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferDeliveryAGSNReport/TransferDeliveryAGSNReport.xdo |
| Transfer Receiving Exception Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferDelivery ExceptionReport/TransferDeliveryExceptionReport.xdo |
| Transfer Receiving Label Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferDelivery Label/TransferDeliveryLabel.xdo |
| Transfer Shipment Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferShipmen tReport/TransferShipmentReport.xdo |
| Transfer Shipment BOL Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferShipmen tBolReport/TransferShipmentBolReport.xdo |
| Transfer Shipment Container Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferShipmen tCartonReport/TransferShipmentCartonReport.xdo |
| Transfer Shipping Label Report | <BIP_XMLPSERVER_URL>/SIOCS/TransferShippin gLabel/TransferShippingLabel.xdo |

> **Note:**
>
> <BIP_XMLPSERVER_URL> is the folder where EICS reports have been uploaded on the BI Publisher server.

# Previewing a Report

Users can download the report from EICS JET UI Home/Operations/Report.

**Figure 3-2    Reports Screen**
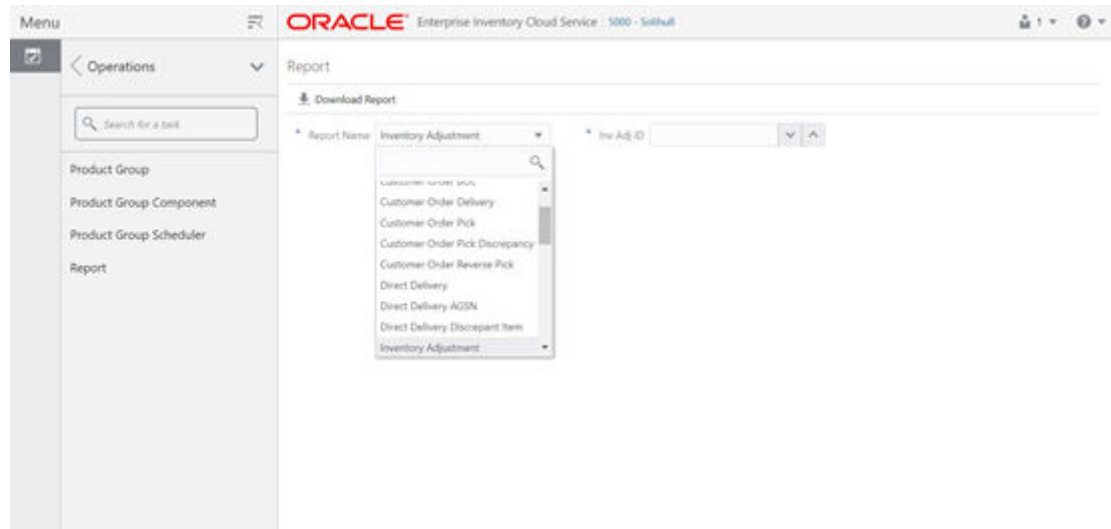


# EICS Operational Reports

The following list shows the EICS Operational Reports.

**Table 3-2    Operational Reports**

| Report Name | Report Parameters | Primary Views or Tables |
| --- | --- | --- |
| Customer Order Bin Label Report | PICK_ID, COPIES | RPRT_FUL_ORD_BIN_V |
| Customer Order BOL Report | DELIVERY_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_FUL_ORD_DLV_BOL_V |
| Customer Order Delivery Report | DELIVERY_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_FUL_ORD_DLV_V |
| Customer Order Pick Discrepancy Report | PICK_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_FUL_ORD_PICK_DISC_V |
| Customer Order Pick Report | PICK_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_FUL_ORD_PICK_V |
| Customer Order Report | ORDER_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_FUL_ORD_V |

**Table 3-2    (Cont.) Operational Reports**

| Report Name | Report Parameters | Primary Views or Tables |
|---|---|---|
| Customer Order Reverse Pick Report | REVERSE_PICK_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_FUL_ORD_RV_PICK_V |
| Direct Delivery AGSN Report | CARTON_ID,COPIES | DSD_LINE_ITEM_UIN, ITEM_UIN |
| Direct Delivery Discrepant Items Report | RECEIPT_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_DSD_DISCREPANT_ITM_V, RPRT_DSD_V |
| Direct Delivery Label Report | CARTON_ID,LOCALE_ID | STOREDSD,DSD_CARTON,DSD_LINE_ITEM,SUPPLIER,ADDRESS, |
| Direct Delivery Report | RECEIPT_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_DSD_V, NOTES |
| Inventory Adjustment AGSN Report | INV_ADJUST_ID, COPIES | ITEM_UIN, INV_ADJUST_LINE_ITEM_UIN |
| Inventory Adjustment Report | INV_ADJUST_ID, LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_INV_ADJUST_V, CONFIG_SYSTEM |
| Item Basket Report | ITEM_BASKET_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | ITEM_BASKET, CODE_DETAIL |
| Item Detail Report | ITEMID,STOREID,LOCALE_ID,STORE_TIMEZONE,COPIES | STORE_SEQUENCE_ITEM,STORE_SEQUENCE_AREA,PRINT_FORMAT,TSF_ALLOCATION, ITEM,WAREHOUSE,RPRT_ITEM_DE TAIL_V |
| Purchase Order Report | PURCHASE_ORDER_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_PURCHASE_ORD_V |
| RFID History Report | ITEM_ID, FROM_DATE, TO_DATE, LOCALE_ID, COPIES | RPRT_RFID_HISTORY_V |
| RTV Report | RETURN_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_RTV_V |
| RTV Shipment BOL Report | SHIP_NUMBER,LOCALE_ID, STORE_TIMEZONE,COPIES | RPRT_RTV_SHIP_BOL_V, NOTES |
| RTV Shipment Container Report | CARTON_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_RTV_SHIP_V |
| RTV Shipment Report | SHIP_NUMBER,LOCALE_ID, STORE_TIMEZONE,COPIES | RPRT_RTV_SHIP_V |
| RTV Shipping Label Report | CARTON_ID,LOCALE_ID, COPIES | RPRT_RTV_SHIP_BOL_V,RTV_SHIP,CODE_DETAIL |

**Table 3-2    (Cont.) Operational Reports**

| Report Name | Report Parameters | Primary Views or Tables |
|---|---|---|
| Scan List Report | REPLENISH_GAP_ID,LOCALE_ID,STORE_TIMEZONE, COPIES | RPRT_REPLENISH_GAP_V |
| Shelf Adjustment Report | SHELF_ADJUST_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_SHELF_ADJUST_V |
| Shelf Replenishment Report | SHELF_REPLENISH_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_SHELF_REPLENISH_V |
| Stock Count All Location Report | STORE_ID,STOCK_COUNT_ID,COPIES | RPRT_STOCK_COUNT_V |
| Stock Count Detail Report | STOCK_COUNT_ID,STOCK_COUNT_CHILD_ID, STORE_TIMEZONE,PHASE,COPIES | RPRT_STOCK_COUNT_V |
| Stock Count Export Report | STOCK_COUNT_ID,COPIES | STOCK_COUNT_LINE_ITEM,STOCK_COUNT,STOCK_COUNT_LINE_ITEM_ UIN |
| Stock Count Rejected Item Report | STORE_ID,LOCALE_ID,COPIES | RPRT_STOCK_COUNT_NOF_V |
| Store Order Report | STORE_ORDER_ID, STORE_TIMEZONE, LOCALE_ID | RPRT_STORE_ORDER_V, STORE_ORDER, STORE_ORDER_CFA, STORE_ORDER_CDA, CUSTOM_ATT_ADMIN |
| Transfer Receiving AGSN Report | CARTON_ID,COPIES | TSF_DELV_LINE_ITEM_UIN,ITEM_UIN |
| Transfer Receiving Exception Report | DELIVERY_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | TSF_DELV,TSF_DELV_CARTON,TSF_DELV_LINE_ITEM,ITEM,STORE,WAREHOUSE,PARTNER,CONFIG_SYSTEM |
| Transfer Receiving Label Report | CARTON_ID,LOCALE_ID | TSF_DELV,TSF_DELV_CARTON,TSF_DELV_LINE_ITEM,STORE,ADDRESS |
| Transfer Receiving Report | DELIVERY_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | TSF_DELV,TSF_DELV_CARTON,TSF_DELV_LINE_ITEM,ITEM,STORE_ITEM_STOCK,STORE,WAREHOUSE,PARTNER,CONFIG_SYSTEM,NOTES |
| Transfer Report | TRANSFER_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_TRANSFER_V |

**Table 3-2    (Cont.) Operational Reports**

| Report Name | Report Parameters | Primary Views or Tables |
| --- | --- | --- |
| Transfer Shipment BOL Report | SHIPMENT_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | TSF_SHIP,TSF_SHIP_CARTON,TSF_SHIP_LINE_ITEM,ITEM,CONFIG_SYSTEM,SHIPMENT_BOL,SHIPMENT_CARTON_DIM,SHIPMENT_CARRIER_SERVICE,<br><br>SHIPMENT_CARRIER, STORE,ADDRESS,NOTES |
| Transfer Shipment Container Report | CARTON_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | TSF_SHIP,TSF_SHIP_CARTON,TSF_SHIP_LINE_ITEM,ITEM,STORE,WAREHOUSE,PARTNER,CONFIG_SYSTEM,SHIPMENT_REASON |
| Transfer Shipment Report | SHIPMENT_ID,LOCALE_ID,STORE_TIMEZONE,COPIES | RPRT_TSF_SHIP_V,<br><br>NOTES |
| Transfer Shipping Label | CARTON_ID,LOCALE_ID | TSF,TSF_SHIP,TSF_SHIP_CARTON,<br>TSF_SHIP_LINE_ITEM,ITEM,SHIPMENT_BOL,STORE,ADDRESS |

# 4
# Internationalization

Internationalization is the process of creating software that can be translated easily. SIOCS has been internationalized to support multiple languages.

This section covers the following:

- Supported Locales
- SOCS Client Translations
- EICS Client Translations
- EICS Server Translations
- Translation Topics
- Translation Keys
- Translation Setup Screen
- Translation File Upload
- Report Translations

## Supported Locales

SIOCS supports translation into following locales:

1. Arabic
2. Chinese (Simplified)
3. Chinese (Traditional)
4. Croatian
5. Dutch
6. English
7. French
8. German
9. Greek
10. Hungarian
11. Italian
12. Japanese
13. Korean
14. Polish
15. Portuguese (Brazilian)
16. Russian
17. Spanish

18. Swedish

19. Turkish

Apart from these, extension hooks are added for following new locales on EICS:

1. Albanian

2. Armenian

3. Azerbaijani

4. Belarusian

5. Bengali

6. Bosnian

7. Bulgarian

8. Burmese

9. Czech

10. Danish

11. Estonian

12. Filipino

13. Finnish

14. Georgian

15. Hebrew

16. Hindi

17. Indonesian

18. Kazakh

19. Khmer

20. Lao

21. Latvian

22. Lithuanian

23. Malay

24. Norwegian

25. Romanian

26. Serbian

27. Slovak

28. Slovene

29. Thai

30. Ukrainian

31. Urdu

32. Uzbek

33. Vietnamese

Translation records for these locales are defaulted to English. Translation value can be updated for these locales by accessing administration screen.

# SOCS Client Translations

Translation of SOCS Graphical User Interface (GUI) and client-based display messages fall under this category.

SOCS client follows an XML format to organize translation records within a translation bundle. Each supported locale will have its on XLF file. For example, translations for French locale could be found under SimMobileViewControllerBundle_fr.xlf file. These XLF files are packaged with the rest of the application when the mobile application is built for deployment and any changes to them will require a new deployment of the mobile application.

# EICS Client Translations

Translation of EICS Graphical User Interface (GUI) and client-based display messages fall under this category.

EICS Admin UI translations relies on following two bundle categories:

- Framework bundles: owned by JET/JRAF/LUX.

- EICS bundles: owned by EICS.

All these bundles are merged at runtime to provide an overall translation bundle which is used to provide translated UI content.

EICS owned translation are maintained in the EICS database. EICS client translation bundle is generated at runtime depending on the user locale and is applied on top of framework bundles by a custom plugin to provide an overall translation bundle for the client.

# EICS Server Translations

Translation of server data, report data, notifications, server error messages, and other server-based message, fall under this category. These translation records are maintained in EICS database and are translated via a cached server translation provider.

# Translation Topics

EICS translation records are grouped under translations topics for ease of management through the administration screens. Each translation key belonging to one of the translation topics below:

**Table 4-1    Translation Topics**

| Translation Topic | Comments |
| --- | --- |
| Barcode | Captures translation keys for barcode processors. |
| Batch | Captures translation keys for batches. |
| Carrier | Captures translation keys for shipment carrier and carrier services. |
| Code Info | Captures translation keys for code type and code details. |

**Table 4-1    (Cont.) Translation Topics**

| Translation Topic | Comments |
| --- | --- |
| Configuration | Captures translation keys for system, store and store default configuration parameters. |
| Custom Attributes | Captures translation keys related to custom attributes. |
| Data | Captures translation keys related to system data like status, types and so on. |
| Date Import | Captures translation keys related to data import. |
| Inventory Adjustment Reason | Captures translation keys for inventory adjustment reason codes. |
| Isn Type | Captures translation keys related to ISN types. |
| Message | Captures translation keys related to server messages which are mostly error messages. |
| Non Sellable Type | Captures translation keys for non-sellable types. |
| Notification | Captures notification related translation keys |
| Reports | Captures translation keys related to reports. |
| Retail Home | Captures translation keys related to retail home tile reports. |
| Security | Captures translation keys for security permissions, groups and roles. |
| Shipment Reason | Captures translation keys for shipment reason codes. |
| UI | Captures translation keys related to user interface. |

# Translation Keys

EICS translation keys follow a dotted naming convention (for example, functional.area.key). The key name also identifies the functional area it belongs to which makes it easy to locate on the Translation Setup EICS Admin Client screen. Each key has a corresponding translation for each language. These translations can be modified using the administration screen.

# Translation Setup Screen

This EICS admin client screen can be used to look up and edit EICS client and server translation records for a translation locale supported by the system. This screen does not support addition and deletion of translation records.

# Bundle Selection Dialog

Bundle selection dialog automatically popups up when you navigate to the Translation Setup Screen.

Select a bundle here and click **Apply** to load translation records.

**Figure 4-1    Bundle Selection Dialog**
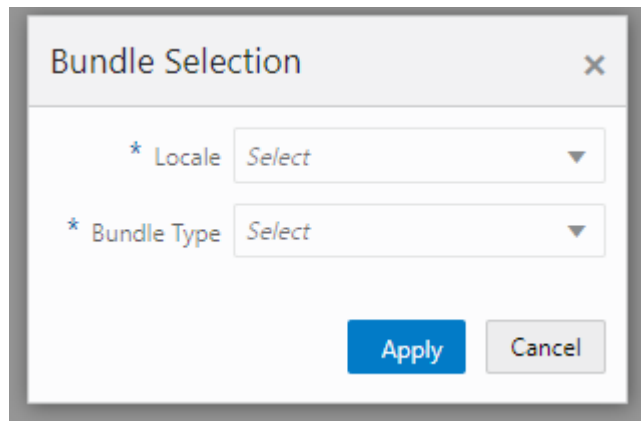


- **Locale Selection**: This drop-down will list all the translation locales supported by EICS server.
- **Bundle Type Selection**: This drop-down will list the translation bundles available for customization.

The screen supports following two bundle types:

- **Server**: This bundle identifies EICS Server translation records.
- **Operations UI**: This bundle identifies EICS Client translation records.

**Dialog Buttons**

- **Apply**: Clicking this button will load the translation records for the selected bundle criteria.
- **Cancel**: Clicking this button will close the dialog without performing any action.

# Translation Setup Screen

The screen allows customization of EICS owned translation records only.

**Figure 4-2    Translation Setup Screen**



**Navigation**: Main Menu/Admin/Translations/Translation Setup

**Search Bar Options**

- **Change Locale**: Clicking this button will popup bundle selection dialog.

- **Filter criteria**: This area displays the current search criteria.

- **Import**: Clicking this button will display the file import dialog. Refer to Translation File Upload section for more details.

**List Buttons**

- **Save**: Saves any changes made to the translation records.

- **Refresh**: Refreshes the translation records by loading them again.

- **Grid View Menu**: It's a drop-down menu that provides access to options like reset view, enable/disable column filter and export grid data to a CSV file.

**List Attributes**

- **Topic**: Translation topic for the translation record.

- **Key**: Translation key for the translation record.

- **Translation**: Actual translated text for the translation record.

- **Description**: Any additional description for the translation record.

**Detail Buttons**

- **Edit**: Enable editing of translation record.

- **Apply**: Apply changes to the translation record.

- **Cancel**: Cancel any changes made to the translation record.

**Detail Attributes**

- **Topic**: Translation topic for the translation record. It is not editable.

- **Key**: Translation key for the translation record. It is not editable.
- **Translation**: Actual translated text for the translation record. It is editable.
- **Description**: Any additional description for the translation record. It is editable.

In addition to the Translation Setup screen, the system maintains translation records for the following JET screens:

**Table 4-2    Translation Data JET Screens**

| JET Screen | Column |
| --- | --- |
| Barcode Processor | Processor Name |
| Carrier | Description |
| Carrier Service | Description |
| Code Info | Description |
| Inventory Adjustment | Description |
| Shipment Reason | Description |
| Sub-bucket | Description |
| Custom Flexible Attribute | Display Label |
| Role Detail | Description |

# Translation File Upload

Translation Setup Screen allow the user to upload translations file. This allows the user to update translations in a file and import that file into the product.

Clicking on the **Import** button on **Translation Setup Screen** displays this dialog that can be used to download the template and upload updated files for the current locale.

**Figure 4-3    Import Translation File**
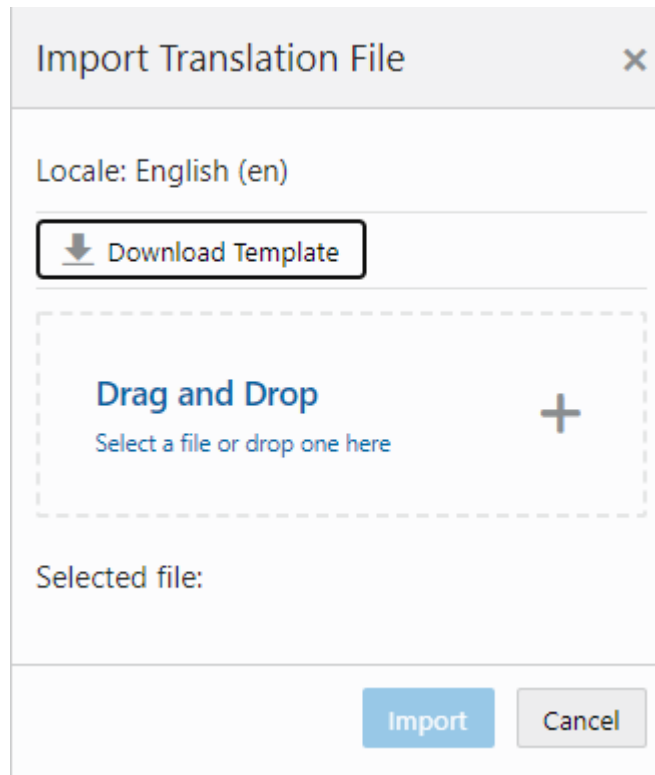


- **Locale**: It displays the selected locale.

- **Download Template**: Clicking on this button downloads a template CSV file which can be used to edit translation values across all available bundles for the selected locale. It does not matter which bundle is currently selected on the **Translation Setup Screen**. The file name would be of the format SIOCS_Translations_<locale>_Template.xlsx where locale is the selected locale.

- **Drag and Drop**: To select an updated file for import, the file can be dragged and dropped into the 'Drag and Drop' space on the dialog. Alternatively, a user can browse the file using the file selection wizard by clicking inside the 'Drag and Drop' space. File size should be > 0 MB and <= 5MB and should belong to the selected locale.

- **Selected File**: It displays the name of the selected file.

- **Import**: Clicking this button initiates the file import process. If there were any errors in the processing, an error file will be generated, and the user will be prompted if they want to save the file.

- **Cancel**: Clicking this button will close the dialog without taking any action.

## Report Translations

Translation of report templates fall under this category. EICS provides XLF files for each report. At runtime BI publisher identifies the user locale and selects the appropriate XLF file to translate report template.

# 5
# Batches

This section information describes the following topics:

- Batch Overview
- Batch Admin Users
- Batch Configuration
- Operational Batches
- BDI-Based Batches
- SFTP File-Based Batches
- Purge Batches
- Batch Job Administration

## Batch Overview

The batch processes are designed to process large volume of data.

The batch jobs which are available for the retailer can be differentiated in following categories:

**Operational Batch Jobs**

These jobs are mostly used to perform background operations on the transaction and includes such tasks as generation of a transaction, closing the transaction after a specified date is passed, auto confirmation of the transaction, and so on.

**Data Purge Jobs**

These jobs are used to perform the purge operation on the application table. Each job is designed to purge the data from specific functional database tables. The number for day to retain the records in database can be configured separately via System Administration parameters.

**Importer Batch Jobs**

These jobs are used to perform import data from another system.

## Batch Admin Users

The following list shows the batch related users. For additional details, please see the *Oracle Retail Enterprise Inventory Cloud Service User Guide*.

**Table 5-1    Batch Users and Roles**

| Job Duties | SIOCS Application Role | IDCS or OCI IAM Application Role |
| --- | --- | --- |
| Manage Batch System Configuration | ADMINISTRATOR | admin_users |
| Access SIOCS Job Admin UI | ADMINISTRATOR | admin_users |
| Manage SIOCS Adhoc Job | ADMINISTRATOR | batch_users |

# Batch Configuration

To access the System Configuration screen, navigate Main Menu/Admin/ Configuration/System Administration.

To view the Batch configuration, filter by Batch topic.

**Figure 5-1    Batch System Administration Screen**
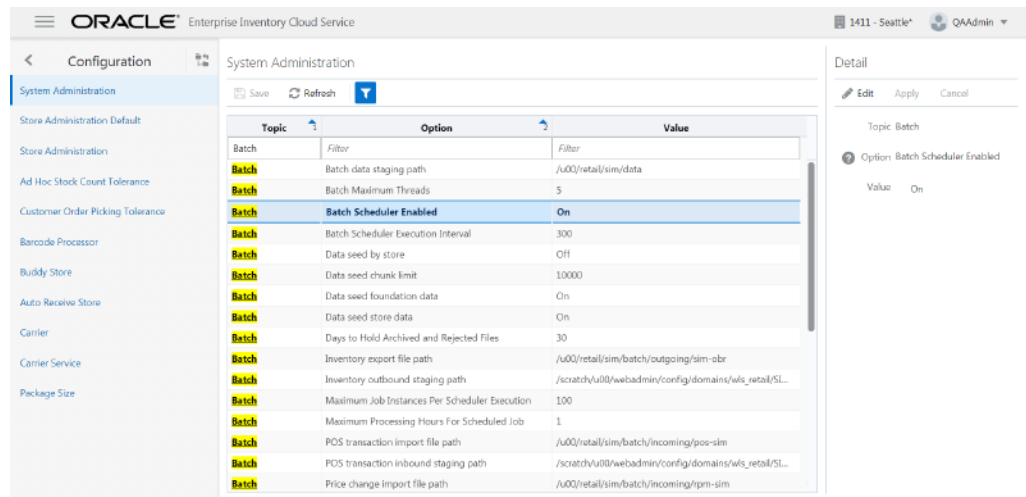


**Table 5-2    Batch System Configuration Options**

| Name | Option | Description | Restricted |
| --- | --- | --- | --- |
| archive.dump.interval.days | Archive dump internal days | Interval in days to create dump files for archived purged data. | Yes |
| archive.max.rows.per.job | Archive maximum rows per batch job | Archive maximum rows per batch job. | Yes |
| batch.max.file.per.job | Batch max file per job | Batch max files per job run. | No |
| batch.data.staging.path | Batch data staging path | Base staging directory for the file processing. | Yes |

**Table 5-2    (Cont.) Batch System Configuration Options**

| Name | Option | Description | Restricted |
|------|--------|-------------|------------|
| batch.job.max.threads | Batch Maximum Threads | Maximum number of threads to be instantiated for batch execution. | No |
| batch.scheduler.enabled | Batch Scheduler Enabled | Batch Scheduler Enabled | Yes |
| batch.scheduler.interval.duration.secs | Batch Scheduler Execution Interval | Batch Scheduler Execution Interval | No |
| batch.scheduler.max.job.per.run | Maximum Job Instances Per Scheduler Execution | Maximum Job Instances Per Scheduler Execution | No |
| batch.scheduler.max.processing.hours.for.job | Maximum Processing Hours For Schedule Job | Maximum Processing Hours For Schedule Job | No |
| days.to.hold.archived.rejected.files | Days to Hold Archived and Rejected Files | Number of days to hold the archived and rejected files after being processed. | No |
| dataseed.by.store | Data seed by store | Enables/Disables data seeding per store. | Yes |
| dataseed.chunk.limit | Data seed chunk limit | Chunk limit for records processed in data seeding. | Yes |
| dataseed.foundation.enabled | Data seed foundation data | Enables/Disables the foundation data seeding. | Yes |
| dataseed.storedata.enabled | Data seed store data | Enables/Disables the store data seeding. | Yes |
| file.export.arvdump.path | Archive dump file export path | Archive dump file export path. | Yes |
| file.export.inventory.path | Inventory export file path | Outgoing directory for inventory extract zip files. | Yes |
| file.export.stock.count.path | Stock count export file path | Outgoing directory for U/A Stock count extract zip files. | Yes |
| file.import.initial.inventory.path | Initial inventory import file path | Incoming directory for Initial inventory upload files. | Yes |
| file.import.pos.transaction.path | POS transaction import file path | Incoming directory for POS transaction zip files. | Yes |
| file.import.price.change.path | Price change import file path | Incoming directory for Price change zip files. This is for on-Premise RPM, will be phased out in future release. | Yes |
| file.import.resa.path | Retail sale audit import file path | Incoming directory for Sales audit zip files. | Yes |
| file.import.stock.count.path | Third-party stock import file path | Incoming directory for Third-party stock count zip files. | Yes |
| file.import.store.sequence.path | Store sequence import file path | Incoming directory for store sequencing zip files. | Yes |

**Table 5-2 (Cont.) Batch System Configuration Options**

| Name | Option | Description | Restricted |
|------|--------|-------------|------------|
| file.import.warehouse.stock.path | Warehouse item stock import file path | Incoming directory for warehouse inventory import zip files. | Yes |
| file.in.stage.initial.inventory.path | Initial inventory inbound staging path | Staging directory for the Initial inventory import file processing. | Yes |
| file.in.stage.pos.transaction.path | POS transaction inbound staging path | Staging directory for the POS transaction import file processing. | Yes |
| file.in.stage.price.change.path | Price Change inbound staging path | Staging directory for the Price change import file processing. | Yes |
| file.in.stage.resa.path | Retail sale audit inbound staging path | Staging directory for the Sales audit import file processing. | Yes |
| file.in.stage.stock.count.path | Stock count inbound staging path | Staging directory for the third-party stock count import file processing. | Yes |
| file.in.stage.store.sequence.path | Store sequence inbound staging path | Staging directory for the Sequencing data import file processing. | Yes |
| file.in.stage.warehouse.stock.path | Warehouse item stock inbound staging path | Staging directory for the Warehouse inventory import file processing. | Yes |
| file.out.stage.arvdump.path | Archive dump file outbound staging path | Archive dump file outbound staging path. | Yes |
| file.out.stage.inventory.path | Inventory outbound staging path | Staging directory for the inventory extract file. | Yes |
| file.out.stage.stock.count.path | Stock count outbound staging path | Staging directory for the U/A Stock Count export file. | Yes |
| procedure.logPurgingInfo.enabled | Procedure Log Purging Info Enabled | Procedure Log Purging Info Enabled. | Yes |
| search.limit.batch.job.days | Search Limit Default for Batch Job Days | Default number of days search limit to load the job activity. | No |
| sftp.mfcs.dest.path | MFCS SFTP incoming default path | MFCS SFTP incoming default path. For example, /mfcsftpuser. | Yes |
| sftp.mfcs.file.out.compress | MFCS Compress data files | MFCS Compress data files. if true, compress data files in zip file. | Yes |
| sftp.mfcs.file.out.push | MFCS SFTP push | MFCS SFTP push. if true, push data files to MFCS sftp default incoming path; if set to false, the files will be moved to EICS's sftp export location. | Yes |

**Table 5-2    (Cont.) Batch System Configuration Options**

| Name | Option | Description | Restricted |
|------|--------|-------------|------------|
| sftp.mfcs.host | MFCS SFTP host | MFCS SFTP host | Yes |
| sftp.mfcs.port | MFCS SFTP port | MFCS SFTP port | Yes |
| sftp.mfcs.user.name | MFCS SFTP User | MFCS SFTP User name | Yes |
| sftp.obr.dest.path | Order Broker SFTP incoming default path | Order Broker SFTP incoming default path, for example, /obrftpuser. | Yes |
| sftp.obr.file.out.compress | Order Broker Compress data files | Order Broker Compress data files. If true, compress data files in zip file. | Yes |
| sftp.obr.file.out.push | Order Broker SFTP push | Order Broker SFTP push. if true, push data files to OBR sftp default incoming path; if set to false, the files will be movedto EICS's sftp export location. | Yes |
| sftp.obr.host | Order Broker SFTP host | Order Broker SFTP host. | Yes |
| sftp.obr.port | Order Broker SFTP port | Order Broker SFTP port. | Yes |
| sftp.obr.user.name | Order Broker SFTP user | Order Broker SFTP user. | |
| sftp.push.max.files | The max files push to SFTP server per job run | The max files push to SFTP server per job run, default to 100 files. | Yes |
| sftp.push.timeout | The SFTP push timeout in milliseconds | The SFTP push timeout in milliseconds. | Yes |
| sftp.shell.log.path | The SFTP push shell script log path | The SFTP push shell script log path. | Yes |
| sftp.shell.working.path | The SFTP push shell script working path | The SFTP push shell script working path. | Yes |
| stock.count.import.chunk.limit | Stock Count Import Chunk Limit | Stock Count Import Chunk Limit | Yes |

> **Note:**
>
> Purge related configurations are listed under purge topic, for complete list of system configuration list, see Configuration chapter for details. Configuration marked as restricted can only be managed by sys op admin user.

## Operational Batches

Operational batches are used to perform background operations on the transaction and includes such tasks as generation of a transaction, closing the transaction after a specified date is passed, auto confirmation of the transaction, and so on.

5-6

- Archive Activity History
- Archive Dump
- Archive Item Uin History
- Archive RFID History
- Archive Store Item Stock History
- Archive Ticket History
- Auto Replenish Capacity
- Auto Ticket Generate
- Auto Ticket Print
- Cleanup Batch Directories
- Cleanup Shelf Replenishment
- DSD Receiving Closure
- Extract Subscription Usage
- Fulfillment Order Reminders
- Fulfillment Order Pick Reminders
- Generate Problem Line Stock Count
- Generate Unit Amount Stock Count
- Generate Unit Stock Count
- Inventory Extract File SFTP Push
- Item Basket Maintenance
- Item Price to History
- Product Group Schedule Clean up
- Return Not After Date Alert
- Stock Count Authorize Recovery
- Stock Count Export
- Stock Count Export File SFTP Push
- Stock Count Unit and Amount Snapshots
- Store Order Auto Approve
- Store Order Auto Cancel
- Store Order Auto Generate
- Transfer Close
- Transfer Delivery Auto Receive
- Transfer Delivery Close
- Transfer Not After Date Alert Batch
- Transfers Overdue
- Vendor Return Closure

# Archive Activity History

This batch moves aged activity history into archive table.

**Batch Job Definition Name**

Audits_ArchiveJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-3    Key Tables for Archive Activity History**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| ACTIVITY_HISTORY | Yes | | | Yes |
| ARV_ACTIVITY_HIST | | Yes | | |

# Archive Dump

This batch generates the Oracle data dump files from archived data purging record tables. After the data dump files are generated, the archived tables are truncated. For Archive Data Dump Files Export, see Batch - Data Archive chapter.

**Batch Job Definition Name**

ArchiveDump_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-4    Key Tables for Archive Dump**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| ARV_ACTIVITY_HIST | Yes | | | Yes |
| ARV_ITEM_UIN_HIST | Yes | | | Yes |
| ARV_STORE_ITEM_STOCK_HIST | Yes | | | Yes |
| ARV_RFID_HIST | Yes | | | Yes |
| ARV_EXPORT | Yes | Yes | Yes | |

# Archive Item Uin History

This batch moves aged activity history into archive table.

**Batch Job Definition Name**

ItemUinHistory_ArchiveJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-5    Key Tables for Archive Item Uin History**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| ITEM_UIN_HISTORY | Yes | | | Yes |
| ARV_ITEM_UIN_HIST | | Yes | | |

# Archive RFID History

This batch moves aged RFID history records into archive table.

**Batch Job Definition Name**

RfidHistory_ArchiveJob

**Batch Job Parameters**

N/A

**Key Tables**

**Table 5-6    Key Tables for Archive RFID History**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| RFID_HISTORY | Yes | | | |
| ARV_RFID_HIST | | Yes | | |

# Archive Store Item Stock History

This batch moves aged store item stock history records into archive table.

**Batch Job Definition Name**

StoreItemStockHistory_ArchiveJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-7    Key Tables for Archive Store Item Stock History**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| STORE_ITEM_STOCK_HISTORY | Yes | | | Yes |
| ARV_STORE_ITEM_STOCK_HIST | | Yes | | |

## Archive Ticket History

This batch moves aged ticket history records into archive table.

**Batch Job Definition Name**

TicketHistory_ArchiveJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-8    Key Tables for Archive Ticket History**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| TICKET_HISTORY | Yes | | | Yes |
| ARV_TICKET_HIST | | Yes | | |

## Auto Replenish Capacity

This job automatically updates the shopfloor to capacity for certain items based on a product group.

**Batch Job Definition Name**

AutoReplenishCapacity_OpsJob

**Batch Job Parameters**

<input_date>input_dateinput date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-9    Key Tables for Auto Replenish Capacity Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| shelf_replenish | Yes | Yes | Yes | Yes |
| shelf_replenish_line_item | Yes | Yes | Yes | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Auto Ticket Generate

This job automatically generates the item tickets and labels depending on the store parameters for events which are subscribes. The events are price changes, clearance event, promotions and item description changes.

**Batch Job Definition Name**

TicketAutoGenerateFromEvent_OpsJob

**Batch Job Parameters**

<input_date>input_dateinput date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-10    Key Tables for Auto Ticket Generate Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| ticket_event | Yes | Yes | Yes | |
| ticket | Yes | Yes | Yes | Yes |
| ticket_format | Yes | | | |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Auto Ticket Print

This batch sends the tickets generated to the 3rd party printing service. The tickets/labels generated for the items identified in the product group will be automatically sent to the 3rd party printing service.

**Batch Job Definition Name**

TicketAutoPrint_OpsJob

**Batch Job Parameters**

<input_date>input_dateinput date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-11    Key Tables for Auto Ticket Print Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| Store_printer | Yes | | | |
| ticket | Yes | Yes | Yes | Yes |
| ticket_format | Yes | | | |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Cleanup Batch Directories

This job deletes the processed files from batch archive folder and failed files from reject folder of the respective job's directory. The retention period is specified by system configuration parameter **Days to Hold Archived and Rejected Files**.

**Batch Job Definition Name**

CleanupBatchDirectories_OpsJob

**Batch Job Parameters**

<input_date>

input_dateinput date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-12    Key Tables for Cleanup Batch Directories**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| ACTIVITY_HISTORY | Yes | | | |

# Cleanup Shelf Replenishment

The end of day batch process runs at the end of each day to reset the delivery bay and close any open pending shelf replenishments. The system takes the entire inventory from the delivery bay and moves it to the back room. Any pending or in progress shelf replenishment are changed to a cancelled state. Users who are performing a shelf replenishment are kicked out of the system. That is, the batch process takes over the shelf replenishment user's application activity locking. The current user's shelf replenishment process is discarded without being saved. After the batch process is run, all shelf replenishments are either completed or cancelled, and the delivery bay has zero inventory.

**Batch Job Definition Name**

CleanupShelfReplenishment_OpsJob

**Batch Job Parameters**

input_date>input_dateinput date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-13    Key Tables for Cleanup Shelf Replenishment Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| shelf_replenish | | | Yes | |
| stock_item_stock | | | Yes | |

# DSD Receiving Closure

This batch program looks for all the open vendor deliveries whose expected date added to store parameter "Auto Close Days after Expected Date" is before today and automatically confirms all the vendor deliveries.

**Batch Job Definition Name**

DSDReceivingClosure_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-14    Key Tables for Close DSD Receiving Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| activity_history | | Yes | | |
| config_store | Yes | | | |
| dsd | Yes | | Yes | |
| dsd_carton | Yes | | Yes | |
| dsd_line_item | Yes | | Yes | |
| dsd_line_item_att | Yes | | Yes | |
| dsd_line_item_uin | Yes | | Yes | |
| inv_adjust_reason | Yes | | | |
| purchase_order | Yes | | Yes | |
| stock_item_v | Yes | | | |
| store | Yes | | | |

# Extract Subscription Usage

This batch is used to extract subscription usage for EICS and SOCS respectively during the subscription period.

It is a restricted batch which by default is scheduled to run every month. This schedule can only be updated by a sysop user.

The batch inserts a usage record in the SUBSCRIPTION_USAGE table for each month within the subscription period starting from the last extract date+1 (or subscription start date) till the end of last month from the batch run date.

When calculating usage metrics, the from date is the latest date between the subscription start date and the last extract date+1. If there were no previous extractions, the subscription start date is considered.

The subscription start date is obtained from PSRAF view RAF_SUBS_CURNT_METRIC_QTY_VW.

The batch also inserts a record in the PSRAF table RAF_SUBS_METRIC_USAGE with monthly usage and running usage info for EICS and SOCS respectively.

The monthly usage for a month is defined below:

**EICS**: The total quantity of items that were sold and returned plus the total quantity of items that were fulfilled as part of web orders.

**SOCS**: The distinct users that logged into SOCS.

The running usage is as defined below:

**EICS**: Pooled monthly usage over the subscription period starting from the subscription start date till usageToDate.

**SOCS**: Average monthly usage during a subscription year within the subscription period where each subscription year is 12 months long.

**Batch Job Definition Name**

ExtractSubscriptionUsage_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-15    Key Tables for Extract Subscription Usage**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| subscription_usage | Yes | Yes | | |
| pos_transaction | Yes | | | |
| ful_ord | Yes | | | |
| ful_ord_line_item | Yes | | | |
| activity_history | Yes | | | |

# Fulfillment Order Pick Reminders

This batch process sends out alerts for fulfillment order picks for which create date has expired by minutes to hold customer orders before sending alert parameter value and the status is new or in progress.

**Batch Job Definition Name**

FulfillmentOrderPickReminders_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-16    Key Tables for Fulfillment Order Pick Reminders Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| ful_ord_pick | Yes | | | |
| ful_ord_pick_line_item | Yes | | | |
| store_item_stock | Yes | | | |
| store | Yes | | | |
| comfig_system | Yes | | | |

# Fulfillment Order Reminders

This batch process sends out alerts for fulfillment orders for which create date has expired by minutes to hold customer orders before sending alert parameter value.

**Batch Job Definition Name**

FulfillmentOrderReminders_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-17    Key Tables for Fulfillment Order Reminders Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| config_store | Yes | | | |
| store | Yes | | | |
| store_item_stock | Yes | | | |
| ful_ord | Yes | | | |
| ful_ord_line_item | Yes | | | |

# Generate Problem Line Stock Count

Before the batch process runs, the retailer establishes a group of items and item hierarchies (by associating them to the problem line group type) and selects applicable parameters (negative SOH, negative available, and so on). The problem line batch process goes through the list of items in the group, determining which fall within the parameters. The system automatically creates a stock count from those items that do fall within the parameters.

If an item is a problem line item (negative inventory for example) on a stock count, and the user does not get the chance to perform the stock count on it that day, the next day the item may no longer be a problem line (positive inventory). However, the system continues to create a stock count for that item because a problem existed at one time.

**Batch Job Definition Name**

GenerateProblemLineStockCount_OpsJob

**Batch Job Parameters**

Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id> Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-18    Key Tables for Problem Line Stock Count Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| group_schedule_extract | Yes | Yes | | |
| prod_group_item_bkdn | | | Yes | Yes |
| stock_count | Yes | Yes | Yes | Yes |
| stock_count_line_item | Yes | Yes | Yes | Yes |
| stock_count_line_item_uin | Yes | Yes | Yes | Yes |
| stock_count_child | Yes | Yes | Yes | Yes |
| product_group_schedule | Yes | | Yes | |
| product_group | Yes | | | |
| product_group_sched_store | Yes | | | |
| item | Yes | | | |
| store_item | Yes | | | |
| stock_count_line_item | Yes | | | |

# Generate Unit Amount Stock Count

This batch program generates Unit Amount stock counts.

Daily, the batch process creates the stock counts that are scheduled for the current day or future date which matches the next scheduled date. The system looks at all the scheduled stock count records and determines whether any are scheduled for today or the user-specified future date. The process creates the stock counts for each individual store. For example, if a scheduled count includes a list of five stores, then five separate stock count records are created.

If an all-location stock count is being run, the batch processing generates individual counts for every macro sequence location.

The date parameter is optional when running the Extract Stock Counts batch. If no date is provided, today's date is used.

**Batch Job Definition Name**

GenerateUnitAmountStockCount_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-19    Key Tables for Generate Unit Amount Stock Count Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| group_schedule_extract | | Yes | | Yes |
| product_group | Yes | | | |
| product_group_hierarchy | Yes | | | |
| product_group_item | Yes | | | |
| product_group_sched_store | Yes | | | |
| product_group_schedule | Yes | | Yes | |
| product_group_item_bkdn | | Yes | | Yes |
| stock_count | Yes | Yes | Yes | |
| stock_count_child | | Yes | Yes | |
| stock_count_line_item | | Yes | Yes | |
| stock_count_line_item_uin | | Yes | Yes | |
| item | Yes | | | |
| store_item | Yes | | | |
| store_item_stock | Yes | | | |
| item_component | Yes | | | |

# Generate Unit Stock Count

This batch program generates Unit stock counts.

Daily, the batch process creates the stock counts that are scheduled for the current day or future date which matches the next scheduled date. The system looks at all the scheduled stock count records and determines whether any are scheduled for today or the user specified future date. The process creates the stock counts for each individual store. For example, if a scheduled count includes a list of five stores, then five separate stock count records are created.

If the system is configured to use unguided stock counts, the batch process does not generate multiple counts even if the item is located at multiple locations within the store.

The date parameter is optional when running the Extract Stock Counts batch. If no date is provided, today's date is used.

**Batch Job Definition Name**

GenerateUnitStockCount_OpsJob

**Batch Job Parameters**

<input_date>

Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id> Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-20    Key Tables for Generate Unit Stock Count Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| group_schedule_extract | | Yes | | Yes |
| product_group | Yes | | | |
| product_group_hierarchy | Yes | | | |
| product_group_item | Yes | | | |
| product_group_sched_store | Yes | | | |
| product_group_schedule | Yes | | Yes | |
| product_group_item_bkdn | | Yes | | Yes |
| stock_count | Yes | Yes | Yes | Yes |
| stock_count_child | | Yes | Yes | Yes |
| stock_count_line_item | | Yes | Yes | Yes |
| stock_count_line_item_uin | | Yes | Yes | |
| item | Yes | | | |
| store_item | Yes | | | |
| store_item_stock | Yes | | | |
| item_component | Yes | | | |

# Inventory Extract File SFTP Push

This batch scan the inventory extract staging out directories, move the files which have the completer files from EICS app server to Order Broker SFTP Server if system configuration - **OBR SFTP push** is set to true.

**Batch Job Definition Name**

InventoryExtractSftpPush_OpsJob

**Batch Job Parameters**

<input_date>Where input_date is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

N/A

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Item Basket Maintenance

This batch updates the item basket status cancelled when item basket has expired.

**Batch Job Definition Name**

ItemBasketMaintenance_OpsJob

**Batch Job Parameters**

<input_date>Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-21    Key Tables for Item Basket Maintenance Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| config_system | Yes | | | |
| Item_basket | Yes | | Yes | |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Item Price to History

This batch writes the active item price records into item price history table. After the active item prices are recorded in the item price history table, the batch updates the ITEM_PRICE table statuses as completed for these records.

**Batch Job Definition Name**

ItemPriceToHistory_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.<store id> Where

store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-22    Key Tables for Item Price to History Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| item_price | Yes | | Yes | Yes |
| item_price_history | | Yes | | |

# Product Group Schedule Clean up

This batch program searches for all open product group schedules that have ended date before today (or user specified date) and change the product group schedule status to closed.

**Batch Job Definition Name**

ProductGroupScheduleCleanup_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-23    Key Table for Close Prod Group Schedule Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| Product_group_schedule | Yes | | Yes | |

# Return Not After Date Alert

This batch process warns users a number of days in advance that the RTV/RTW is about to reach the **Not After** date and must be dispatched. The value for the number of days of advance warning is configurable using the system's administration screens.

**Batch Job Definition Name**

ReturnNotAfterDateAlert_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-24    Key Tables for Return Not After Date Alert Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| rtv | Yes | | | |
| rtv_line_item | Yes | | | |
| shipment_reason | Yes | | | |
| stock_item | Yes | | | |

# Stock Count Authorize Recovery

This batch process looks for stock counts that are stuck in Authorize Processing state. This is a unique state that appears when an error occurs during the final processing of a stock count. The batch attempts to fully authorize the stock count. Errors that occur during the batch process are logged to the server error logs and will indicate the reason for any further processing failures. Successfully authorized stock counts will move to authorized completed state.

**Batch Job Definition Name**

StockCountAuthorizeRecovery_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-25    Key Tables for Stock Count Authorize Recovery Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| stock_count | Yes | | Yes | |
| stock_count_child | Yes | | Yes | |
| stock_count_line_item | Yes | | Yes | |
| stock_count_line_item_uin | Yes | | | |
| item_uin | Yes | | Yes | |
| store_item | Yes | | | |
| store_item_stock | | | Yes | |
| product_group_schedule | Yes | | | |
| product_group_sched_store | Yes | | | |
| store | Yes | | | |
| stock_count_sale | Yes | | | Yes |
| inv_adjust_reason | Yes | | | |

# Stock Count Export

This batch process looks for a stock count that is stuck in approval authorized state during authorizing a unit amount stock count process. This is a unique state that appears when an error occurs during the final processing of a unit amount stock count. The batch attempts to generate stock count export file and set stock count status to authorize complete state. Errors that occur during the batch process are logged to the server error logs and will indicate the reason for any further processing failures. See Appendix: Unit and Amount Stock Counts Export for details.

**Batch Job Definition Name**

StockCountExport_OpsJob

**Batch Job Parameters**

<stock_count_id>

Where the stock_count_id is the stock count identifier

**Key Tables**

**Table 5-26    Key Tables for Stock Count Export Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| stock_count | Yes | | Yes | |
| stock_count_child | Yes | | Yes | |
| stock_count_line_item | Yes | | Yes | |
| stock_count_line_item_uin | Yes | | | |
| item_uin | Yes | | Yes | |
| store_item | Yes | | | |
| stock_count_export | Yes | | Yes | Yes |

# Stock Count Export File SFTP Push

This batch scan the stock count export staging out directories, move the files which have the completer files from EICS app server to MFCS SFTP Server if system configuration - `MFCS SFTP push` is set to true.

**Batch Job Definition Name**

StockCountExportSftpPush_OpsJob

**Batch Job Parameters**

<input_date>Where input_date is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

N/A

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Stock Count Unit and Amount Snapshots

This job takes the snapshot of current inventory for Unit and Amount Type of Stock Counts for the items across all stores or for a specific store.

**Batch Job Definition Name**

StockCountUnitAndAmountSnapshot_OpsJob

**Batch Job Parameters**

<store id> Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-27    Key Tables for Stock Count Unit And Amount Snapshot Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| stock_count | Yes | | Yes | |
| stock_count_child | Yes | | Yes | |
| stock_count_line_item | Yes | | Yes | |
| Store_item | Yes | | | |
| store_item_stock | Yes | | | |

# Store Order Auto Approve

This batch processes looks for store orders which requested date is X days of old than the system defined **Days before auto approving Store Orders**, and the process attempts to set those store orders to approved state when applicable.

**Batch Job Definition Name**

StoreOrderAutoApprove_OpsJob

**Batch Job Parameters**

N/A

**Key Tables**

**Table 5-28    Key Tables for Store Order Auto Approve Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| store_order | Yes | | Yes | |
| store_order_line_item | Yes | | Yes | |

# Store Order Auto Cancel

This batch processes looks for store orders which requested date is X days of old than the system defined **Days to hold before Auto Canceling Store Orders**, and the process attempts to set those store orders to cancel state when applicable.

**Batch Job Definition Name**

StoreOrderAutoCancel_OpsJob

**Batch Job Parameters**

N/A

**Key Tables**

**Table 5-29    Key Tables for Store Order Auto Cancel Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| store_order | Yes | | Yes | |
| store_order_line_item | Yes | | Yes | |

# Store Order Auto Generate

This batch processes generate store order records for store order product group schedules.

**Batch Job Definition Name**

StoreOrderAutoGenerate_OpsJob

**Batch Job Parameters**

N/A

**Key Tables**

**Table 5-30    Key Tables for Store Order Auto Generate Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| store_order | Yes | Yes | Yes | |
| store_order_line_item | Yes | Yes | Yes | |
| group_schedule_extract | | Yes | | Yes |
| product_group | Yes | | | |
| product_group_hierarchy | Yes | | | |
| product_group_item | Yes | | | |
| product_group_sched_store | Yes | | | |

**Table 5-30 (Cont.) Key Tables for Store Order Auto Generate Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| product_group_schedule | Yes | | Yes | |

# Transfer Close

This batch program looks for all the open transfers which have past their not after date and are in valid state for closure.

**Batch Job Definition Name**

TransferClose_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-31 Key Tables for Close Transfers Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| activity_history | | Yes | | |
| config_store | Yes | | | |
| mps_staged_message | | Yes | | |
| store | Yes | | | |
| store_item_stock | Yes | | Yes | |
| tsf | Yes | | Yes | |
| tsf_line_item | Yes | | Yes | |
| tsf_ship | Yes | | | |
| tsf_ship_carton | Yes | | | |
| tsf_ship_line_item | Yes | | | |

# Transfer Delivery Auto Receive

Transfer Delivery Auto Receive batch does the following:

- Retrieves a list of all stores.

- Retrieves the auto receive configuration option for the location types (Store, Warehouse, and Finisher).

- For each store, if the Auto Receive store parameter is set to Date Driven, then the batch auto-receives all deliveries that are in New and In Progress status and whose Ship Date added to the Auto Receive Number of Days is less than the current date.

**Batch Job Definition Name**

TransferDeliveryAutoReceive_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-32    Key Tables for Auto Receive Transfer Deliveries**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| activity_history | | Yes | | |
| config_store | Yes | | | |
| inv_adjust_reason | Yes | | | |
| item_uin | Yes | | Yes | Yes |
| store_item_stock | Yes | | Yes | |
| store_item_stock_history | | Yes | | |
| store_sequence_area | Yes | | | |
| store_sequence_item | Yes | | | |
| tsf | Yes | | Yes | |
| tsf_allocation | Yes | | Yes | |
| tsf_delv | Yes | | | |
| tsf_delv_carton | Yes | | Yes | Yes |
| tsf_delv_line_item | Yes | | Yes | |
| store | Yes | | | |

# Transfer Delivery Close

This batch program looks for all the open transfer deliveries and auto confirms all the transfer deliveries based on the store parameter "Auto Close Receipt".

When the parameter value is "0", close the deliveries at the end of day today and when value is "x" close the deliveries at the end of "x" days stating from today.

**Batch Job Definition Name**

TransferDeliveryClose_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-33    Key Tables for Close Transfer Deliveries Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| activity_history | | Yes | | |
| config_store | Yes | | | |
| inv_adjust_reason | Yes | | | |
| item_uin | Yes | | Yes | Yes |
| store_item_stock | Yes | | Yes | |
| store_item_stock_history | | Yes | | |
| store_sequence_area | Yes | | | |
| store_sequence_item | Yes | | | |
| tsf | Yes | | Yes | |
| tsf_allocation | Yes | | Yes | |
| tsf_delv | Yes | | | |
| tsf_delv_carton | Yes | | Yes | Yes |
| tsf_delv_line_item | Yes | | Yes | |
| store | Yes | | | |

# Transfer Not After Date Alert Batch

This batch process generates alerts for any pending transfer requests with not after date coming up within number of days specified in the system parameter "Days to Send Email Alert Before Not After Date for Transfer Requests".

**Batch Job Definition Name**

TransferNotAfterDateAlert_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-34    Key Tables for Transfers Not After Date Alert Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| tsf | Yes | | | |
| tsf_line_item | Yes | | | |
| store | Yes | | | |

# Transfers Overdue

This batch process sends user alert for dispatched transfers which have not been received after a number of days. The value for the number of days of alert is configurable using the system's administration screen.

**Batch Job Definition Name**

TransfersOverdueBatch_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-35    Key Tables for Transfers Overdue Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| tsf_delv | Yes | | | |
| tsf_delv_carton | Yes | | | |

# Vendor Return Closure

This batch program looks for all the open vendor returns which are in valid state (Closed /Rejected) for closure.

**Batch Job Definition Name**

VendorReturnClosure_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-36    Key Tables for Close Vendor Return**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| rtv | Yes | Yes | Yes | |
| rtv_line_item | Yes | Yes | Yes | Yes |
| rtv_line_item_uin | | | | Yes |
| store_item_stock | Yes | | Yes | |
| shipment_reason | Yes | | | |

# BDI-Based Batches

BDI (Bulk Data Integration) is the Oracle Retail Integration tool for bulk data movement. The BDI infrastructure applications move data from one application to another. So, there is data producing applications and data consuming applications. See BDI-Based Batch Integration in Integration chapter for BDI Integration Process details.

This section describers the importer details.

- Clearance Transaction Import
- Price Change Transaction Import
- Rfid Transaction Import
- Third Party Pricing Import
- Warehouse Avail Inventory Import

## Clearance Transaction Import

This batch imports the clearance price changes from BDI Inbound Interface tables into EICS application tables. This importer job is an integral part of the Clearance Price BDI Process Flow.

**Batch Job Definition Name**

Clearance_Tx_ImporterJob

**Batch Job Parameters**

<dataSetId>

**Key Tables**

**Table 5-37    Key Tables for Clearance Transaction Import**

| Table |
| --- |
| CLEARANCE_IN |
| ITEM_PRICE |

## Price Change Transaction Import

This batch imports the regular price changes from BDI Inbound Interface tables into EICS application tables. This importer job is an integral part of the Price Change TX BDI Process Flow.

**Batch Job Definition Name**

PriceChange_Tx_ImporterJob

**Batch Job Parameters**

<dataSetId>

If dataSetId is not specified, then each batch job run will find the last 5 datasets within last 7 days and process the datasets in parallel threads via Batch partitioning.

**Key Tables**

**Table 5-38    Key Tables for Price Change Transaction Import**

| Table |
| --- |
| PRICE_CHANGE_IN |
| ITEM_PRICE |

# Rfid Transaction Import

This batch imports the external RFID from BDI Inbound Interface tables into EICS application tables.

This importer job is an integral part of the BDI process.

**Batch Job Definition Name**

ExtRfid_Tx_ImporterJob

**Batch Job Parameters**

<dataSetId>

If dataSetId is not specified, then each batch job run will find the last 5 datasets within last 7 days and process the datasets in parallel threads via Batch partitioning.

**Key Tables**

**Table 5-39    Key Tables for Rfid Transaction Import**

| Table |
| --- |
| EXT_RFID_IN |
| RFID |

# Third Party Pricing Import

This batch imports the external item pricing from BDI Inbound Interface tables into EICS application tables.This importer job is an integral part of the BDI process.

**Batch Job Definition Name**

ExtPrice_Tx_ImporterJob

**Batch Job Parameters**

<dataSetId>

If dataSetId is not specified, then each batch job run will find the last 5 datasets within last 7 days and process the datasets in parallel threads via Batch partitioning.

**Key Tables**

**Table 5-40    Key Tables for Third Party Pricing Import**

| Table |
| --- |
| EXT_PRICE_IN |
| ITEM_PRICE |

> **✎ Note:**
>
> Special Notes to Pricing Imports:
>
> By default, EICS integrate with RPCS clearance and regular price change via BDI process flow.
>
> EICS does not integrate with Promotion Offer interface due to it does not provide pricing at item/date level, so there is nothing for SIOCS to consume.
>
> Third Party Pricing Import provides an alternative for import promotion data into EICS.
>
> See Appendix: Third Party Pricing Integration for additional details.

# Warehouse Avail Inventory Import

This batch imports the warehouse available inventory from BDI Inbound Interface tables into EICS application tables. This importer job is an integral part of the BDI process.

**Batch Job Definition Name**

InvAvailWh_Tx_ImporterJob

**Batch Job Parameters**

<dataSetId>

If dataSetId is not specified, then each batch job run will find the last 5 datasets within last 7 days and process the datasets in parallel threads via Batch partitioning.

**Key Tables**

**Table 5-41    Key Tables for Warehouse Avail Inventory Import**

| Table |
| --- |
| INV_AVAIL_WH_IN |
| WAREHOUSE_ITEM |

# SFTP File-Based Batches

EICS SFTP-Based File batches are file integration via SFTP.

The following topics are covered:

- Clearance Price Change Import (with On-premise RPM)
- High Level Batch File Process Overview
- Initial Inventory Import
- Inventory Extract
- Pos Transaction Import
- Promotion Price Change Import (with On-premise RPM)
- Regular Price Change Import (with On-premise RPM)
- Retail Sale Audit Import
- SFTP Nightly Batch File Uploads
- Store Sequence Import
- Third Party Stock Count Import
- Warehouse Item Stock Import

## Clearance Price Change Import (with On-premise RPM)

See Appendix: Pricing Import with On-premise RPM.

> **Note:**
>
> Pricing file import with On-premise RPM will be phased out in future release.
>
> See Clearance Transaction Import for clearance pricing integration with RPCS.

## High Level Batch File Process Overview

- The file import job processes each file from incoming directory when a corresponding completer file presents. The completer is a zero-byte file which indicates file transfer is complete and ready for processing.
- The batch file process also handles the movement of input, output and reject files to the respective directories.
- Output files and reject files will be pushed to outgoing for client review and usage.
- Incoming files can be placed on SFTP as single file or zipped by each file type.
- Outgoing files will be zipped by file type and pushed to outgoing location.

# Initial Inventory Import

Most Oracle Retail customers implement Retail Merchandising Foundation Cloud Service (RMFCS) followed by SIOCS. In some instances, SIOCS is implemented prior to RMFCS. In those cases, there is a need for importing initial inventory data for a store or group of stores from legacy or non-Oracle/third-party systems during the implementation phase.

This enhancement addresses a specific use case of initial inventory upload when stores are brought live on SIOCS by seeding initial inventory without any validations.

This batch is meant to be used ONLY during implementation. It is designed for optimal upload to rewrite SOH. This batch will not perform any validations. It should not be used after the one-time inventory upload.

Note: UINs must be uploaded only once. Unlike stock on hand, UINs are state driven and control the stock on hand. Only new UINs in the flat file will be considered for stock on hand update. As such, if UINs already exist, they will not reflect into the new SOH.

The Initial Inventory Import batch is used to wipe out the existing SOH data for items in a store and override it with the new SOH data from the third-party/non-Oracle/legacy systems.

The data gets imported via a flat file.

**File Specification**

**File Name format**: EXTSTK_<date in YYYYMMDDHH24MISS format>.dat

**File Layout**: See Initial Inventory Import File Layout Specification.

**Batch Job Definition Name**

InitialInventoryImport_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if file staging fails. The staging process is all or none, so if an error occurs during the batch import process, none of the transactions in the file will be staged. The user will need to rerun the batch using the same file after resolving the errors.

For all invalid records in the file such as Unit UOM items with decimal quantity, items that are not ranged to the store, item records with negative quantity, an error record will be inserted in the BATCH_DATA_ERROR table for each invalid record.

**Key Tables**

**Table 5-42    Key Tables for Initial Inventory Import Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| store_item_stock | Yes | | Yes | |
| item_uin | Yes | Yes | Yes | |

# Inventory Extract

This batch extracts the inventory to a file that has been altered on or after the specified date for the specified store.

**File Specification**

**File Name format**: PRODUCT_LOCATION_INV _<store id>_<date in YYYYMMDDHH24MISS format>.txt

Where the store id is unique store identifier.

**File Layout**: See Inventory Extract Export File Specification.

**Batch Job Definition Name**

InventoryExtract_OpsJob

**Batch Job Parameters**

<input_date>Where input date parameter is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.<storeId>: If provided, only specified store will be processed. When store id is not specified, the batch will generate exports for all EICS managed stores.

**Key Tables**

**Table 5-43    Key Tables for Inventory Extract Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| item | Yes | | | |
| store_item | Yes | | | |
| store_item_stock | Yes | | | |

# Pos Transaction Import

This batch imports pos transaction records from the flat file (SIMT-LOG file) that came from POS database staging table where polling timer framework will pick those staged requests and update the stock tables.

The batch process takes the sales/order transaction data and stage them to the database staging table from where they are picked up by the polling timer framework to update the store item's inventory buckets (for example, store item's total quantity, shop floor quantity), if applicable.

The file will contain both sale and order transactions. The batch will assign separate request IDs to sales and order transactions.

For sale transactions, a single request ID cannot contain more than MAX_VALUE = 500 transaction line items with an exception that a single transaction ID cannot span across multiple request IDs.

For order transactions, a single request ID cannot contain more than MAX_VALUE = 500 transaction line items with an exception that a single customer order ID cannot span across multiple request IDs.

The file contains transactions for a single store.

Each job run will pick number of files (defined by **Maximum Job Instances Per Batch**) in system configuration and process them in multi-threads. The number of files to be processed is default to 20, the value can be configured via system configuration screen.

Customer can set the job scheduler to be run multiple times per day by changing the schedule intervals.

**File Specification**

**File Name format**: SIMTLOG _<date in YYYYMMDDHH24MISS format>_<loc id>.dat

where loc id is the store identifier.

> **✎ Note:**
>
> The integration incoming files from file provider which are in old release which are not using the .zip and completer indicator file. EICS is backward compatible with the incoming files which are not in zip and without completer indicator file.

**File Layout**: See POS Sale Transaction Import File Specification.

**Batch Job Definition Name**

PosTransactionImport_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if the file staging fails. The staging process is all or none transaction so if an error occurs during the batch process, none of the transactions in the file will be staged. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table 5-44    Key Tables for Pos Transaction Import Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| pos_transaction | Yes | Yes | | |
| item | Yes | | | |
| inv_adjust_reason | Yes | | | |

# Promotion Price Change Import (with On-premise RPM)

See Appendix: Pricing Import with On-premise RPM.

> **Note:**
>
> Pricing file import with On-premise RPM will be phased out in future release.
>
> See Appendix: Third Party Pricing Integration batch for promotional price import as alternative.

# Regular Price Change Import (with On-premise RPM)

See Appendix: Pricing Import with On-premise RPM.

> **Note:**
>
> Pricing file import with On-premise RPM will be phased out in future release.
>
> See Price Change Transaction Import for price integration with RPCS.

# Retail Sale Audit Import

This batch program imports sales/order transaction data (ReSA File) that originated in Oracle Retail Xstore Point of Service. The external audit system will provide in its sales upload file a percentage or quantity that indicates how much the inventory needs to be reduced by, in addition to the sold quantity.

For example, meat will become lighter as fluids evaporate. Other items, for example cheese or ham, will only be reduced when of the outside layers are cut off to sell the item.

The batch process takes the sales transaction data to update the store item's inventory buckets. From the batch program, SIOCS learns about inventory movement (that is, what is sold, what is returned, what is reserved and what is fulfilled). Once SIOCS attains the data, SIOCS assumes that sales should be taken from the store's shelf-related inventory buckets. This assumption is important to SIOCS 's shelf replenishment processing. SIOCS assumes that returns should go to the backroom bucket; the system's logic is that returns must be inspected.

The batch takes the sales/order transaction data and stage them to the SIOCS database staging table from where they are picked up by the polling timer framework to update the store item's inventory buckets (for example, store item's total quantity, shop floor quantity), if applicable.

The file will contain both sales and order transactions. The batch job combines the transaction number and register number to form the transaction ID in SIOCS. Request IDs are assigned to the transactions in such a way that a single request ID will not

contain more than MAX_SIZE=500 records with an exception that a single transaction ID should not span across multiple request IDs.

During processing staging records, batch also writes each failure record into a transaction log table.

Each job run will pick number of files (defined by Maximum Job Instances Per Batch) in system configuration and process them in multi-threads. The number of files to be processed is default to 20, the value can be configured via system configuration screen.

Customer can set the job scheduler to be run multiple times per day by changing the schedule intervals.

**File Specification**

**File Name format**: SIMT_<date in YYYYMMDDHH24MISS format>_<loc id>

Where <loc id> is store id.

**File Layout**: See Retail Sale Audit Import File Specification Specification.

**Batch Job Definition Name**

RetailSalesAuditImport_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if the file staging fails. The staging process is all or none transaction so if an error occurs during the batch process, none of the transactions in the file will be staged. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table 5-45    Key Tables for Retail Sale Audit Import Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| pos_transaction | | Yes | | |
| inv_adjust_reason | Yes | | | |

# SFTP Nightly Batch File Uploads

The following is the SFTP file upload process. The Private/Public Keys must be generated, and the public Key must be associated with your SFTP Account for the file uploads. The Adding Authorized Keys section describes the step-by-step method to generate the Keys (2048-bit RSA Keys).

- Adding Authorized Keys
- Login to WinSCP
- Uploading a Batch File

• **Export File Downloads**

## Adding Authorized Keys

Use this process to generate a 2048-bit RSA key and add the same to the SFTP server. With Windows, use the WinSCP tool or with Linux, use ssh-keygen.

1. Launch WinSCP and select Tools/Run PuttyGen.

2. Select SSH-2 RSA for the type of key to generate and enter 2048 for the number of bits in a generated key field and click **Generate**.

**Figure 5-2    Key Generator**
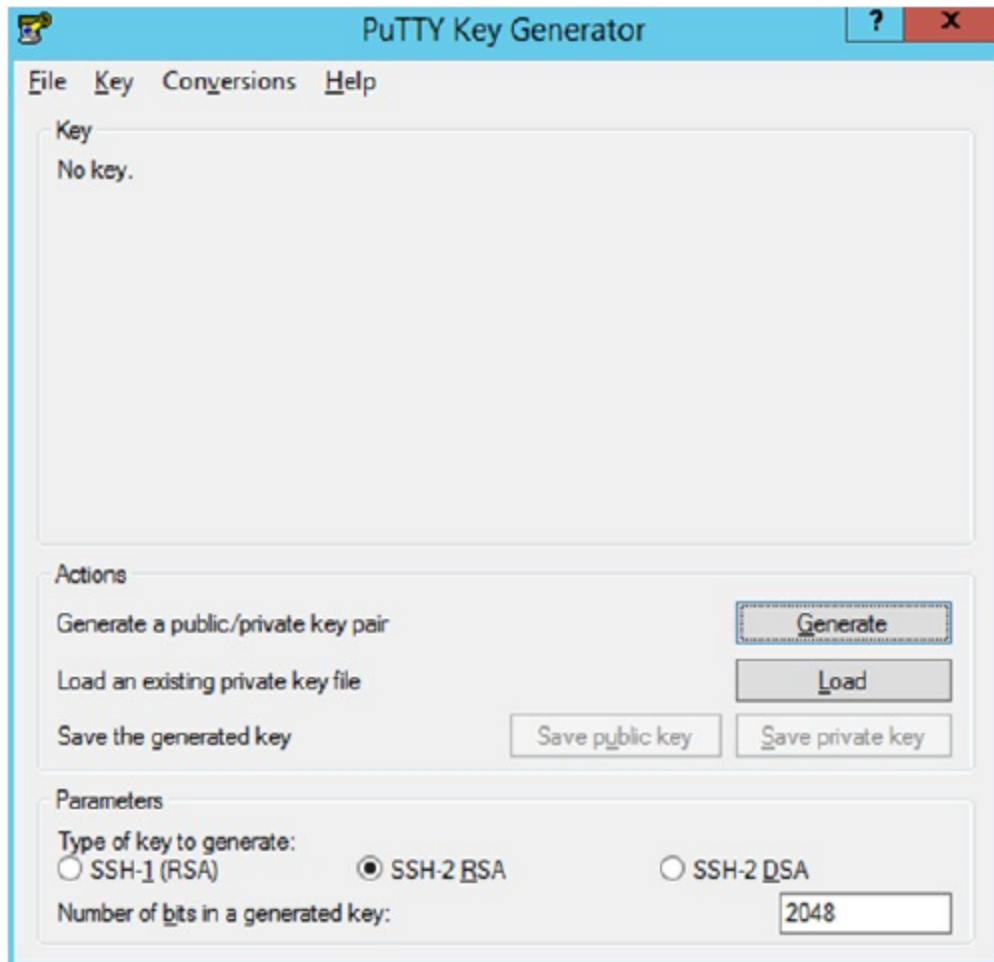


3. Move the mouse over the blank space in the window until the key is generated.

Chapter 5
SFTP File-Based Batches

**Figure 5-3    Key Generator Progress**
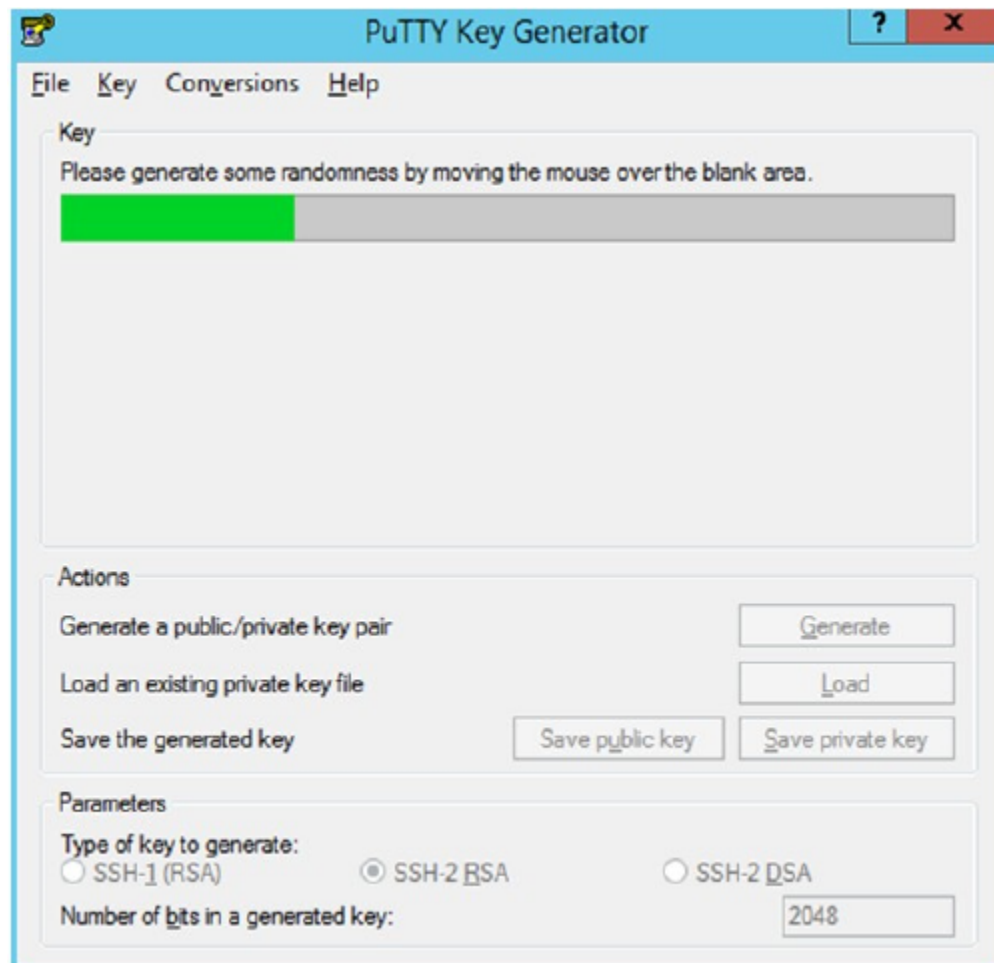


4. Once the key is generated, click **Save public key** to save the public key to a file.

5. Click **Save private key** to save the Private key to a file. Confirm to save it with or without a passphrase.

6. Open an SR with Oracle Support, to associate the Public half of the Key with your SFTP account (attach the Key with the SR).

## Login to WinSCP

These upload steps use the private key generated in section, Adding Authorized Keys.

1. Launch WinSCP and connect to <SFTP Server> using port 22.

2. Enter the username and then click **Advanced**.

3. Click **Authentication**.

4. In the Private Key File field, click **Browse** and select the private key created in the section, Adding Authorized Keys.

5-39

**Figure 5-4    Advanced Site Settings**
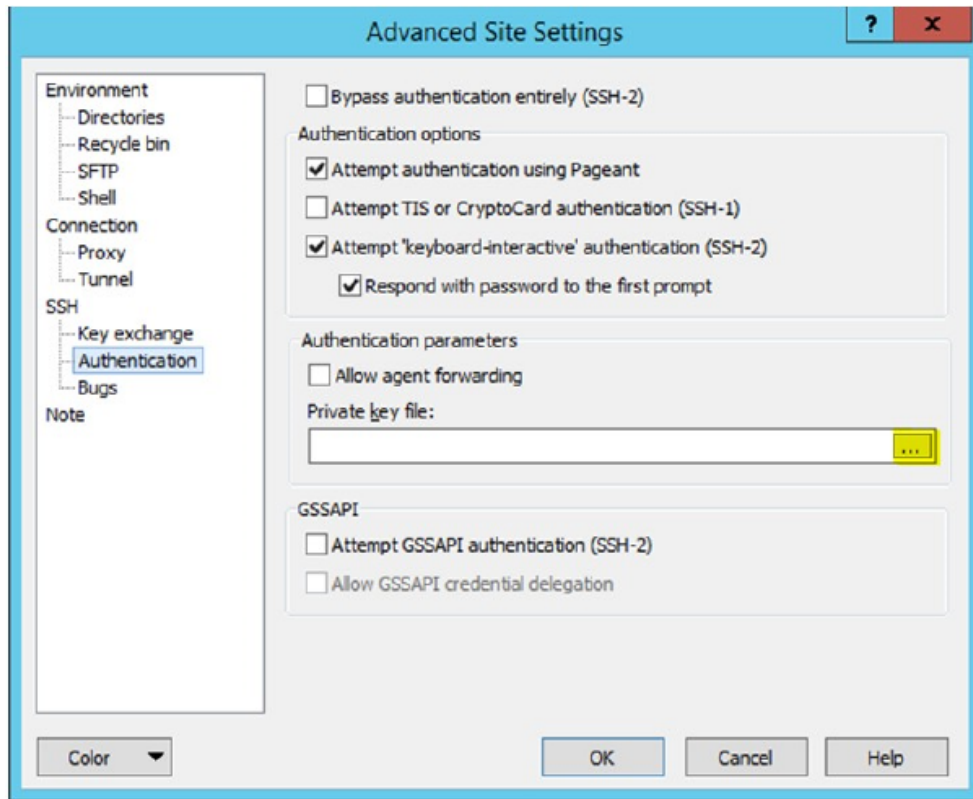


5.    After loading the private key file, click **OK.**

**Figure 5-5    Private Key File Loaded**
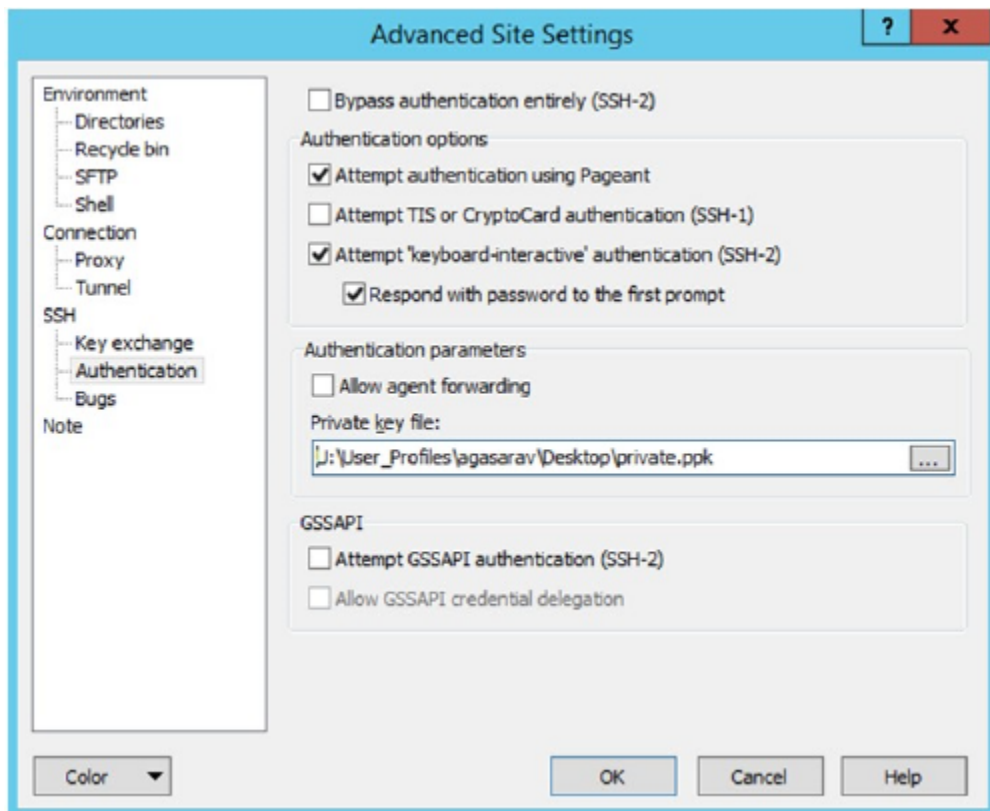


6. Click **Login**. The window does not prompt for a password and logs into the SFTP server. Provide a passphrase if one has been set up.

> **Note:**
>
> Login can only be performed using the authorized keys. Login with username / password is not supported.

## Uploading a Batch File

Login to the WinSCP by following the steps in Login to WinSCP.

1. Transfer the file to be copied (for example, test) to /<SFTP User>.

**Figure 5-6    <SFTP User> Directory**

2. Transfer an empty file <filename>.complete (for example, test.complete) to the directory /<SFTP User>.

**Figure 5-7     Transferring Empty File**



3. If multiple files have to be transferred, copy all the files to /<SFTP_user>.

**Figure 5-8     Transferring Multiple Files**



4. Transfer all the corresponding <filename>.complete files to the /<SFTP_user> directory for the transfer to complete.

**Figure 5-9     Transferring .complete Files**



## Export File Downloads

The following steps show the download file process.

1. Login to the WinSCP by following the steps in Login to WinSCP.

2. Change the directory to /<SFTP User>/EXPORT.

3. Download all data files.

## Store Sequence Import

This batch imports store sequencing information from a flat file. Before importing, the batch will delete the existing sequencing information including sequence items and

sequence areas excluding no-location store area which is the default store sequence area.

Each job run will pick number of files (defined by **Maximum Job Instances Per Batch**) in system configuration and process them in multi-threads. The number of files to be processed is default to 20, the value can be configured via system configuration screen.

Customer can set the job scheduler to be run multiple times per day by changing the schedule intervals.

**File Specification**

**File Name format**: SSEQ_<date in YYYYMMDDHH24MISS format>_<loc id>

Where <loc id> is store id.

**File Layout**: See Store Sequence Data Parser Import File Layout Specification.

**Batch Job Definition Name**

StoreSequenceImport_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The file loading process is all or none transaction so if an error occurs during the batch process, none of the transactions in the file will be committed. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table 5-46    Key Tables for Store Sequence Import Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| store_sequence_area | | Yes | | Yes |
| store_sequence_item | | Yes | | Yes |

# Third Party Stock Count Import

This batch imports the stock count quantities which are setup in SIOCS and physical counting is conducted by a third-party. The batch updates the store stock on hand quantities; invalid records are saved in the rejected item table.

When the stock count is set up as auto-authorize **Unit and Amount Stock Count**, a **Unit and Amount Stock Counts Export** file is generated as a result of stock count auto authorization. See Appendix: Unit and Amount Stock Counts Export for details.

Each job run will pick number of files (defined by **Maximum Job Instances Per Batch**) in system configuration and process them in multi-threads. The number of files to be processed is default to 20, the value can be configured via system configuration screen.

Customer can set the job scheduler to be run multiple times per day by changing the schedule intervals.

**File Specification**

**File Name format**: STK_<date in YYYYMMDDHH24MISS format>.dat

**File Layout**: See Third Party Stock Count Parser Import File Layout Specification.

**Batch Job Definition Name**

ThirdPartyStockCountImport_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if the file staging fails. The staging process is all or none transaction so if an error occurs during the batch process, none of the transactions in the file will be staged. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table 5-47    Key Tables for Third Party Stock Count Import Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| stock_count_import | Yes | | Yes | |
| stock_count_rejected_item | | Yes | | |
| stock_count | Yes | | Yes | |
| stock_count_child | Yes | | Yes | |
| stock_count_line_item | Yes | | Yes | |
| item_price | Yes | | | |
| item | Yes | | | |
| store_item | Yes | | | |
| item_uin | Yes | | | |
| stock_count_line_item_uin | Yes | | | |

# Warehouse Item Stock Import

This batch imports the inventory of an items in a warehouse from flat files into warehouse item table. The batch updates the warehouse item total quantity.

Each job run will pick number of files (defined by **Maximum Job Instances Per Batch**) in system configuration and process them in multi-threads. The number of files to be processed is default to 20, the value can be configured via system configuration screen.

Customer can set the job scheduler to be run multiple times per day by changing the schedule intervals.

**File Specification**

**File Name format**: WHIMS_<date in YYYYMMDDHH24MISS format>.dat

**File Layout**: See Warehouse Stock Import File Layout Specification.

**Batch Job Definition Name**

WarehouseItemStockImport_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The import process is all or none transactions so if an error occurs during the batch process, the batch job will be marked as failed. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table 5-48    Key Tables for Warehouse Item Stock Import Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| warehouse_item | | Yes | | Yes |

# Purge Batches

- Purge Adhoc Stock Count
- Purge Areas
- Purge Batch Log
- Purge Closed Transfers
- Purge Completed UINs
- Purge DSD and Purchase Orders
- Purge Customer Orders
- Purge Invalid Users
- Purge Invalid User Roles
- Purge Inventory Adjustments
- Purge Item
- Purge Item Baskets
- Purge Item Hierarchy
- Purge Item Price

- • Purge Lockings
- • Purge Notifications
- • Purge Price Change Worksheet
- • Purge Price Histories
- • Purge Related Items
- • Purge Resolved UIN Problems
- • Purge RFID
- • Purge Sales Posting
- • Purge Shelf Adjustments
- • Purge Shelf Adjustments
- • Purge Staged Message
- • Purge Stock Counts
- • Purge Temporary UINs
- • Purge Ticket Histories
- • Purge Vendor Returns

# Purge Adhoc Stock Count

PurgeAdHocStockCount batch does the following:

- • Deletes ad hoc stock counts with a status of in progress.
- • Any ad hoc stock count with a creation date/time stamp older than the **Days to Hold In Progress Ad Hoc Counts** parameter value will be deleted. For example, the default value is 1. If the batch program is run with the default value, the batch program would delete all in-progress counts more than 24 hours old.

This batch program deletes ad hoc stock counts with a status of in progress. Any ad hoc stock count with a creation date/time stamp older than the **Days to Hold In Progress Ad Hoc Counts** parameter value will be deleted. For example, the default value is 1. If the batch program is run with the default value, the batch program would delete all in-progress counts more than 24 hours old.

**Batch Job Definition Name**

AdhocStockCount_PurgeJob

**Batch Job Parameters**

<input_date>Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-49    Key Tables for Purge AdHoc Stock Count Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| stock_count | Yes | | | Yes |
| stock_count_rejected_item | Yes | | | Yes |
| stock_count_child | Yes | | | Yes |
| stock_count_line_item | Yes | | | Yes |
| stock_count_line_item_uin | Yes | | | Yes |
| stock_count_line_item_att | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run from by starting a new batch job after the issues are resolved.

# Purge Areas

Purges product basket records (with status of cancelled) based on the retention period. The retention period is specified by system configuration parameter- **Days to Hold Area**.

**Batch Job Definition Name**

ProductBasket_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-50    Key Tables for Purge Area Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| product_basket_store | Yes | | | Yes |
| product_basket | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Batch Log

Purge Batch Log batch does the following:

- Deletes old batch log records.
- Batch log record with an end date/timestamp older than the **Days To Hold Batch Logs** system configuration value and with the Status value of 2 (COMPLETED) is deleted. For example, if the default value is 30 and the batch program is run with the default value, the batch program would delete all the records that are more than 30 days old and are in completed status.
- Delete purge error logs.

**Batch Job Definition Name**

BatchLog_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-51    Key Tables for Purge Batch Log Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| batch_data_error | Yes | | | Yes |
| batch_execution | Yes | | | Yes |
| purge_error_log | Yes | | | Yes |
| purge_error_log | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Closed Transfers

Purge Closed Transfers batch does the following:

- Deletes all the closed transfer which are in either cancelled or completed status and shipments related to them.Any closed transfer with an update date older than the **Days to Hold Transfer Documents** parameter value will be deleted. For example, the default value is 30. If the batch program is run with the default value, the batch program would delete all closed transfers, transfer requests and related shipments which are closed and was last updated 30 days ago.

**Batch Job Definition Name**

ClosedTransfers_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-52    Key Tables for Purge Closed Transfers Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| tsf | Yes | | | Yes |
| tsf_line_item | | | | Yes |
| tsf_ship | Yes | | | Yes |
| tsf_ship_carton | | | | Yes |
| tsf_ship_line_item | Yes | | | Yes |
| tsf_ship_line_item_uin | | | | Yes |
| tsf_ship_line_item_att | | | | Yes |
| shipment_bol | | | | Yes |
| tsf_delv | Yes | | | |
| tsf_delv_line_item | Yes | | | |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

## Purge Completed UINs

Purge CompletedUINs batch does the following:

- Deletes completed UIN Detail records. A completed UIN is any UIN with a status of Removed from Inventory, Missing, Sold, Shipped to Vendor, or Shipped to Warehouse.

- Any UIN detail record with a complete status and update date at least X days in the past (where X is with system parameter **Days to Hold Completed UINs**) will be deleted from ITEM_UIN and ITEM_UIN_PROBLEM table.

**Batch Job Definition Name**

CompletedUin_PurgeJob

**Batch Job Parameters**

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-53    Key Tables for Purge Completed UIN Detail Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| item_uin | Yes | | | Yes |
| item_uin_problem | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge DSD and Purchase Orders

Purge DSD Receiving and Purge orders batch process does the following:

• Deletes the Direct Store Delivery receiving.

• Any DSD record which is in Closed/Cancelled status, and which has a complete date older than **Days to Hold Received Shipments** is an eligible record for purge.

• In effect, a DSD record can be purged only if its associated PO records can be purged.

**Batch Job Definition Name**

DsdAndPurchaseOrders_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-54    Key Tables for Purge DSD Receivings Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| purchase_order | Yes | | | Yes |
| purchase_order_line_item | Yes | | | Yes |
| dsd | Yes | | | Yes |
| notes | Yes | | | Yes |
| dsd_adjustment | Yes | | | Yes |

**Table 5-54    (Cont.) Key Tables for Purge DSD Receivings Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| dsd_carton | Yes | | | Yes |
| dsd_line_item | Yes | | | Yes |
| dsd_line_item_uin | Yes | | | Yes |
| dsd_line_item_att | Yes | | | Yes |
| purchase_order | Yes | | | Yes |
| purchase_order_line_item | | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Customer Orders

Purge Customer Orders Batch does the following:

- Deletes all the fulfillment order records which are not in New or In Progress status and for which the update date has expired the purge_date by number of days more than **Days to Hold Customer Order** parameter value.

- Additionally, only those fulfillment orders will be deleted for which customer order ID and fulfillment order ID combination does not exist for any Transfer, Return, Purchase Order, and Warehouse delivery transaction.

**Batch Job Definition Name**

FulfillmentOrders_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-55    Key Tables for Purge Fulfillment Orders Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| tsf | Yes | | | |
| purchase_order | Yes | | | |
| tsf_delv_line_item | Yes | | | |
| ful_ord | Yes | | | Yes |
| ful_ord_line_item | Yes | | | Yes |

**Table 5-55    (Cont.) Key Tables for Purge Fulfillment Orders Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| ful_ord_dlv | Yes | | | Yes |
| ful_ord_dlv_line_item | Yes | | | Yes |
| ful_ord_dlv_line_item_uin | Yes | | | Yes |
| ful_ord_dlv_line_item_att | Yes | | | Yes |
| ful_ord_rv_pick | Yes | | | Yes |
| ful_ord_rv_pick_line_item | Yes | | | Yes |
| ful_ord_pick | Yes | | | Yes |
| ful_ord_pick_line_item | Yes | | | Yes |
| ful_ord_bin | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Invalid Users

This job deletes invalid application users from data store for those usernames are not found in identity store.

**Batch Job Definition Name**

InvalidUser_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-56    Key Tables for Purge Invalid Users Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| security_user | Yes | | | |
| security_user_store | Yes | | | |
| security_user_role | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Invalid User Roles

PurgeInvalidUserRoles batch program does the following:

- Removes all expired user roles and orphaned user roles (roles that were deleted by removing a store) from the SIOCS system.

- The batch process finds user role assignments that have an end date that is at least *X* days in the past (where *X* is specified by the system parameter **Days to Hold Expired User Roles**) and deletes these expired role assignments.

- The users (excluding super users) with role assignments that have no matching store assignments (orphaned role assignments) are also deleted from SECURITY_USER_ROLE table.

**Batch Job Definition Name**

InvalidUserRole_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-57    Key Tables for Purge Invalid User Roles Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| security_user | Yes | | | |
| security_user_store | Yes | | | |
| security_user_role | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Inventory Adjustments

Deletes inventory adjustments records with a create date/timestamp older than **Days To Hold Completed Inventory Adjustments** parameter value will be deleted. For example, the default value is 30. If the batch program is run with the default value, the batch program would delete all the inventory adjustment records, which are more than 30 days old.

**Batch Job Definition Name**

InventoryAdjustment_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-58    Key Tables for Purge Inventory Adjustments Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| inv_adjust | Yes | | | Yes |
| inv_adjust_line_item | Yes | | | Yes |
| inv_adjust_line_item_ uin | Yes | | | Yes |
| inv_adjust_line_item_ att | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

## Purge Item

This batch program deletes items with a status of Delete (D).

There are two segments in the PurgeItem Batch which do the following different tasks:

- Validate if the Item should be deleted.
- Delete item from all associated tables if validation check is passed.
- Validate if the item should be deleted. The Validations include:
  - If SOH of item, item parent and item grandparent is 0.
  - If any transfers exist for item, item parent and item grandparent.
  - If any RTV exists for item, item parent and item grandparent.
  - If any Inventory adjustment exists for item, item parent and item grandparent and so on.
  - If any Item Basket exists for the item.
  - If any Product Group exists for the item.
  - If any Stock Count exists for the item.
  - If any Store Order exists for the item.
  - If any Item Request exists for the item.

- – If any Direct Store Delivery exists for the item.
  - – If any Warehouse Delivery exists for the item.
- Delete item from all associated table. If the validations checks are met, the records related to the item which is marked for the purge action are deleted.

**Batch Job Definition Name**

Item_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-59    Key Tables for Purge Item Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| item_componen | Yes | | | Yes |
| Item | Yes | | | Yes |
| store_item_stock | Yes | | | |
| purchase_order_line_ite m | Yes | | | |
| rtv_line_item | Yes | | | |
| stock_count_line_item | Yes | | | |
| tsf_line_item | Yes | | | |
| rtv_ship_line_item | Yes | | | |
| dsd_line_item | Yes | | | |
| dsd_adjustment | Yes | | | |
| stock_count_line_item_u in | Yes | | | |
| tsf_ship_line_item | Yes | | | |
| tsf_delv_line_item | Yes | | | |
| item_basket_line_item | Yes | | | |
| print_store_order_item | Yes | | | |
| item_request_line_item | Yes | | | |
| ful_ord_line_item | Yes | | | |
| ful_ord_pick_line_item | Yes | | | |
| inv_adjust_line_item | Yes | | | |
| inv_adjust_template_ite m | Yes | | | |
| shelf_adjust_line_item | Yes | | | |
| shelf_replenish_line_ite m | Yes | | | |

**Table 5-59    (Cont.) Key Tables for Purge Item Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| tsf_allocation | Yes | | | |
| item_uin | Yes | | | Yes |
| store_item | Yes | | | Yes |
| item_image | Yes | | | Yes |
| store_sequence_item | Yes | | | Yes |
| store_item_stock | Yes | | | Yes |
| store_item_stock_nonsell | Yes | | | Yes |
| supplier_item_manufacture | Yes | | | Yes |
| warehouse_item | Yes | | | Yes |
| partner_item | Yes | | | Yes |
| supplier_item_uom | Yes | | | Yes |
| supplier_item | Yes | | | Yes |
| supplier_item_country | Yes | | | Yes |
| supplier_item_country_dim | Yes | | | Yes |
| item_component | Yes | | | Yes |
| item_price | Yes | | | Yes |
| item_price_history | Yes | | | Yes |
| store_uin_admin_item | Yes | | | Yes |
| item_uda | Yes | | | Yes |
| product_group_item | Yes | | | Yes |
| related_item | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Item Baskets

Purges item basket records (with status of cancelled or completed) based on the retention period. The retention period is specified by system configuration parameter-**Days to Hold Item Basket**.

**Batch Job Definition Name**

ItemBasket_PurgeJob

**Batch Job Parameters**

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-60    Key Tables for Purge Item Basket Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| scan_block_item_att | Yes | | | Yes |
| scan_block | Yes | | | Yes |
| item_basket_line_item | Yes | | | Yes |
| item_basket_cda | Yes | | | Yes |
| item_basket_hierarchy | Yes | | | Yes |
| item_basket | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Item Hierarchy

Purges all Item Hierarchies that is in delete status. It also records from referencing tables such as STOCK_COUNT_ADHOC_CONFIG and ITEM_HIERARCHY_ATTRIB.

**Batch Job Definition Name**

ItemHierarchy_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-61    Key Tables for Purge Item Hierarchy Batch**

| Table | Select | Insert | Update | Delete |
|---|---|---|---|---|
| item | Yes | | | |
| product_group_hierarchy | Yes | | | |
| stock_count_adhoc_config | Yes | | | Yes |
| item_hierarchy_attrib | Yes | | | Yes |
| item_hierarchy | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Item Price

Purge Item Price batch process does the following:

- Purges records which were expired or were marked as deleted from ITEM_PRICE table based on the retention period. The retention period is specified by system configuration parameter **Days to hold expired item price**.

Following are the rules defining records to be purged:

- **Regular Price Change**: Has status of completed or deleted, effective date was X number of days in the past (relative to the specified date if specified). At any given time, at least one completed latest regular price must be retained in ITEM_PRICE for a store item.
- **Promotion Change**: Has status of completed or deleted, and end date is number of days in the past (relative to the specified date if specified).
- **Clearance Change**: Has status of completed or deleted, and end date is number of days in the past (relative to the specified date if specified).

**Batch Job Definition Name**

ItemPrice_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-62    Key Tables for Purge Item Price Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| item_price | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Lockings

This batch process deletes lockings records from ACTIVITY_LOCK table. Any lock record with a lock date/timestamp older than **Days to Hold Locking Records** system

configuration value will be deleted. For example, the default value is 30. If the batch program is run with the default value, the batch program would delete all the shelf replenishment records, which are more than 30 days old.

**Batch Job Definition Name**

Lockings_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-63    Key Tables for Purge Lockings Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| activity_lock | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Notifications

This job deletes notifications. The retention period is specified by system configuration parameter **Days to Hold Notifications**.

**Batch Job Definition Name**

Notifications_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-64    Key Tables for Purge Notifications Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| notification | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Price Change Worksheet

This batch process deletes price change worksheet records from the staging table which are in Rejected/Completed status. Any price change record with an effective date/timestamp older than **Days To Hold Price Changes** parameter value will be deleted. For example, the default value is 30. If the batch program is run with the default value, the batch program would delete all the price change records, which are more than 30 days old.

**Batch Job Definition Name**

PriceChangeWorksheet_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-65    Key Tables for Purge Price Change Worksheet Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| price_change_worksheet | Yes | | | Yes |
| item_price | | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Price Histories

This batch process deletes price histories. At least a minimum of 4 historical prices are maintained for an item/store. **Days To Hold Price History** will determine the number of days that price histories can be kept in the database.

**Batch Job Definition Name**

PriceHistories_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

<store id>

Where store id is store identifier. If store id is not specified, then data for all managed stores will be processed in parallel processing.

**Key Tables**

**Table 5-66    Key Tables for Purge Price Histories Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| item_price_history | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Related Items

This batch process deletes the related items for which the end date has expired for more than **Days To Hold Related Items** system configuration value.

**Batch Job Definition Name**

RelatedItems_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-67    Key Tables for Purge Related Items Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| related_item | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Resolved UIN Problems

Deletes resolved UIN exception records. UIN exception records with status of resolved and resolved date is at least X days in the past (where X is system parameter **Days to Hold Resolved UIN Exceptions**) are deleted from ITEM_UIN_PROBLEM table.

**Batch Job Definition Name**

ResolvedUinProblem_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-68    Key Tables for Purge Resolved UIN Problems Batch**

| Table | Select | insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| item_uin_problem | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge RFID

This job deletes RFIDs which is not present in location. The retention period is specified by system configuration parameter **Days to Hold RFID**.

**Batch Job Definition Name**

Rfid_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-69    Key Tables for Purge Notifications Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |

**Table 5-69    (Cont.) Key Tables for Purge Notifications Batch**

| Tables | Select | Insert | Update | Delete |
|--------|--------|--------|--------|--------|
| Rfid | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Sales Posting

This batch process deletes the Point-of-Service transaction from the Oracle Retail Xstore Point of Service transaction staging table. It reads the **Days to Hold Sales Posting** and **Days to Hold Failed Sales** configuration parameters and all the transactions which are present beyond the configuration parameter are deleted. It also purges the POS transaction logs for the request IDs that are in processed status.

**Batch Job Definition Name**

SalesPosting_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-70    Key Tables for Purge Sales Posting Batch**

| Tables | Select | Insert | Update | Delete |
|--------|--------|--------|--------|--------|
| config_system | Yes | | | |
| pos_transaction | Yes | | | |
| pos_transaction_log | Yes | | | |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Shelf Adjustments

Purges shelf adjustment records (with status of completed) based on the retention period. The retention period is specified by system configuration parameter- **Days to Hold Shelf Adjustment List**.

**Batch Job Definition Name**

ShelfAdjustments_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-71    Key Tables for Purge Shelf Adjustment Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| shelf_adjust_line_item | Yes | | | Yes |
| shelf_adjust_cda | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Shelf Replenishment

This batch process deletes shelf replenishment lists which are in Completed/Cancelled state. Any shelf replenishment list record with a status date/timestamp older than **Days To Hold Shelf Replenishment** parameter value will be deleted. For example, the default value is 1. If the batch program is run with the default value, the batch program would delete all the pick list records, which are more than a day old.

In addition, the batch will also delete shelf adjustment lists which are in Complete state. Any shelf adjustment record with an update date/timestamp older than **Days to Hold Shelf Adjustment Lists** parameter value will be deleted.

**Batch Job Definition Name**

ShelfReplenishments_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-72    Key Tables for Purge Shelf Replenishment Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| shelf_adjust | Yes | | | |

**Table 5-72    (Cont.) Key Tables for Purge Shelf Replenishment Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| product_group | Yes | | | |
| shelf_replenish_line_item | Yes | | | Yes |
| shelf_replenish | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Staged Message

This batch finds integration staging records that are marked as processed or deleted, and update date is at least $X$ days in the past (where $X$ is the system parameter **Days to Hold Completed Staging Records**), the batch process deletes these records from MPS_STAGED_MESSAGE table.

Rebuilding the indexes on the MPS_STAGED_MESSAGE table each day is recommended after batch process completes.

**Batch Job Definition Name**

StagedMessage_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-73    Key Tables for Purge Staged Message Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| mps_staged_message | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Stock Counts

This batch process deletes stock counts which are in Completed/Cancelled status. Any stock count with a schedule date/timestamp older than **Days To Hold Completed Stock Counts** parameter value will get deleted. For example, the default value is 30. If the batch program is

run with the default value, the batch program would delete all the stock return records, which are more than 30 days old.

**Batch Job Definition Name**

StockCounts_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-74    Key Tables for Purge Stock Counts Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| config_system | Yes | | | |
| stock_count | Yes | | | Yes |
| stock_count_child | Yes | | | Yes |
| stock_count_rejected_item | Yes | | | Yes |
| shelf_replenish_line_item | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Temporary UINs

This batch process deletes temporary UIN records. UIN records with no status and update date is at least $X$ days in the past (where $X$ is system parameter **Days to Hold Temporary UINs**) are deleted from ITEM_UIN table.

**Batch Job Definition Name**

TemporaryUin_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-75    Key Tables for Purge Temporary UIN Detail Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| Item_uin | Yes | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Ticket Histories

Purges ticket history records based on the retention period. The retention period is specified by system configuration parameter- **Days to Hold Ticket History**.

**Batch Job Definition Name**

TicketHistory_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-76    Key Tables for Purge Ticket Histories Batch**

| Table | Select | Insert | Update | Delete |
|-------|--------|--------|--------|--------|
| config_system | Yes | | | |
| ticket_history | Yes | | | |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Purge Vendor Returns

This batch process deletes vendor returns which are in closed or completed status. Any vendor return record with a closed date/timestamp older than **Days to Hold RTV** system configuration value will be deleted. For example, the default value is 30. If the batch program is run with the default value, the batch program would delete all the vendor return records, which are more than 30 days old.

**Batch Job Definition Name**

VendorReturn_PurgeJob

**Batch Job Parameters**

<input_date>

Where input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-77    Key Tables for Purge Vendor Returns Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| config_system | Yes | | | |
| rtv_ship | Yes | | | Yes |
| rtv_ship_line_item | | | | Yes |
| rtv_ship_line_item_att | | | | Yes |
| rtv_ship_line_item_uin | | | | Yes |
| rtv | | Yes | | Yes |
| rtv_line_item | | | | Yes |
| notes | | | | Yes |

**Restart/Recovery**

This batch can be re-run by starting a new batch job after the issues are resolved.

# Data Purging Batches - BDI Staging Data

- [Purge BDI Inbound Price](#)
- [Purge BDI Item](#)
- [Purge BDI Item Loc](#)
- [Purge BDI Location](#)
- [Purge BDI Misc Group](#)
- [Purge BDI Price Hist](#)
- [Purge BDI Rfid](#)
- [Purge BDI Store Inventory Avail](#)
- [Purge BDI Supp Item](#)
- [Purge BDI UDA](#)
- [Purge BDI Warehouse Item Avail](#)

## Purge BDI Inbound Price

This job deletes BDI pricing staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiTxPriceGrp_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-78    Key Tables for Purge BDI Warehouse Item Avail**

| Tables |
| --- |
| EXT_PRICE_IN |
| PRICE_CHANGE_IN |
| CLERANCE_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

# Purge BDI Item

This job deletes BDI item hierarchy and item staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndMerchItemGrp_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-79    Key Tables for Purge BDI Item**

| Tables |
| --- |
| ITEM_HDR_IN |
| ITEM_IMAGE_IN |
| MERCH_HIER_IN |
| RELATED_ITEM_DTL_IN |
| RELATED_ITEM_IN |

**Table 5-79    (Cont.) Key Tables for Purge BDI Item**

| Tables |
| --- |
| PACKITEM_BREAKOUT_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

## Purge BDI Item Loc

This job deletes item location records from BDI inbound interface staging tables. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndItemLocGrp_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-80    Key Tables for Purge BDI Item Loc**

| Tables |
| --- |
| ITEM_LOC_IN |
| REPL_ITEM_LOC_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

## Purge BDI Location

This job deletes BDI location entities and their addresses staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndLocationGrp_PurgeJob

**Batch Job Parameters**

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-81    Key Tables for Purge BDI Location**

| Tables |
| --- |
| PARTNER_IN |
| PARTNER_ADDR_IN |
| PARTNER_ORG_UNIT_IN |
| STORE_IN |
| STORE_ADDR_IN |
| SUPS_IN |
| SUPPLIER_ADDR_IN |
| SUPS_CFA_EXT_IN |
| WH_IN |
| WH_ADDR_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

# Purge BDI Misc Group

This job deletes BDI code and UOM miscellaneous staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndMiscGrp_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-82    Key Tables for Purge BDI Misc Group**

| Tables |
| --- |
| CODE_HEAD_IN |
| CODE_DETAIL_IN |
| DELIVERY_SLOT_IN |
| UOM_CONVERSION_IN |

**Table 5-82    (Cont.) Key Tables for Purge BDI Misc Group**

| Tables |
| --- |
| UOM_CLASS_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

# Purge BDI Price Hist

This job deletes BDI price history staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndPriceHist_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-83    Key Tables for Purge BDI Price Hist**

| Tables |
| --- |
| PRICE_HIST_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

# Purge BDI Rfid

This job deletes BDI Rfid staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiTxRfid_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-84    Key Tables for Purge BDI Rfid**

| Tables |
| --- |
| EXT_RFID_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

## Purge BDI Store Inventory Avail

This job deletes BDI Available store inventory staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiTxInvAvailStore_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-85    Key Tables for Purge BDI Store Inventory Avail**

| Tables |
| --- |
| INV_AVAIL_STORE_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

## Purge BDI Supp Item

This job deletes BDI Supplier items staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndSupItemGrp_PurgeJob

**Batch Job Parameters**

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-86    Key Tables for Purge BDI Supp Item**

| Tables |
| --- |
| ITEM_SUPPLIER_IN |
| ITEM_SUP_CTY_DIM_IN |
| ITEM_SUPP_COUNTRY_IN |
| ITEM_SUP_MAN_CTY_IN |
| ITEM_SUPP_UOM_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

# Purge BDI UDA

This job deletes BDI UDA staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiFndUdaGrp_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-87    Key Tables for Purge BDI UDA**

| Tables |
| --- |
| UDA_IN |
| UDA_VALUES_IN |
| UDA_ITEM_DATE_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

## Purge BDI Warehouse Item Avail

This job deletes BDI warehouse item available inventory staging data. The retention period is default to 6 months.

**Batch Job Definition Name**

BdiTxInvAvailWh_PurgeJob

**Batch Job Parameters**

<input_date>

input_date is defaulted to current timestamp if not specified. It is used for comparing if a record date is a configured number of days prior to the input date. System batch input date format is used for parsing input date if specified.

**Key Tables**

**Table 5-88    Key Tables for Purge BDI Warehouse Item Avail**

| Tables |
| --- |
| INV_AVAIL_WH_IN |

**Restart/Recovery**

This batch can be restarted/re-run from previous failures after the issues are resolved.

# Batch Job Administration

This provides information about the processing and operating details of batch job administration and operations, and covers the following topics:

- Job Administration
- Job Scheduler

The batch jobs are installed and configured by the EICS application installer. The batch processes are designed to process large volume of data. The batch jobs can be scheduled as per the retailer's choice to be executed on specific intervals on the SIOCS GUI.

The following diagram illustrates the high-level architecture of the current batch processing implemented for the EICS Application Server.

**Figure 5-10    EICS Application Server Batch Processing Architecture**
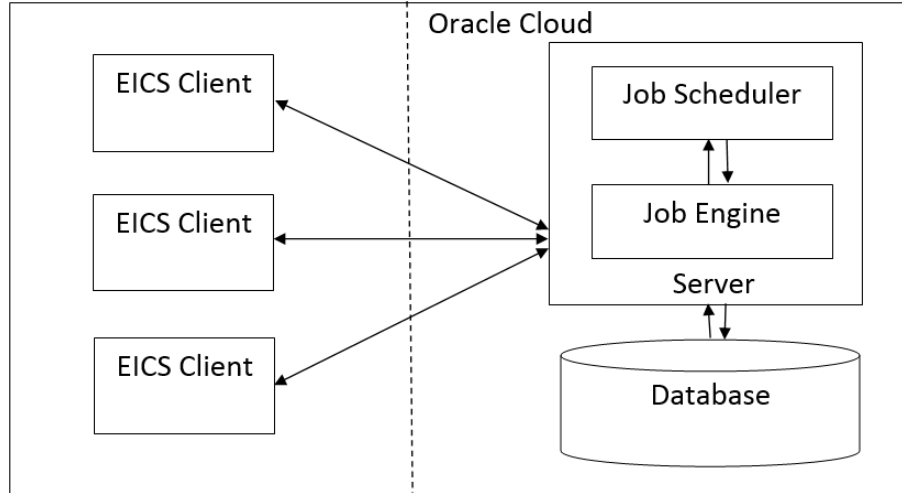


- **EICS Client - Server Communication**

  EICS client provides an option to the retailer to run the batch jobs on demand. This call to the server is made via HTTP REST service call. The batch job selected, and the parameter selected by the user are set on the request and sent to the server on this call. The server handles this request and invokes a start job on the batch engine for the respective job.

- **Job Engine**

  The job engine manages the state of a running job and guarantees the execution of each step defined for the job. The call made from the client or job scheduler is passed onto the job engine to start a new instance of the batch job. Users also have an option to stop the running job or to restart any particular job which failed during the processing.

- **Job Scheduler**

  This feature of EICS allows a retailer to schedule the batch jobs to run at a specific time interval. Each batch job will initially execute at a pre-configured schedule interval. Retailers will have an option to enable or disable specific batch job schedules as well. For more information, please refer to Job Scheduler Section below.

  > **Note:**
  >
  > A few of the jobs which are system required will not be available for the retailer to change the schedule interval or disable them. These jobs will always be enabled and can be managed only by cloud admin.

# Job Administration

SIOCS Job Admin is a web application that provides the runtime and GUI for managing batch jobs.

SIOCS provides an independent user interface for executing and scheduling of the batch jobs. These user interface screen will facilitate users to perform following operations:

- Execute Ad hoc Jobs
- Manual Data Seeding Importer Jobs
- View the Job Execution Log
- View the list of Jobs executed
- Schedule Jobs for execution on specific intervals

The following diagrams show the Batch Admin and relevant GUI components:

**Figure 5-11    Batch Admin and GUI Components**
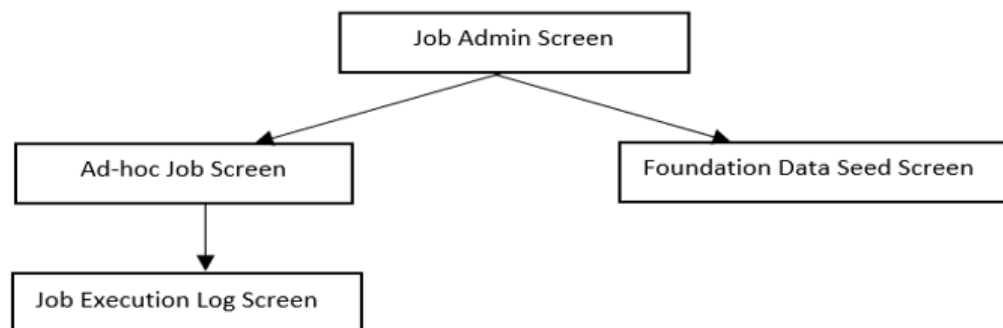


# Job Admin Screen

This screen views the list of the job that have been instantiated on the server. User can filter the list of the job loaded on screen on Job Status, Reviewed and Search Limit. The default filter will load the list of the failed job when the screen is initially loaded.

User requires appropriate permission to access this screen on SIOCS. Navigation path for the Job Admin screen is:

Admin/Technical Maintenance/Job Admin

**Figure 5-12    Job Admin Screen**



**Screen Options**

- **Search**

  Allows user to filter the list of the batch jobs instantiated by user and the scheduler.

- **Refresh**

  This option refreshes the list loaded on the screen; call is made to server to load the list of batch jobs with current state.

- **Ad hoc Job**

  This option refreshes the list loaded on the screen; call is made to server to load the list of batch jobs with current state.

- **Data Seed**

  This option navigates user to Data Seed Job Launcher screen. User can start a new data seed job from this screen by setting the required parameters and data seeding options on the job. User will require appropriate access permission to view this option on the screen.

- **Stop Job**

  This option allows user to stop a running job. This option is available only when selected job is in running state that is, STARTED and STARTING.

**Table Information**

- **Execution ID**

  This column displays the job execution identifier generated by the job engine.

- **Job Name**

  This column displays the name of the batch job.

- **Instance ID**

  This column displays the job instance identifier generated by the job engine.

- **Status**

  This column displays the current status of the batch job.

- **Start Time**

  This column displays the time when job was started.

- **End Time**

  This column displays the time when job was completed. If the job is currently in running state, the end time will not be available.

- **User**

  This column displays the user name of the user who started the job. The job started by the scheduler will have server user name in this column.

- **Reviewed**

  This column denotes if user has review this job.

- **Parameter**

  This column displays the parameter if entered while starting the batch job.

## Job Execution Log Screen

This screen displays the execution details for the batch job. Users can navigate to this screen by clicking **Job Execution ID** on the Job Admin screen. This screen is designed to view the error logs for the batch jobs and to mark the job as reviewed.

Review flag denotes the selected job has been reviewed by the user. By using this, the user can keep a track of which of the failed jobs have already been reviewed and take necessary actions as required.

The execution detail table load the entries of the execution record if exist for the select batch job.

**Figure 5-13    Job Execution Screen**
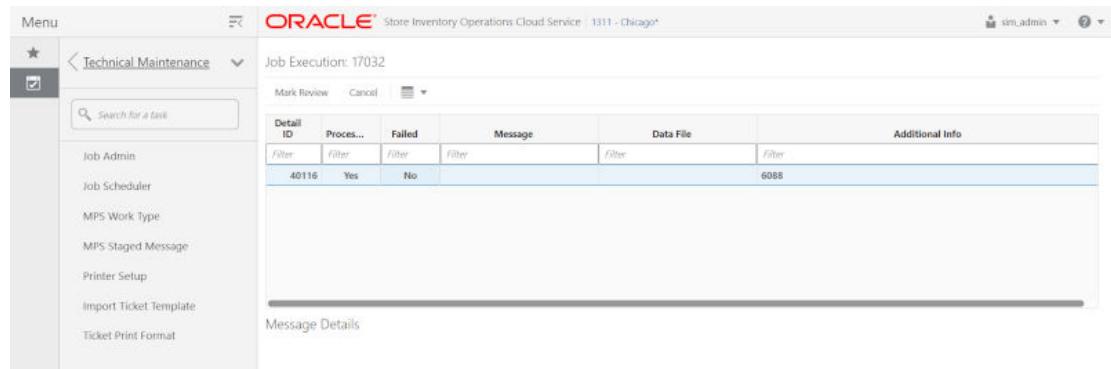


**Screen Options**

- **Mark Review**

  This option allows user to mark the batch job as review. Once the job is marked as review user is navigated back to the Job Admin screen.

- **Cancel**

  This option allows user to navigate back to the Job Admin screen.

**Table Information**

- **Detail ID**

  This column displays the execution identifier generated by the job engine.

- **Processed**

  This column denotes if the execution record for the job was successfully processed.

- **Failed**

  This column denotes if the execution records was failed during the processing.

- **Message**

  This column displays the message from the server to viewing the further details on the job status. This column will be empty for the execution records which are been successfully processed.

- **Data File**

  This column displays the file name for which the execution record was created. This column holds data only if the job details with file import processing for example, Retail Sales Audit Import Job.

- **Additional Info**

  This column holds the addition info if any for the execution record.

**Message Details**

This section in non-editable and displays the detailed explanation for the job failure. This section will display respective data on selection job execution record. The log traces of the exact point of failure shall be printed on this section of the screen.

# Job Launch Screen

This screen allow user to start a new instance for a job. Select the batch job and enter the parameter, if required, to start job.

The job launcher screen has the following categories:

- Ad hoc Job
- Data Seed Importer Job
- Start an Ad hoc Job
- View Details for Job
- Start a Data Seed Job
- Stop a Running Job

## Ad hoc Job

This screen allows user to run an ad hoc job. All the job which are either operational or purge job can be started from this screen. All the batch job will not support the execution date and store identifier parameters. The job for which store level processing can be performed will consider these parameters. User will not receive an error or confirmation if the parameters are not valid for the selected job.

User will require appropriate data permissions to select and run a particular job. Navigation to this screen is as follows:

Admin/Technical Maintenance/Job Admin /Ad hoc Job

**Figure 5-14    Ad hoc Job Screen**
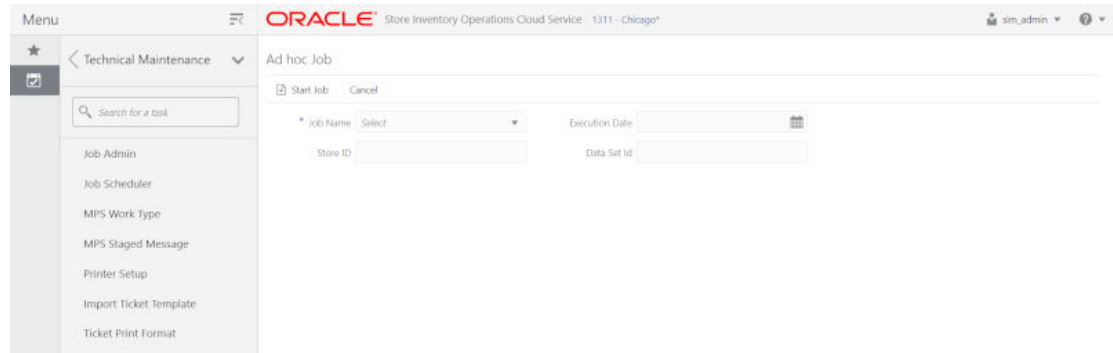


**Screen Options**

- **Start Job**

  This option allows user to start a new instance of the selected job. When the user selects this option a call is made to server to start a selected batch job. Server internally makes a call to Batch Operator specifying the Job Name and the parameter if any.

- **Cancel**

  This option navigates user back to Job Admin Screen.

**Menu Options**

- **Job Name**

  Allows user to select the job to be started. This is a required field on this screen. The user will be able to view only those jobs in the drop-down list which are permitted to the user via data permissions.

- **Store ID**

  Allow user to enter the store identifier if the job is expected to be executed for specific store. Not all job accepts store identifier as parameter. For the jobs which doesn't take store identifier as parameter for execution shall ignore this parameter.

- **Execution Date**

  Allows user to enter the execution date for the job, if not entered the job will consider the current date for the execution. Not all job accepts execution date as parameter. For the jobs which doesn't take execution date as parameter for execution shall ignore this parameter.

- **Data Set Id**

  The Data Set Id is optional parameter for importer job to process the specified data set from bulk data interface table.

## Data Seed Importer Job

This screen allows user to start data seeding importer specific jobs only. For additional details, please refer to Data Seeding.

User will need appropriate access privileges to navigate to this screen.

Navigation to this screen is as follows:

Admin /Technical Maintenance/Job Admin/Data Seed

**Figure 5-15    Data Seed Screen**
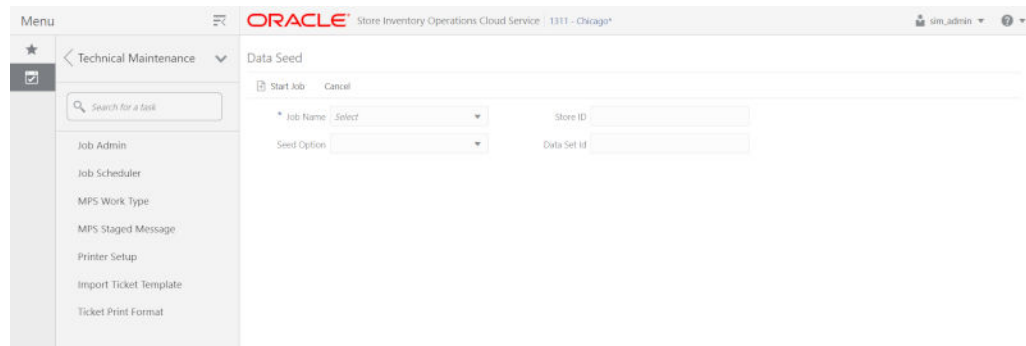


**Screen Options**

- **Start Job**

  This option allows user to start a new instance of the selected job. When the user selects this option a call is made to server to start a selected batch job. Server internally makes a call to Batch Operator specifying the Job Name and the parameter if any.

- **Cancel**

  This option navigates user back to Job Admin screen.

**Menu Options**

- **Job Name**

  Allows user to select the job to be started. This is a required field on this screen. The user will be able to view only those jobs in the drop-down list which are permitted to the user via data permissions.

- **Seed Option**

  User can select one of the option available as per the operation to be performed. The seed options are more over job specific and are not applicable for all job. If the invalid seed option is selected for the job, server will ignore it and perform a normal seed operation.

  The options available are:

  1. Seed: Seeds the data from BDI integration tables to the application tables.

  2. Delete and Seed: Deletes the data from the application tables and reseed the available data in BDI integration tables.

3. Delete Seeded: Deletes the data from the application tables.

4. Delete In: Deletes the data from BDI integration tables.

5. Disable Constraint: Disable the constraints for all the application tables.

6. Enable Constraint: Enable the constraint for the application tables.

For example, if user selects Disable Constraint Option for Store Job, this won't disable the constraint for store table. The server will simply ignore the seed option and continue this as not seed operation.

- **Store ID**

  Allows user to perform store level data seeding. The store identifier is not applicable for all the job. For the job which does not support store level data processing for example, Item Header or Supplier will simply ignore the store identifier if entered.

- **Data Set Id**

  The Data Set Id is optional parameter for importer job to process the specified data set from bulk data interface table.

## Start an Ad hoc Job

1. Login to SIOCS and navigate to **Job Admin** screen.

2. Click on **Ad hoc Job** menu to navigate to **Ad hoc Job** screen

3. Select the required **Job** from the drop-down menu.

4. Set the required job parameter for the job, that is, Store ID and/or Execution Date.

   > ✏ **Note:**
   >
   > Job Parameter doesn't apply to all available batch jobs, for the job which do not take any parameter input will simply ignore the value in case entered.
   >
   > Some of batch jobs belongs BDI TX (Transaction) category, see Appendix: BDI Transmitter CLI Procedure - Load External Data File, BDI Transaction (TX) Importer Jobs section for details.

5. Click on **Start Job** to start the new instance of job.

6. The new execution record will be populated on the **Job Admin** screen list.

## View Details for Job

1. Login to SIOCS and navigate to **Job Admin** screen.

2. Click the **Job Execution Id** to navigate to details.

3. To navigate back to **Job Admin** screen, click **Cancel**.

4. To mark job execution reviewed, click **Mark Reviewed**.

## Start a Data Seed Job

1. Login to SIOCS and navigate to **Job Admin** screen.

2. Click on **Data Seed** menu to navigate to **Data Seed** screen.

3. Select the required job populated in the drop-down menu.

4. Select required **Seed Option** from drop down.

5. Enter **Store ID** if required.

> ✎ **Note:**
>
> Store ID is not supported by all job, will be applicable only for jobs which processed store level data

## Stop a Running Job

1. Login to SIOCS and navigate to **Job Admin** screen.

2. Select the job which is currently in running state that is, STARTED or STARTING.

3. Click on **Stop Job** to stop the execution of the running job.

4. Prompted to confirm if the job needs to be stopped. Click **Yes** to continue.

5. Request is sent to job engine to stop the instance of the job. In case the job is already completed or stopped appropriate message shall be displayed to user.

6. Refresh the list on the screen to view the changes.

# Job Scheduler

EICS Job Scheduler allows user to schedule the job available to run under **Ad hoc Job** screen.

The key features of Job Scheduler are as follows:

• **Interval Based Schedule**

User can schedule the job to run on interval basis. User will be provided below specified interval to be set for each job for execution.

**Table 5-89    Interval Based Schedule**

| Interval | Detail | Execution Hour |
|----------|--------|----------------|
| 30th Minute | Job execution will be every 30th minute of the hour starting from 01:00. | 01:00, 01:30, 02:00, 02:30, 03:00, 03:00 … 23:00, 23:30, 00:00, 00:30. |
| 1 Hour | Job execution will be every 1 hour starting from 01:00. | 01:00, 02:00, 03:00, 04:00, 05:00 … 22:00, 23:00, 00:00. |
| 2 Hours | Job execution will be every 2nd hour of the day starting from 01:00. | 01:00, 03:00, 05:00, 07:00, 09:00 … 21:00, 23:00. |
| 3 Hours | Job execution will be every 3rd hour of the day starting from 01:00. | 01:00, 04:00, 07:00, 10:00, 13:00, 16:00, 19:00, 22:00. |

**Table 5-89    (Cont.) Interval Based Schedule**

| Interval | Detail | Execution Hour |
|---|---|---|
| 4 Hours | Job execution will be every 4th hour of the day starting from 01:00. | 01:00, 05:00, 09:00, 13:00, 17:00, 21:00. |
| 6 Hours | Job execution will be every 6th hour of the day starting from 01:00. | 01:00, 07:00, 13:00, 19:00. |
| 8 Hours | Job execution will be every 8th hour of the day starting from 01:00. | 01:00, 09:00, 17:00. |
| 12 Hours | Job execution will be every 12th hour of the day starting from 01:00. | 01:00, 13:00. |
| 24 Hours | Job execution will be every 24th hour of the day starting from 01:00. | 01:00. |

- **Schedule Management**

  User will have an option enabling or disabling the scheduler feature for a job at any given point of time. Once disabled the auto execution on the future scheduled interval will be stopped.

- **Scheduler Configuration**

  User will have option of configuring the scheduler related configuration.

**Table 5-90    Batch Job Schedules**

| Batch Job | Interval | Restricted |
|---|---|---|
| AdhocStockCount_PurgeJob | 24 hours | No |
| ArchiveDump_OpsJob | 24 hours | Yes |
| Audits_ArchiveJob | 24 hours | No |
| AutoReplenishCapacity_OpsJob | 24 hours | No |
| BatchLog_PurgeJob | 24 hours | No |
| BdiFndItemLocGrp_PurgeJob | 24 hours | No |
| BdiFndLocationGrp_PurgeJob | 24 hours | No |
| BdiFndMerchItemGrp_PurgeJob | 24 hours | No |
| BdiFndMiscGrp_PurgeJob | 24 hours | No |
| BdiFndPriceHist_PurgeJob | 24 hours | No |
| BdiFndSupItemGrp_PurgeJob | 24 hours | No |
| BdiFndUdaGrp_PurgeJob | 24 hours | No |
| BdiTxInvAvailStore_PurgeJob | 24 hours | No |
| BdiTxInvAvailStore_PurgeJob | 24 hours | No |
| BdiTxPriceGrp_PurgeJob | 24 hours | No |
| BdiTxRfid_PurgeJob | 24 hours | No |
| CleanupBatchDirectories_OpsJob | 24 hours | Yes |

**Table 5-90    (Cont.) Batch Job Schedules**

| Batch Job | Interval | Restricted |
|---|---|---|
| CleanupShelfReplenishment_OpsJob | 24 hours | Yes |
| ClearancePriceChange_OpsJob | 24 hours | No |
| Clearance_Tx_ImporterJob | 24 hours | Yes |
| ClosedTransfers_PurgeJob | 24 hours | No |
| CompletedUin_PurgeJob | 24 hours | No |
| DSDReceivingClosure_OpsJob | 24 hours | Yes |
| DsdAndPurchaseOrders_PurgeJob | 24 hours | No |
| ExtPrice_Tx_ImporterJob | 30 minutes | No |
| ExtPrice_Tx_ImporterJob | 30 minutes | No |
| FulfillmentOrderPickReminders_OpsJob | 24 hours | Yes |
| FulfillmentOrderPickReminders_OpsJob | 24 hours | Yes |
| FulfillmentOrders_PurgeJob | 24 hours | No |
| GenerateProblemLineStockCount_OpsJob | 24 hours | No |
| GenerateUnitAmountStockCount_OpsJob | 24 hours | No |
| GenerateUnitStockCount_OpsJob | 24 hours | No |
| InvAvailWh_Tx_ImporterJob | 30 minutes | No |
| InvalidUserRole_PurgeJob | 24 hours | Yes |
| InvalidUser_PurgeJob | 12 hours | Yes |
| InventoryAdjustment_PurgeJob | 24 hours | No |
| InventoryExtractSftpPush_OpsJob | 24 hours | No |
| InventoryExtract_OpsJob | 24 hours | No |
| ItemBasketMaintenance_OpsJob | 24 hours | No |
| ItemBasket_PurgeJob | 24 hours | No |
| ItemBasket_PurgeJob | 24 hours | No |
| ItemPriceToHistory_OpsJob | 24 hours | Yes |
| ItemPriceToHistory_OpsJob | 24 hours | No |
| ItemUinHistory_ArchiveJob | 24 hours | No |
| Item_PurgeJob | 24 hours | No |
| Lockings_PurgeJob | 24 hours | No |
| Notifications_PurgeJob | 24 hours | No |
| PosTransactionImport_OpsJob | 24 hours | No |
| PriceChangeWorksheet_PurgeJob | 24 hours | No |
| PriceChange_Tx_ImporterJob | 24 hours | Yes |
| PriceHistories_PurgeJob | 24 hours | No |
| ProductBasket_PurgeJob | 24 hours | No |
| ProductGroupScheduleCleanup_OpsJob | 24 hours | Yes |
| PromotionPriceChange_OpsJob | 24 hours | No |
| RegularPriceChange_OpsJob | 24 hours | No |
| RelatedItems_PurgeJob | 24 hours | No |

**Table 5-90    (Cont.) Batch Job Schedules**

| Batch Job | Interval | Restricted |
|---|---|---|
| ResolvedUinProblem_PurgeJob | 24 hours | No |
| RetailSalesAuditImport_OpsJob | 24 hours | No |
| ReturnNotAfterDateAlert_OpsJob | 24 hours | Yes |
| RfidHistory_ArchiveJob | 24 hours | No |
| Rfid_PurgeJob | 24 hours | No |
| SalesPosting_PurgeJob | 24 hours | No |
| SftpLogFile_PurgeJob | 24 hours | No |
| ShelfAdjustments_PurgeJob | 24 hours | No |
| ShelfReplenishments_PurgeJob | 24 hours | No |
| StagedMessage_PurgeJob | 24 hours | No |
| StockCountAuthorizeRecovery_OpsJob | 24 hours | No |
| StockCountExportSftpPush_OpsJob | 24 hours | No |
| StockCountUnitAndAmountSnapshot_OpsJob | 24 hours | No |
| StockCounts_PurgeJob | 24 hours | No |
| StoreItemStockHistory_ArchiveJob | 24 hours | No |
| StoreSequenceImport_OpsJob | 24 hours | No |
| TemporaryUin_PurgeJob | 24 hours | No |
| ThirdPartyStockCountImport_OpsJob | 24 hours | No |
| TicketHistory_PurgeJob | 24 hours | No |
| TransferClose_OpsJob | 24 hours | Yes |
| TransferDeliveryAutoReceive_OpsJob | 24 hours | Yes |
| TransferDeliveryClose_OpsJob | 24 hours | Yes |
| TransferNotAfterDateAlert_OpsJob | 24 hours | Yes |
| TransfersOverdueBatch_OpsJob | 24 hours | Yes |
| VendorReturnClosure_OpsJob | 24 hours | Yes |
| VendorReturn_PurgeJob | 24 hours | No |
| WarehouseItemStockImport_OpsJob | 24 hours | No |

ORACLE®

> **✎ Note:**
>
> Batch job schedules which are marked as restricted are scheduled by default to execute in 24-hour intervals as defined in #GUID-ACF1E483-550A-4565-A696-23EDBE96BA5F/CIHBAIFB. Most of these batches are keeping the system healthy and accurate by execution maintenance task. These are not visible to the Retailers' Admin Users and are disabled from being scheduled.
>
> However, they can be executed on an adhoc basis from the **Adhoc Job** screen for testing purpose or to correct a malfunction.
>
> Please use caution while running adhoc jobs since they can interfere with scheduled batch runs on same data set leading to records getting locked for one of the batches. So please run ad-hoc jobs when it is safe to do so by either waiting for same batch to complete its scheduled run OR making sure ad-hoc job can get completed before next scheduled job run for that batch.

## Job Scheduler Screen
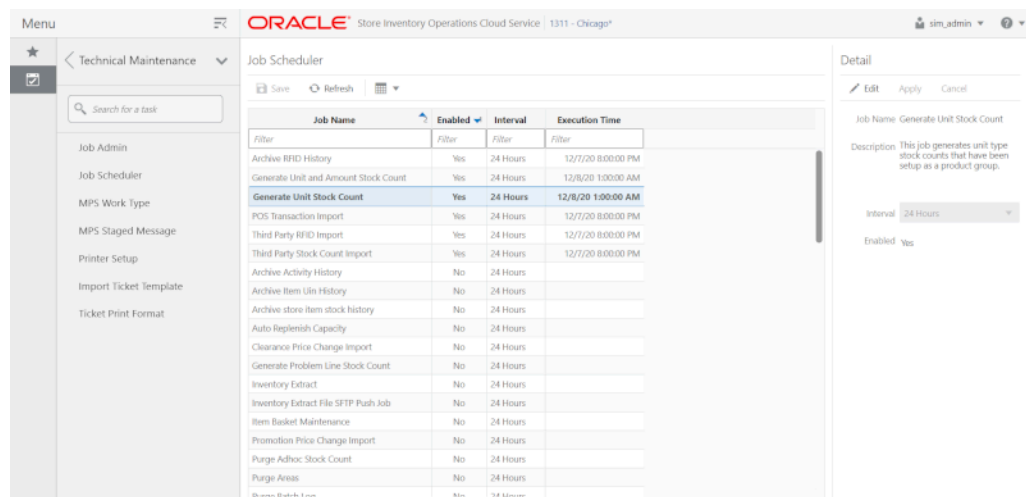
This screen allows user to manage the schedules for the batch jobs. The list of job available to be scheduled will be sort by the enabled flag followed by Job Name alphabetically that is, the jobs for which the scheduling is enable will be on the top of the list sorted by the Job Names in alphabetical order.

User can edit the batch job and perform following operations:

1. Enable / Disable a schedule.

2. Setting the execution interval for a batch job.

User will require appropriate authorization to access this screen. Retailer will not be able to view or disable the batch jobs which are system required. This batch jobs will be accessible to Cloud Admin user will have a predefined schedule. Navigation to this screen is as follows: Admin/Technical Maintenance/Job Scheduler

**Figure 5-16    Job Scheduler Screen**

**Screen Options**

- **Save**

  This option allows user to save the changes made to the job schedules. User can make changes to multiple schedules in edit mode and apply the changes, however the changes will take affect only when the save operation is performed.

- **Refresh**

  This option refreshes the list of values on the screen.

- **Edit**

  This option will be enabled only when a schedule is selected to be edited and user have privileges to edit the job schedule. This option takes user to the edit mode in which user can change the execution interval for the job and enable or disable the schedule.

- **Apply**

  This option will only apply the changes which are been made during edit mode. This option does not save the changes, it only applies the changes for the job and exit the edit mode. Click the Save option to save the changes.

- **Cancel**

  This option is enabled only when user is in edit mode. This option allows user to discard any changes made to the schedule on edit mode or to exit the edit mode.

**Menu Options**

- **Interval**

  User have option to select the appropriate interval for the batch job execution. Depending on the selected interval, server calculates the execution time for the batch job and runs it according. The base time for calculating the execution time on basis of the interval is 01:00 AM. For the store-based job this is as per the store time zone and for system jobs this is as per the server time zone.

  Option for selecting the execution interval are as follows:

  1. 30 Minutes - Runs every 30 minutes.
  2. 1 Hour - Runs every hour.
  3. 2 Hours - Runs every second hour of the day.
  4. 3 Hours - Runs every third hour of the day.
  5. 4 Hours - Runs every fourth hour of the day.
  6. 6 Hours - Runs every sixth hour of the day.
  7. 8 Hours - Runs every eighth hour of the day.
  8. 12 Hours - Expected to be executed twice a day.
  9. 24 Hours - Expected to be executed only once a day.

- **Enabled**

**Table Information**

- **Job Name**

  This column displays the name of the job.

- **Enabled**

  This column denotes whether a job is current scheduled.

- **Interval**

  This column denotes the interval of the job execution.

- **Execution Time**

  This column denotes the next execution time for the batch job. This column will not contain data for the jobs which are been disabled.

## Configuring a Job Schedule

1. Login in SIOCS and navigate to the **Job Scheduler** screen.

2. Select the desired job from the list.

> **Note:**
>
> If the Edit option is not available in the menu section, the user is not authorized to edit the batch job. Kindly assign appropriate user group or contact system admin for desired changes.
>
> Some of batch jobs belongs BDI TX (Transaction) category, see Appendix: BDI Transmitter CLI Procedure - Load External Data File, BDI Transaction (TX) Importer Jobs section for details.

3. Edit the Job details on right panel.

4. Set required interval for execution and **Enabled** to **Yes**.

5. Click **Apply** to exit the edit mode.

6. For configuring multiple jobs on a go, repeat Step 2 for each job.

7. Click **Save** to save the current changes made on the screen.

## Disabling a Job Schedule

1. Login in SIOCS and navigate to Job Scheduler screen.

2. Select the job from the list which needs to be disabled.

3. Edit the Job details on right panel.

4. Set **Enabled** as **No**.

5. Click **Apply** to exit the edit mode.

6. For disabling multiple jobs on a go, repeat Step 2 for each job.

7. Click **Save** to save the current changes made on the screen.

# 6

# Batch - Data Archive

SIOCS is an inventory transaction system and not a data warehouse.

Large volume of data can cause performance issues for transnational products. It could also cause other operational challenges such as backups could take longer time and storage could go up.

To reduce overall size of the operational tables and improve performance, for selected history tables, SIOCS archives data into archive tables. These archive tables can be purged at later point.

## Archive Data Batches

**Figure 6-1    Archive Data Batches**
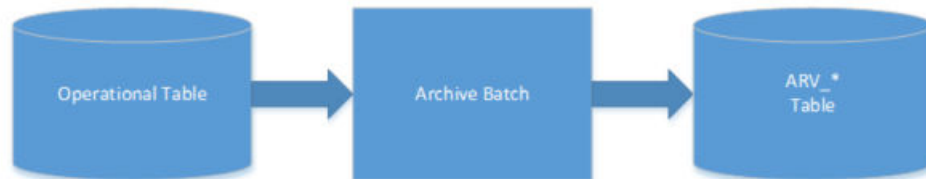


**Table 6-1    Archive Data Batch List**

| Batch Job | Description |
| --- | --- |
| Archive RFID History | Archive RFID history records into archive table. See batch chapter Archive RFID History for batch details. |
| Archive Activity | Archive activity data into activity archive tables. See batch chapter Archive Activity for batch details. |
| Archive Item UIN History | Archive item UIN data into item UIN archive tables. See batch chapter Archive Item UIN History for batch details. |
| Archive Store Item Stock History | Archive store item stock history data into store item stock history archive tables. See batch chapter Archive Store Item Stock History for batch details. |

## Archive Dump Batches

The archive dump process creates the data dump files and truncates the archive tables.

The dump files are created using oracle Data Pump API, these files are written in a proprietary, binary format which are made up of one or more disk files that contain table data.

**Figure 6-2    Archive Dump Batches**
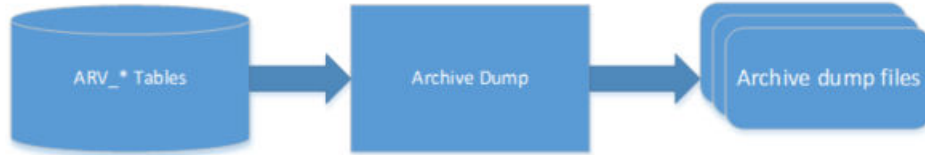


**Table 6-2    Archive Dump Batch List**

| Batch Job | Description |
|-----------|-------------|
| Archive Dump | Export archive history data into Dump files and truncate archive history tables. |
| | See Batches chapter: Archive Dump for batch details. |

# Archive Batch Scheduling Consideration

Archive Data Batches need to be run prior to Archive Dump Batches.

The Archive Dump Batch Job need to be run after Archive Data batches.

# Archive Data Dump Files Export

The archived history data dump files can be collected from database directory and exported to retailer per customer's request, Oracle AMS team will need to raise ticket to cloud DBA team to collect the data dump files. Once the data dump files are exported, it is up to retailer to take the files. EICS will no longer keep the data dump files after files are exported to customer.

If customer does not request to be exported to customers, these data dump files will be kept in database archive locations up to time determined by Oracle cloud service agreement.

> **✎ Note:**
>
> The data dump file location in database server can be queried using following:
>
> Login to SIOCS PDB: SELECT DIRECTORY_PATH FROM DBA_DIRECTORIES WHERE DIRECTORY_NAME = 'SIOCS_DATA_PUMP_DIR';

# 7

# Technical Maintenance Screens

This section covers the following topics:

- Barcode Processor
- Credential Administration
- External Printer Setup
- MPS Staged Message
- MPS Work Type
- POS Transaction Resolution
- Sequence Administration

## Barcode Processor

Barcode Processors are used within EICS to scan items. When the application takes a barcode (or manual entry of one), it passes it along to the barcode processors to extra the correct sku number from the barcode in order to retrieve the item. This form allows the control of which barcode processors the application uses and what order they are used in. The barcode processor form is located under Configuration Barcode Processor.

**Figure 7-1    Barcode Processor**
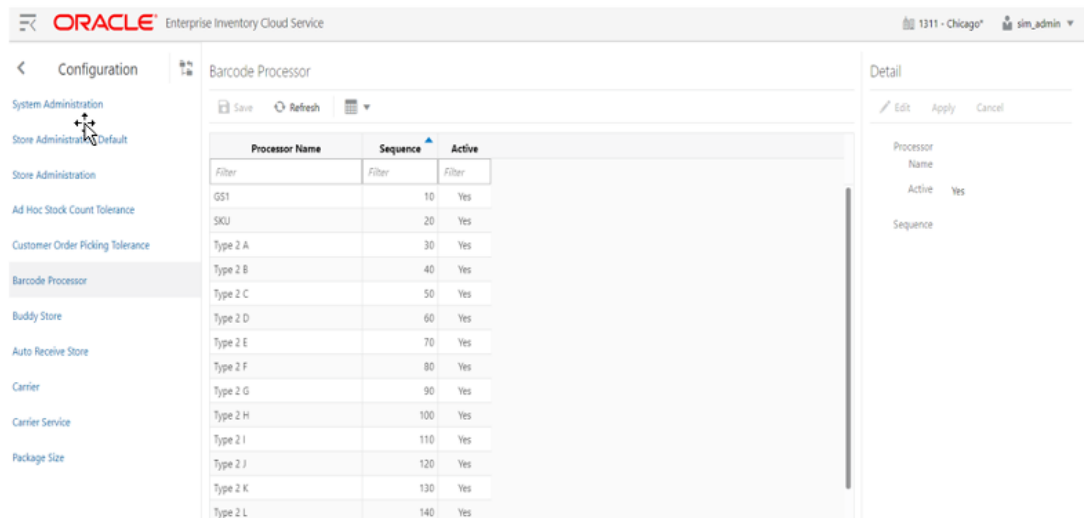


The above screen picture shows the list and detail panel in display mode.

**List Buttons**

- **Save**: Persists currently altered information.
- **Refresh**: Refreshes the screen with currently persisted information.
- **Filter**: Enable or disable the filtering row in the table.

**Barcode Processor Attributes**

- **Processor Name**: The name of the processor. This does not alter how the processor works. For example, Type 2 E will continue to parse Type 2 E, even if it is renamed to Type 2 K.
- **Sequence**: Sequence indicators the order the barcode processors will attempt to analyze the item scanned. It starts with the lowest sequence number and proceeds upward. Items will be returned by the first barcode processor that finds a valid item within the barcode.
- **Active**: Yes, indicates that the barcode processor is active and will be used when scanning items.

**Detail Panel**

**Figure 7-2    Barcode Processor Detail Panel in Edit Mode**
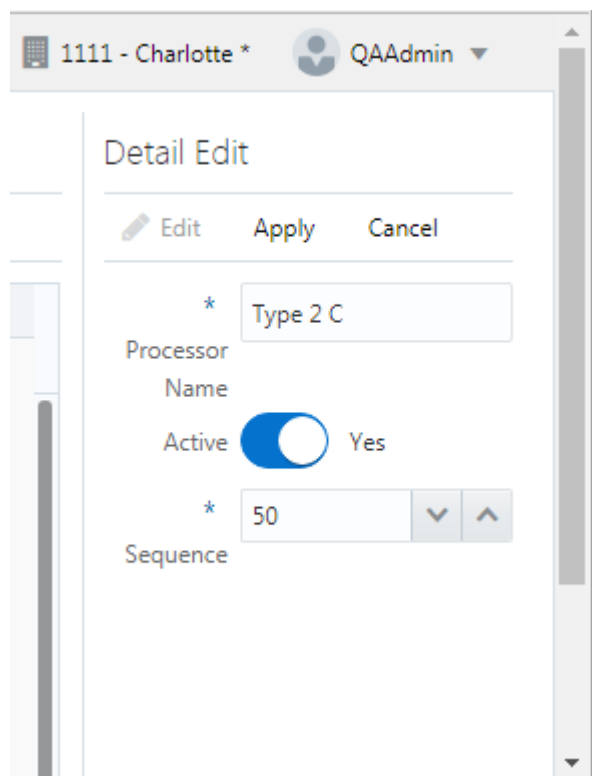


**Detail Buttons**

- **Edit**: Places the detail information displayed into edit mode.

- **Apply**: Applies the current data to the barcode processor and updates the list information.
- **Cancel**: Places the panel back into display mode without applying the information.

## Barcode Processors

**Table 7-1    Barcode Processors**

| Barcode Processor | Description |
| --- | --- |
| SKU | Parses the data scanned as a direct sku number or Universal Identification Number. |
| GS1 | Parses the data scanned as a GS1 (Global Standards One) seeking out item number within. |
| Type 2 A-L | Barcode processor parses the data as a Universal Product Code (Type 2), otherwise known as random weight barcodes. There is a whole series of available code patterns from sub-type A to sub-type L. |
| UPC E | Parses the data scanned as a Universal Product Code (Type E), a six-digit barcode. |
| VPN | Parses the data scanned is a Vendor Product Number. |
| RFID | Parses the data as Radio Frequency Identification (RFID) tag. |
| RFID GTIN-12 | Parses the data as Radio Frequency Identification (RFID) GTIN-12 tag. |
| RFID GTIN-13 | Parses the data as Radio Frequency Identification (RFID) GTIN-13 tag. |
| ISN | Parses the data as Item Scan Number (ISN) seeking out the IMEI number. |

> **Note:**
>
> Any changes to the Barcode Processor configuration require EICS server restart. You can request a server bounce by contacting Oracle Support via a Service Request on My Oracle Support.

## Credential Administration

The Credential Administration screen is used to setup integration credentials to connect with external/third-party systems.

To access this screen, user need to be assigned the following role/permission:

- **IDCS or OCI IAM app roles**: <SIOCS Primary APP>.admin_users
- **Security Permission**: Access Credential Administration

**Figure 7-3    Credential Administration**



**List Buttons**

- **Save**: Persists currently altered information.

- **Refresh**: Refreshes the screen with currently persisted information.

- **Delete Selected**: Delete the user and password value for the records selected from database.

- **Filter**: Activates the filter dialog allowing the user to filter information.

**List Attributes**

- **Alias**: The alias for the user.

- **Description**: The description of the user.

- **User**: The user name.

- **Password**: Displays whether a password has been set currently for the user.

- **Update Date**: The last update date.

- **Update User**: The user who updated the record last.

**Detail Panel**

**Figure 7-4    Credential Administration Detail Panel in Edit Mode**



**Detail Buttons**

- **Edit**: Places the detail information displayed into edit mode.
- **Apply**: Applies the current data to the credential admin and updates the list information.
- **Cancel**: Places the panel back into display mode without applying the information.

**Detail Attributes**

- **Alias**: Shows the alias for the user. Pre-defined read only field.
- **Description**: Displays the user description. Pre-defined read only field.
- **User**: Field to enter user name.
- **Password**: Field to enter password for the user.
- **Confirm Password**: Field to enter password one more time in order to make sure the user has entered the right password.

# External Printer Setup

The Printer's screen is used to configure printers. The user can add, view, edit, and delete a printer as well as assign a network address for each printer. They can also indicate whether the printer is the default printer for manifest printing or pre-shipment printing.

**Figure 7-5    Print Setup Navigation and List Screen**
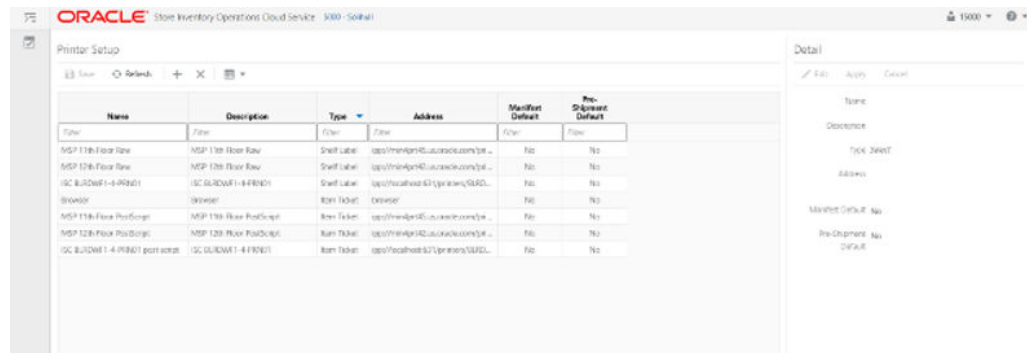


**List Buttons**

- **Save**: Persists currently altered information.

- **Refresh**: Refreshes the screen with currently persisted information.

- **+**: Add a new printer to configuration. It clears out the detail panel so the user can enter new printer information.

- **x**: Remove a printer from configuration. A dialog confirmation warning will occur prior to deletion of the record.

- **Filter**: Activates the filter dialog allowing the user to filter information.

**Printer Attributes**

- **Name**: The name of the printer.

- **Description**: Longer name for the printer.

- **Type:** The type of the printer. Valid values are: Item Ticket, Shelf Label and Postscript.

- **Address**: The address of the printer.

- **Manifest Default**: Yes, if this is the default printer for manifest printing, otherwise No.

- **Pre-Shipment Default**: Yes, if this is the default printer for pre-shipment printing, otherwise No.

**Detail Panel**

**Figure 7-6    Printer Setup Detail Panel In Edit Mode**



**Detail Buttons**

- **Edit**: Places the detail information displayed into edit mode.
- **Apply**: Applies the current data to the printer and updates the list information.
- **Cancel**: Places the panel back into display mode without applying the information.

# MPS Staged Message

The MPS Staged Message administration form is used to monitor the messages in the Message Processing System queue. Each message contains the detailed information being transmitted between EICS and an external system.

**Figure 7-7    MPS Staged Message**



**Filter Bar**

- **Search**: This is a button that displays a filtering dialog.

- **Description**: A non-labeled text field follows the Filter button that displays what criteria are currently selected to filter the staged message table.

**Filter Dialog Attributes**

- **Family**: Limits the search results to only those messages in the specified family.

- **In/Out**: Indicates if the message is inbound to EICS or outbound to an external system.

- **Search Limit**: Limits the number of messages found to the specified limit or less.

- **Show Pending**: If true, only messages that are pending will be displayed.

- **Show Retry**: If true, only messages that are in retry will be displayed.

**Figure 7-8    MPS Staged Message Filter**
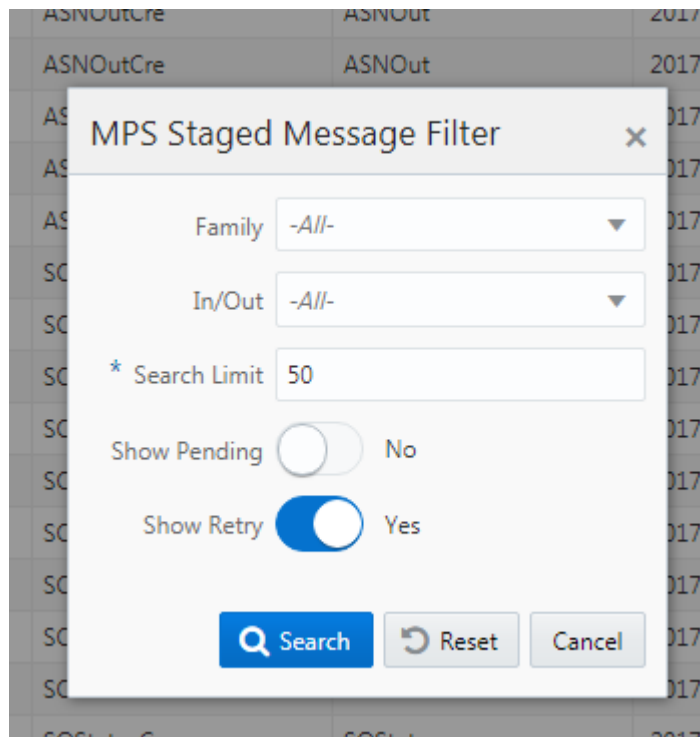


**Filter Dialog Buttons**

- **Search**: Executes a search for staged messages based on the selected criteria and refreshes the list table with that information.

- **Reset**: Results the filter dialog back to its default settings.

- **Cancel**: Closes the dialog without taking any action.

**List Buttons**

- **Refresh**: Refreshes the staged message list with current information.

- **Retry**: If a staged message is selected, selecting this button will retry the processing of the staged message.

- **Delete Selected**: If a stage message is selected, selecting this button will attempt to delete the message.

- **Filter**: Activates the filter dialog allowing the user to filter information.

**List Attributes**

- **Record Id**: A unique identifier assigned to the stage message.

- **In/Out**: Indicates if the message is inbound to EICS or outbound to an external system.

- **Type**: Type of message within a family. For example, most families of messages have a create, modify, and delete type of message.

- **Family**: Describes the family that the message belongs to. This will align with a MPS Work Type scheduled to process this family of messages.

- **Create Time**: The timestamp of the first time the message was created.

- **Update Time**: The timestamp of the last time the message was updated.

- **Execution Count**: The number of times the system has attempted to process the message.

- **Business ID**: A business identifier associated to the message. More than one message may carry a business identifier, and so this can be used to associate messages that may be related to the same activity.

- **Store ID**: The identifier of the store associated to the message.

- **Job ID**: The job identifier is a server-generated sequence number used for grouping related messages. If a message is singular with no other related messages, then its record identifier and job identifier will be identical. For messages, that must be executed sequentially together as a group, they will be assigned the same job identifier but different record identifiers.

- **Description**: A brief formatted description of the staged message that gives some indication of the contents within.

# MPS Work Type

The MPS Work Type administration form is used to configure the MPS work types. MPS stands for Message Processing System and each work type represents an external inbound or outbound message family or grouping to be delivered to another system. These work types do not represent the messages themselves (see MPS Staged Message) but the working queue that handles the processing of these external messages.

**Figure 7-9    MPS Work Type List**



**List Buttons**

- **Save**: Persists currently altered information.

- **Refresh**: Refreshes the screen with currently persisted information.

- **Filter**: Activates the filter dialog allowing the user to filter information.

**List Attributes**

- **Work Type**: The work type is the name of the message queue being worked on.

- **Direction**: Indicates if the message queues is inbound to EICS or outbound to an external system.

- **Active**: Yes, indicates the work type is currently active and attempting to process messages. No indicates it has been disabled.

- **Retry Limit**: The number of times to attempt to process a single message before marking it as failed.

- **Pending Count**: Number of messages pending processing.

- **Retry Count**: Number of times the system as attempted to process the message. Zero times means processing has not been attempted yet.

- **Fail Count**: Number of messages that have failed to be processed.

- **Last Update**: The timestamp of the last time a staged message record was updated in the database for this specific work type.

- **Last New**: The timestamp of the last time a staged message record was created in the database for this specific work type.

- **Retry Delay Secs**: The delay in seconds between retries.

- **Retry Delay Max Secs**: The maximum delay in seconds between retries.

- **Retry Delay Random**: The factor used to limit the range of retry delay randomization.

- **Purge Processed**: Indicates if automatic purging of messages that are processed successfully is enabled.

**Detail Panel**

**Figure 7-10    MPS Work Type Detail Panel In Edit Mode**



**Detail Buttons**

- **Edit**: Places the detail information displayed into edit mode.

- **Apply**: Applies the current data to the work type and updates the list information.

- **Cancel**: Places the panel back into display mode without applying the information.

**Detail Attributes**

- **Retry Limit**: The number of times to attempt to process a single message before marking it as failed. Zero indicates it should only be attempted once and will not be retried. A positive value indicates how many attempts to process the message should be made in addition to the original attempt.

- **Retry Delay Seconds**: The number of seconds between attempts to process a message. Zero indicates no delay whereas a value of 30 indicates 30 seconds between attempting to process the message. This value does not represent an absolute value as it used with other parameters below to produce a calculated message time delay.

- **Retry Delay Max Seconds**: This is the maximum number of seconds for a delay between processing messages. This caps the maximum value for the calculated delay.

- **Retry Delay Factor**: This attribute produces an increased delay between each retry of a failed message in the queue. It is a decimal value starting at 1.0 and increasing. If you enter 1.0, it means there will be no increase in the retry delay seconds during repeated attempts to process a message. A value of 1.5 indicates that the retry delay seconds will be 150% of the retry delay seconds on retry.

    A linear delay of 1.0 can be used but may result in messages reaching their retry limit prior to any issues being resolved. A value of 1.5 or larger will produce an increased delay that may allow time for other dependent messages that could be holding up execution to arrive.

- **Retry Delay Random**: Defines the amount of the delay to increase or decrease by a random amount. It accepts values from 0.0 to 1.0. A value of 0.0 disables random variation of the delay. A value of 0.5 indicates the calculated delay may be randomly altered by up to 50% of its value whereas 1.0 indicates it could be randomly altered up to 100% of its value. It is recommended to use at least some small random variation to improve message throughout as this will reduce resource convention and help avoid timing issues.

- **Notes**: The parameters are applied to retries in the following manner. The retry delay seconds is first increased by the retry delay factor, then the retry delay max seconds is applied, and finally the random delay is added or subtracted to determine the final number of seconds to wait between message processing attempts.

- **Purge Processed**: Switch to enable/disable automatic purge of messages that are processed successfully.

- **Active**: Switch to enable/disable a work type.

# POS Transaction Resolution

The POS Transaction Resolution dialog is used to view the list of Troubled POS transactions that are not processed in the system due to various data issues in the inbound data. The interface allows the user to edit the unrestricted data fields to rectify the data and let the system to process the transaction and resolve.

This dialog is mostly used by technical members to resolve the POS transactions that are in trouble.

Based on the nature of the field and to maintain the integrity of the data, the system allows the user to modify certain fields and update the transactions.

**Figure 7-11    POS Transaction Resolution List**



The POS Transaction Resolution List screen appears when the user selects the POS Transaction Resolution menu from the Admin/Technical Maintenance.

When the user enters this screen, the system set the search criteria to current date and displays the current day failed transactions that can be resolved.

The user can select the search option to refine the search by providing the search criteria.

The retry option allows the user to retry processing a selected transaction. The user must select a row and the system will process the complete transaction of the selected block id.

Selecting the link on the item will take the user to the item detail screen for that item.

On this screen, the list shows the details of the transaction for the user to know and act.

**Figure 7-12    POS Transaction Resolution List Search Criteria**



This search criteria screen can be used to filter the records based on the given criteria.

By default, the system sets the date range for the current day and allows the user to set various criteria to search the failed POS transactions. The data on the list screen is shown based on the criteria provided.

**Figure 7-13    POS Transaction Resolution Detail**

On the POS Transaction Resolution Detail screen, the system displays the transaction attributes of the selected transaction and allows the user to edit certain values and update the transaction.

Once the transaction is updated, the system will automatically retry and process it successfully if the system finds the transaction without any error.

The user must enter all the required fields on this screen.

# Sequence Administration

The Sequence Administration screen is intended to be used for initial implementation between integration systems to split the sequence ranges between integrated applications.

If sequence reaches maximum value, the retailer would need to make decision if need to cycle the sequence and raise service request to have oracle DBA to alter the sequence.

To access this screen, user need to be assigned the following role/permission:

- **IDCS or OCI IAM app roles**: <SIOCS Primary APP>.admin_users

- **Security Permission**: Access Sequence Admin

**Figure 7-14    Sequence Administration**



**List Buttons**

- **Save**: Persists currently altered information.

- **Refresh**: Refreshes the screen with currently persisted information.

**List Attributes**

- **Description**: Description of the Sequence.

- **Current Sequence**: The last currently used number written to Database Disk (Including the sequence caching size).

- **Cache Size**: The number of sequences kept in DB cache.

- **Minimum Value**: The minimum value the sequence can become.
- **Maximum Value**: The maximum value the sequence can become.
- **Modified Start**: Retailer specified start number for the sequence.
- **Modified End**: Retailer specified end number for the sequence.
- **Update Date**: The last update date.
- **Update User**: The last update user.

**Detail Panel**

**Figure 7-15    Sequence Administration Detail Panel In Edit Mode**

Detail

✏ Edit      Apply      Cancel

Description  DSD

Current Sequence  287

Cache Size  20

Minimum Value  1

Maximum Value  999999999999

Modified Start  287

Modified End  999,999,999,999

**Detail Buttons**

- **Edit**: Places the detail information displayed into edit mode.
- **Apply**: Applies the current data to the work type and updates the list information.
- **Cancel**: Places the panel back into display mode without applying the information.

**Detail Attributes**

- **Description**: Description of the sequence.
- **Current Sequence**: The last currently used number written to Database Disk (Including the sequence caching size).
- **Cache Size**: The number of sequences kept in DB cache.
- **Minimum Value**: The minimum value the sequence can become.
- **Maximum Value**: The maximum value the sequence can become.

- **Modified Start**: Retailer specified start number for the sequence.
- **Modified End**: Retailer specified end number for the sequence.

# 8

# Integration

This section describes the integration through RIB, Batches and web services.

- Overall Integration Touchpoints
- Retail Integration Cloud Service (RICS) - based Integration
- Batch Integration
- Web Services
- Sales Integration
- Integration with Manifesting Systems
- Integration for Ticket Printing
- Integration for Sales Forecast
- Integration for Store Order
- Retail Home Integration

# Overall Integration Touchpoints

**Figure 8-1    Overall Integration Touchpoints**
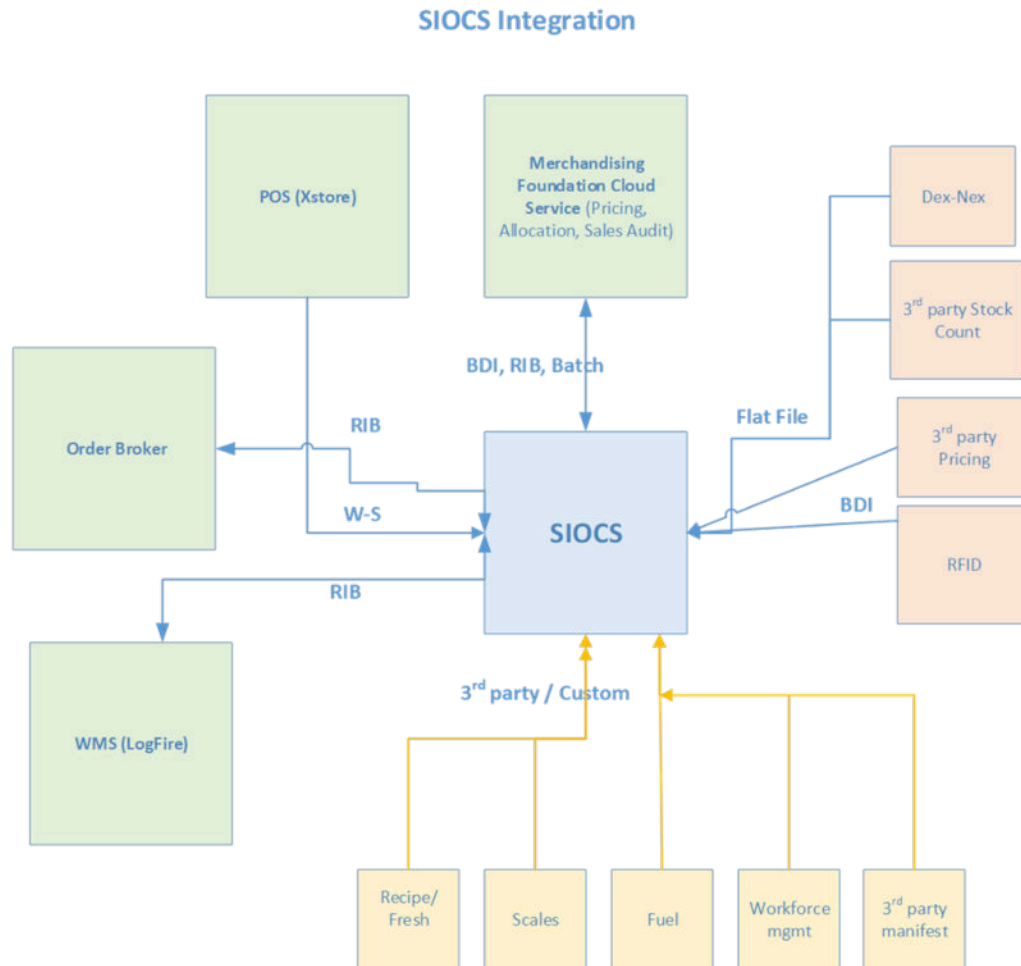


# Retail Integration Cloud Service (RICS) - based Integration

RICS is a platform containing RIB and BDI.

- Customer Orders

- Picking

- Deliveries

- Reverse Picking

- Multi Leg

- RIB Payloads

- Purchase Orders and Vendor Deliveries

- Inventory Adjustments
- Items
- Stock Counts
- Transfers
- Transfer Creation
- Transfer Messages
- Transfer Shipment Creation
- Transfer Receiving
- Transfer Doc
- Transfer Shipment
- Transfer Receiving
- Vendor Return

# Customer Orders

- Customer Order Create is used for Customer Orders that are a type of Web Order integrated through a message (FulfilOrdDesc). These integrations are used for the customer order from the Order Management System (OMS).
- The Customer Order Create failure message (FulfilOrdCfmDesc) is a message that will be sent out to external system when we get a Customer Order that comes into the system through the RIB and fails due to validation issues such as an invalid item. The purpose of the create failure is so other systems will know it has failed when it came in and that it is not being processed.
- The Stock Order Status message (SOStatusDesc) will be sent out with an SI upon reserving inventory for the customer order.

# Picking

- A Stock Order Status message (SOStatusDesc) is sent out with a type of SI upon reserving inventory. This happens when more is picked than what was on the order due to tolerances. This could also occur when a substitute item is added during the picking process.
- The Stock Order Status message (SOStatusDesc) with a type of SD will be published to un-reserve the original items inventory when a substitute item has been added during picking.
- A Stock Order status message (SOStatusDesc) is sent out with a type of PP when picking is completed.
- Item Substitutes are sent to EICS from the merchandising system through the item message (ItemDesc).

# Deliveries

- An ASN Out message (ASNOutDesc) is sent out upon dispatching of the Delivery. This will be done for pick-ups and for shipments.

- The Stock Order Status message (SOStatusDesc) with a type of PP will be published for the pick quantity in the scenario that more was delivered than what was picked.

- The Stock Order Status message (SOStatusDesc) with a type of SI will be published for the reserved quantity. This will occur when more was delivered than what was reserved. This can happen when picking was not required, the reservation occurs upon receipt of a delivery, and the full amount had not been received, therefore not reserved.

## Reverse Picking

- Customer Order Cancellations (FulfilOrdRef) will come into EICS from external system such as an OMS through the RIB. This service will perform all the validations to determine if it should create a reverse pick and whether that reverse pick should be auto completed.

- Customer Order Cancellation Confirmation (FulfilOrdRef) is a message to send to OMS upon completing of the system-generated reverse pick.

- Stock Order Status message (SOStatusDesc) with a type of SD will be published for the reserved quantity to un-reserve the inventory for the reverse pick for system-generated picks.

- Stock Order Status message (SOStatusDesc) with a type of PU will be published for the reverse picked quantity to un-pick the inventory for system-generated picks.

## Multi Leg

The following integrations are in addition to the standard integrations that already exist such as receipt message, and so on:

- The Stock Order Status message (SOStatusDesc) with a type of SI will be published for the reserved quantity.

- The Stock Order Status message (SOStatusDesc) with type of PP will be published for the picked quantity.

## RIB Payloads

RIB payloads are used to communicate information to external systems through RIB Integration.

| RIB Payload | Description |
| --- | --- |
| FulfilOrdDesc | RIB payload that contains information about a new web order type of fulfillment order to be created in. |
| FulfilOrdCfmDesc | RIB payload sent from EICS that contains fulfillment order information when that order creating in EICS failed |
| FulfilOrdRef | RIB payload that contains information about a fulfillment order cancelation. It is sent to EICS to convey a cancelation request and sent from EICS to convey actual cancellations. |
| SOStatusDesc | Sent from EICS to convey changes in item status for a specific fulfillment order. Such changes of status include (un)reservation and (un)picking. |

| RIB Payload | Description |
| --- | --- |
| ASNOutDesc | Sent from EICS to convey a delivery for specified fulfillment order. |

# Purchase Orders and Vendor Deliveries

MERCHANDISING publishes the Purchase Orders created for the direct store deliveries using RIB messages. EICS subscribes to these messages and stores them in the EICS database to enable receipt against Purchase Orders.

MERCHANDISING publishes the unit cost of the item at the item/supplier/country level for EICS to use in the receiving process.

EICS publishes the receipts done against the Purchase Order to the merchandising system (Receiving message).

EICS publishes the DSD receipts created in EICS without a Purchase Order to the merchandising system (DSDReceipts and DSD Deals messages).

EICS publishes the receiver unit adjustment done for the deliveries that are already confirmed (receiving message).

EICS is also capable of subscribing to the vendor EDI ASNs through RIB using the ASN In message format.

RIB payloads are used to communicate information from EICS to external systems and from external system to EICS through RIB Integration.

| RIB Payload (Subscriber) | Description |
| --- | --- |
| PORef | RIB payload that contains reference level information of a purchase order.This payload is used for removal of purchase orders. |
| PODesc | RIB payload that contains detailed information of a purchase order.This payload is used for creation and modification of purchase orders. |
| ASNInRef | RIB payload that contains reference level information of an ASN. This payload is used for removal of an ASN. |
| ASNInDesc | RIB payload that contains detailed information about the ASN. This payload is used for creation of a direct delivery (document type= 'P') or a warehouse delivery (document type= 'D'). |
| | EICS consumes this payload from warehouse when source and/or destination for ASN is a warehouse system. |

| RIB Payload | Description |
| --- | --- |
| ReceiptDesc | RIB payload that contains detailed information of the direct delivery receipt. This is published when the purchase order is not null. |
| | EICS also consumes this payload for warehouse receiving. |
| DSDReceiptDesc | RIB payload that contains detailed information of the direct delivery receipt. This is published when the purchase order is null. |

| RIB Payload | Description |
|---|---|
| SOStatusDesc | RIB payload sent from EICS to convey changes in item status for a specific fulfillment order. |
| | EICS also consumes this payload from warehouse for stock movements originating at the warehouse. |
| InvAdjustDesc | RIB payload that contains information about destination of the adjustment and an InvAdjustDtl. |

## Inventory Adjustments

Inventory adjustments integrate to MERCHANDISING at the item level using the RIB. EICS creates the adjustments and groups them together by a header with multiple items, but for integration purposes they are published out at an item level.

Inventory adjustments are published for all manual and external system generated adjustments where the Publish indicator for the reason code is checked. Adjustments are also published for other types of transactions in EICS where the merchandise system is expecting an adjustment for stock on hand updates, for example, receiving a DSD with damaged goods. An adjustment is created behind the scenes only for publishing purposes to notify the merchandising system to move the goods into the unavailable bucket. These system type adjustments are not considered an adjustment within EICS; however, they are published as such for integration purposes.

EICS subscribes to inventory adjustment messages from warehouse systems and updates the warehouse inventory buckets in EICS.

RIB payloads are used to communicate to external systems through RIB Integration.

The following table shows the list of RIB Payloads available for inventory adjustments.

| RIB Payload | Description |
|---|---|
| InvAdjustDesc | RIB payload that contains information about destination of the adjustment and an InvAdjustDtl. |
| InvAdjustDtl | Contains detailed information about the item adjustment. |

## Items

Items come to EICS from a merchandising system through the RIB (items, item loc messages). EICS also gets information about items associated to a supplier through the RIB. Extended attributes are not received or sent on RIB payloads.

| RIB Payload | Description |
|---|---|
| ItemDesc | This payload contains information about an item. It contains a wide variety of information about the item including suppliers, UPCs, ticketing information, image information, UDAs, and related items |
| ItemLocDesc | This payload contains information about an item at a specific location. |
| ItemSupDesc | This payload contains information about an item for a specific supplier. |

| RIB Payload | Description |
| --- | --- |
| ItemSupCtyDesc | This payload contains information about an item for a specific supplier within a specific country. |

## Stock Counts

Stock counts generate inventory adjustment when completed.

RIB payloads are used to communicate to external systems through RIB.

| RIB Payload | Description |
| --- | --- |
| InvAdjustDesc | RIB payload that contains information about destination of the adjustment and an InvAdjustDtl. |
| InvAdjustDtl | Contains detailed information about the item adjustment. |

EICS does not integrate using a web service to any other Oracle Retail products for stock counts.

## Transfers

The Transfer Shipping allows for creating shipment, dispatching shipment, canceling shipment, creating container, approving container, adjusting container, and canceling the container.

The Transfer Receiving dialog allows for confirming receipt, copying misdirected container, receiving container and detailed receiving.

This section covers creating transfer documents which are then included in a transfer shipment and dispatched to another store, warehouse, or finisher.

## Transfer Creation

Transfer documents can be created in the following ways:

- Requesting store can create a transfer request.
- Sending store can initiate a transfer by creating a transfer.
- Merchandising can create a transfer request.

Each transfer document will have one or more items.

## Transfer Messages

EICS will publish messages to Merchandising when the following happen:

- Transfer is rejected.
- Transfer is approved.
- Transfer quantity is updated from the shipment.

# Transfer Shipment Creation

Transfer Shipment describes the containers and the items for the shipment taking place. The shipment may be for one or more transfer documents if the transfer is going to the same destination. Dispatching a shipment will update the transfer document.

The user can create a shipment without referencing existing transfers or can create a new transfer on fly (Ad hoc transfer) based on the shipment information.

# Transfer Receiving

This transaction captures a delivery that took place from a warehouse, store, or finisher to the store receiving the delivery. It describes the containers and the items of the delivery that should be received by the store. Receiving a container of the delivery will update the transfer document.

**Figure 8-2    Transfer Request Flow**
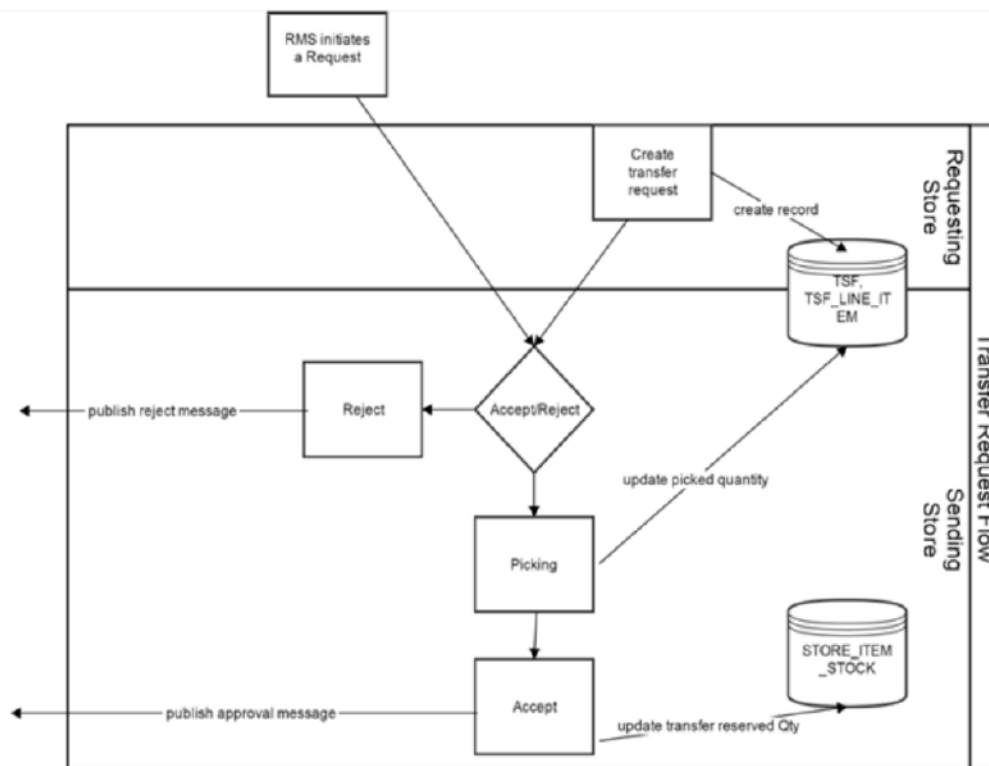
**Figure 8-3    Transfer Create Flow**
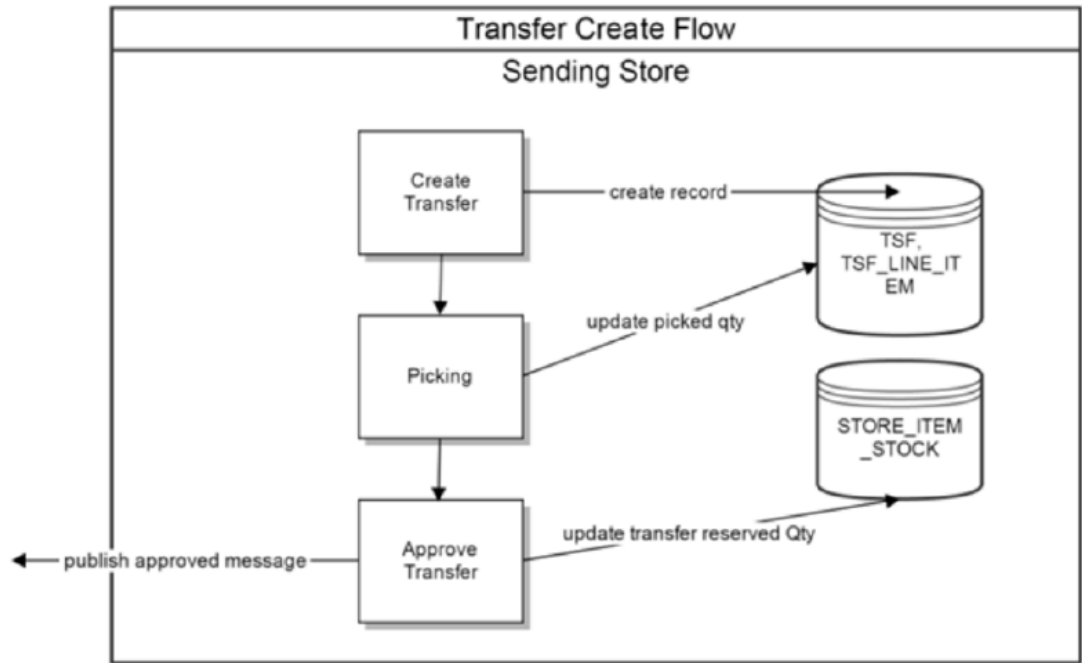


**Figure 8-4    Transfer Shipment Creation Flow**
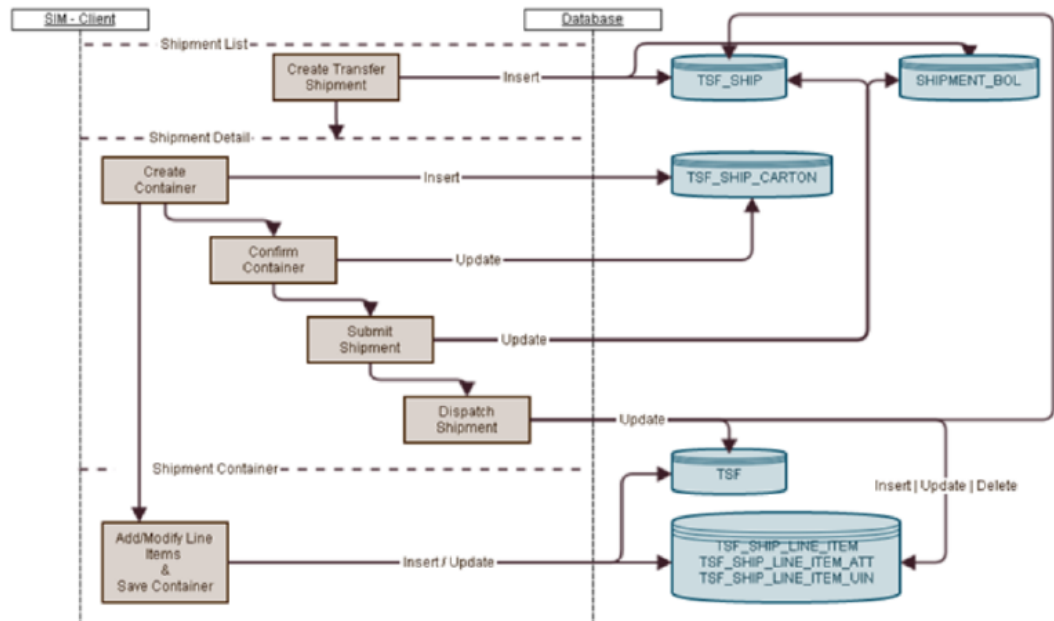
**Figure 8-5    Transfer Receiving Process Flow**
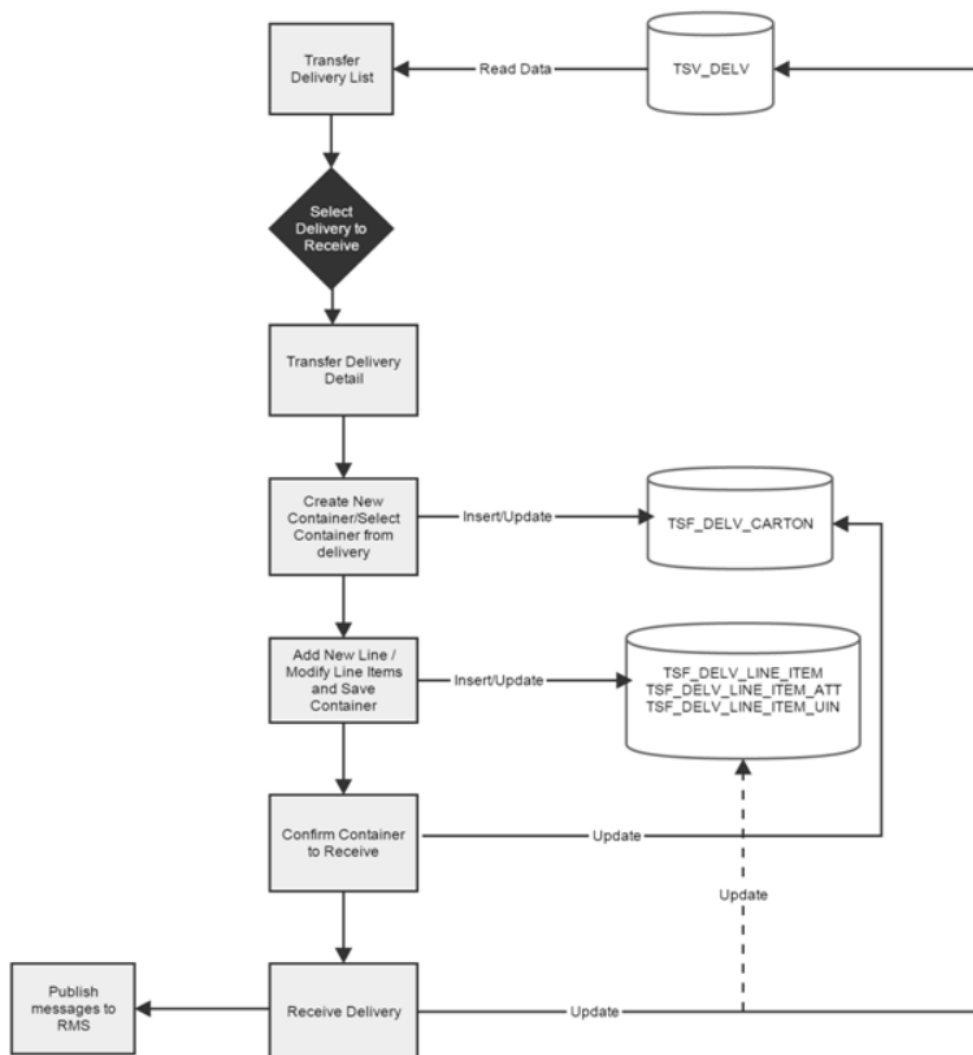


## Transfer Doc

| RIB Payload | Description |
|---|---|
| SODesc | This message is received from external systems when a stock order/transfer has been created |
| SOStatusDesc | This message is received from external systems when a stock order/transfer has been modified. |
| SORef | This message is received from external systems when a stock order/transfer has been deleted. |

# Transfer Shipment

| RIB Payload | Description |
| --- | --- |
| ASNOutDesc | This message is sent to external systems when the transfer shipment is dispatched. |
| ManifestCloseVo | This message is received from an external system to indicate physical shipment has been accepted. This will attempt to auto close the transfer shipment if all items are shipped. |
| ManifestDesc | This message is sent to an external system when manifesting is activated, and a transfer shipping container is confirmed. |
| ShipInfoDesc | This message is sent to an external system when pre-shipment notifications are active, and a transfer shipment is either submitted or dispatched (without previously being submitted). |
| SOStatusDesc | This message is sent to an external system when a transfer shipment container is saved with shipping quantities. It is also sent when a transfer shipment container is canceled but had shipping quantities. Increase and decrease of quantities is indicated by the SI or SD codes. |

# Transfer Receiving

| RIB Payload | Description |
| --- | --- |
| ASNInDesc | Sent from external system to indicate a delivery is tracking place. It creates a transfer delivery record within EICS when a store location is involved. |
| ReceiptDesc | Sent to external system when a transfer delivery is confirmed. |
| | Sent from external warehouse system when a transfer delivery is received at the warehouse. |

# Vendor Return

## RTV Creation

RTVs can only be created by a request from MERCHANDISING:

Each vendor return will have one or more items.

## RTV Shipment

Each RTV shipment will tie back to a single vendor return document.

RTV shipment can be created in two ways:

- From an externally initiated approved vendor return document.
- Creation of ad hoc vendor return shipment which will create an approved vendor return on the fly.

Each vendor return shipment will have one or more containers; each container in turn will have one or more items.

EICS may publish messages when the following happens:

- RTV shipment container is updated, and saved (Return To Vendor Publish)
- RTV shipment is cancelled or rejected (Return To Vendor Publish)
- RTV shipment is dispatched (Return to Vendor Publish and Ship Info Desc Publish, if dispatched without submitting)
- RTV shipment is submitted (Ship Info Desc Publish)
- RTV shipment container is confirmed (RTV manifesting, if configured)
- RTV shipment is submitted (Pre-shipment notification, if configured)

**Figure 8-6    RTV Creation Flow**
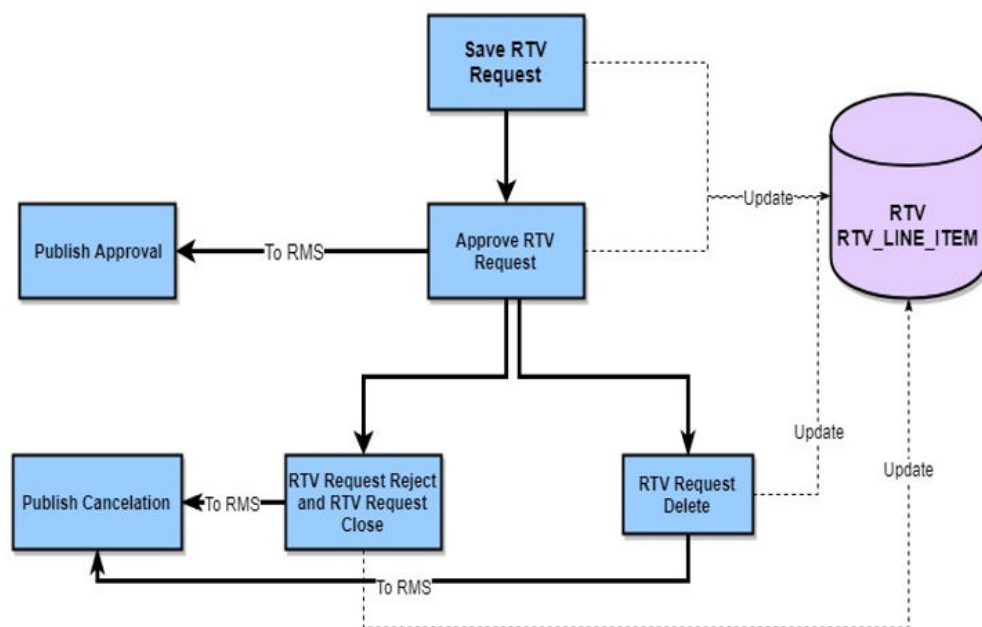
**Figure 8-7    RTV Shipment Flow**
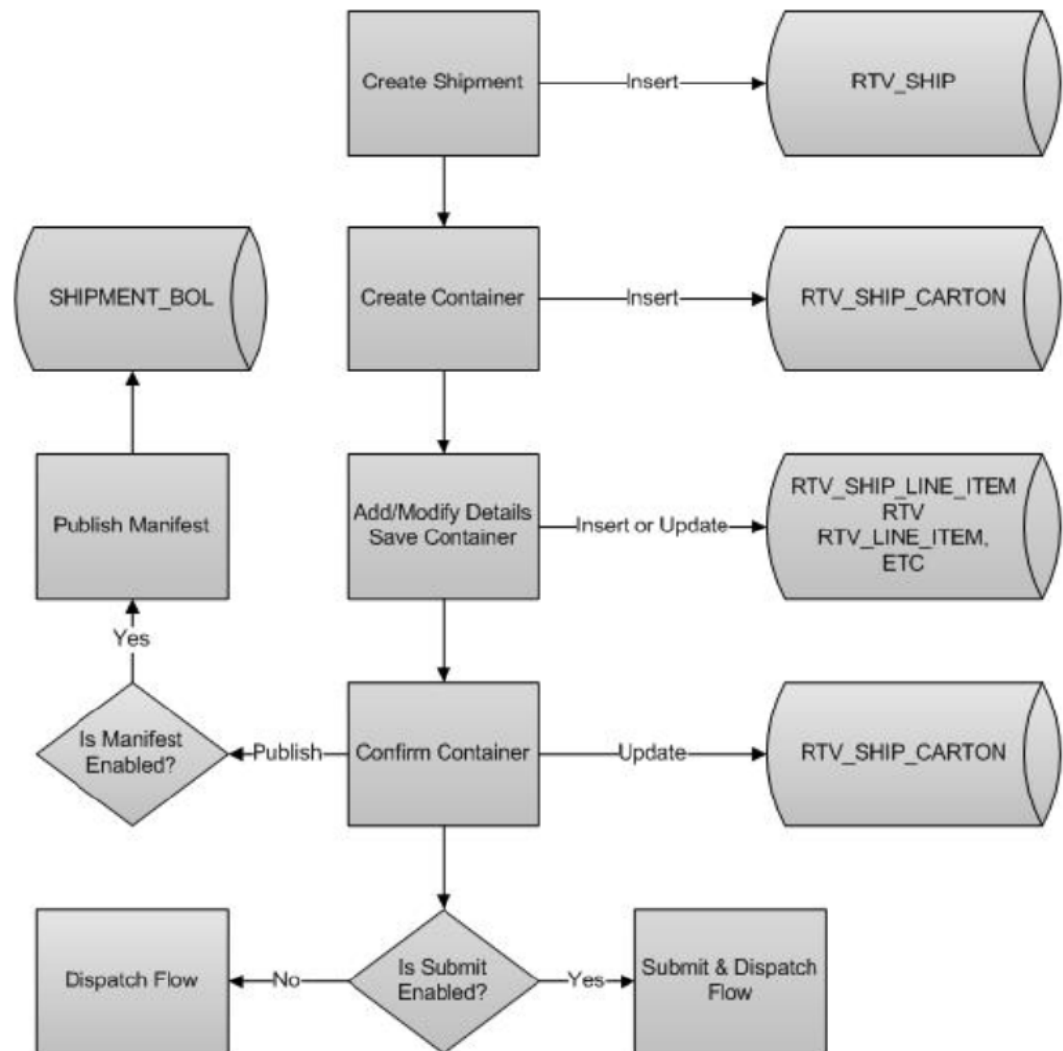
**Figure 8-8    RTV Shipment Submit and Dispatch Flow**
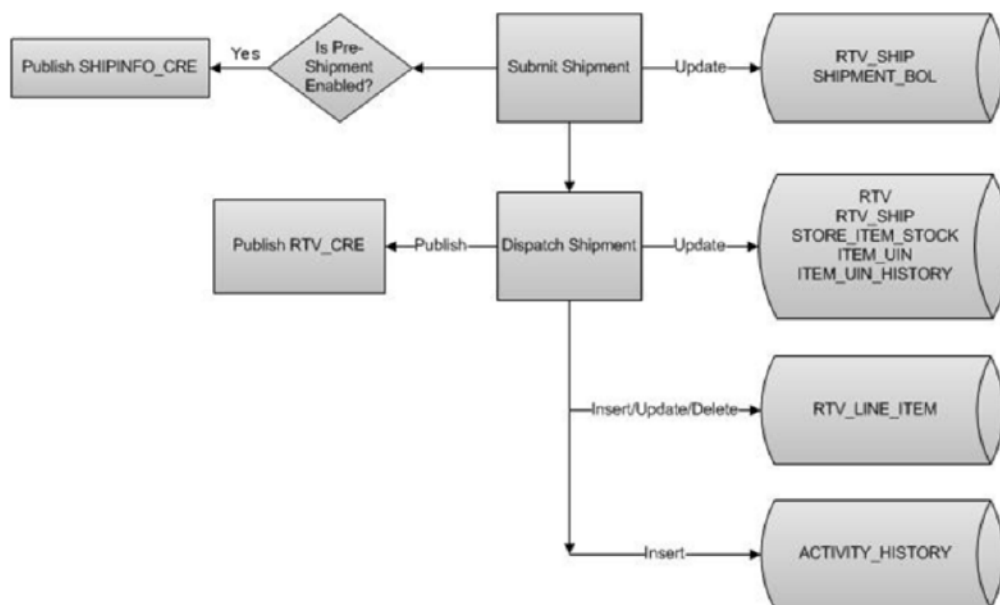


**Figure 8-9    RTV Shipment Dispatch Flow**
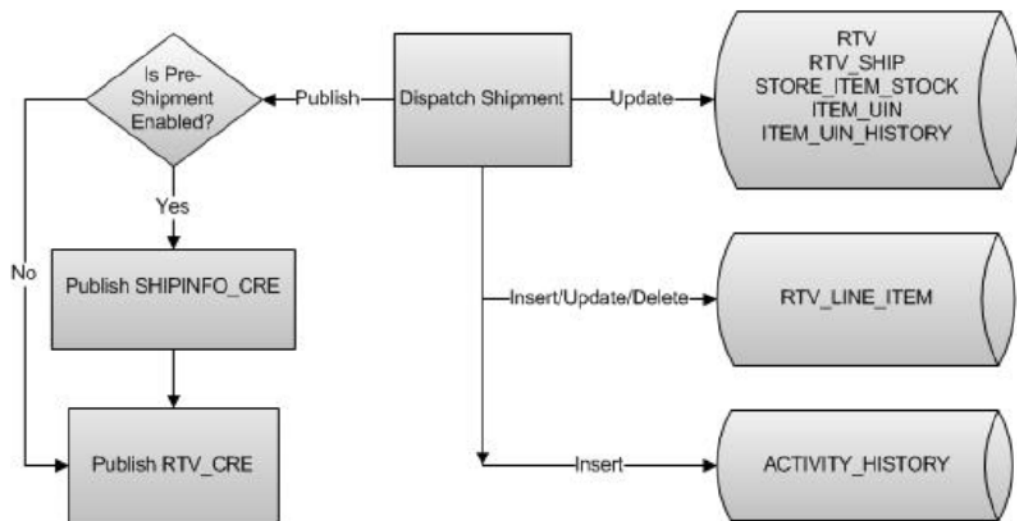


The following payloads are used in RTV operations.

| RIB Payload | Description |
| --- | --- |
| RTVReqDesc | This payload is sent from an external system to indicate a request for a vendor return. It creates or updates a vendor return document within EICS. It contains a series of RTVReqDtl. |
| RTVReqDtl | This payload contains the detailed information about the items on the vendor return. |

| RIB Payload | Description |
| --- | --- |
| RTVReqRef | This payload contains reference information about a vendor return when an external system wishes to attempt to cancel the return. |
| RTVDesc | This payload is sent from EICS to external systems when an RTV shipment is dispatched. |
| | This payload is sent from external warehouse system for vendor returns originating at warehouse. |

# Batch Integration

This section contains the following:

- Overview
- BDI-Based Batch Integration
- SFTP-Based Batch File Integration

## Overview

Batch bulk data is still a predominant integration style within Oracle Retail and its customers. Currently SIOCS bulk data integrations are implemented as BDI-Based batch integration or SFTP-Based batch integration.

## BDI-Based Batch Integration

Oracle BDI (Bulk Data Integration) is the latest Oracle Retail Integration tool for bulk data movement. The BDI infrastructure applications move data from one application to another. So, there is data producing applications and data consuming applications.

**Figure 8-10    BDI-Based Batch Integration Process**
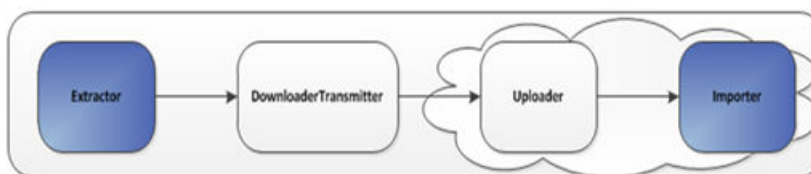


The BDI process:

- An Extractor extracts data from a data producing system into an outgoing intermediate integration interface table.
- A Downloader and transporter services data to receiver service.
- An Uploader uploads the data to an incoming intermediate integration interface table.
- An Importer imports the data from the interface stage tables into SIOCS application tables.

For more information about data seeding, see the *Oracle® Retail Enterprise Inventory Cloud Service Administration Guide*.

## Initial Foundation Data Seeding

Using BDI (Bulk Data Integration) Process Flow, you can load foundation data in bulk, including inventory and stock holding entities, from a data producing system into Oracle SIOCS.

Data Seeding is most suitable for:

- Loading initial inventory holding data
- Loading large volumes of data

Typically, you use data seeding for an initial data seeding and maintain the data in SIOCS thereafter, or through near-real time data integration with data producing system systems.

**Table 8-1    BDI Foundation Data Seeding Integration Object List**

| Business Object | Inbound/Outbound | Source/Destination |
| --- | --- | --- |
| Code Detail | Inbound | RMFCS |
| Code Head | Inbound | RMFCS |
| Delivery Slot | Inbound | RMFCS |
| Differentiator Group | Inbound | RMFCS |
| Differentiator | Inbound | RMFCS |
| Item Header | Inbound | RMFCS |
| Item Image | Inbound | RMFCS |
| Item Location | Inbound | RMFCS |
| Inventory Available Store | Inbound | RMFCS |
| Item Supplier | Inbound | RMFCS |
| Item Supplier Country Dimension | Inbound | RMFCS |
| Item Supplier Country | Inbound | RMFCS |
| Item Supplier Manufacturing Country | Inbound | RMFCS |
| Item Supplier UOM | Inbound | RMFCS |
| Merchandise Hierarchy | Inbound | RMFCS |
| Partner | Inbound | RMFCS |
| Partner Address | Inbound | RMFCS |
| Partner Org Unit | Inbound | RMFCS |
| Price History | Inbound | RMFCS |
| Pack Item | Inbound | RMFCS |
| Related Item | Inbound | RMFCS |
| Replenishment Item Location | Inbound | RMFCS |
| Supplier | Inbound | RMFCS |
| Supplier Address | Inbound | RMFCS |
| Store | Inbound | RMFCS |

**Table 8-1    (Cont.) BDI Foundation Data Seeding Integration Object List**

| Business Object | Inbound/Outbound | Source/Destination |
| --- | --- | --- |
| Store Address | Inbound | RMFCS |
| UDA | Inbound | RMFCS |
| UDA Item Date | Inbound | RMFCS |
| UDA Item Free Form | Inbound | RMFCS |
| UDA Item LOV | Inbound | RMFCS |
| UDA Values | Inbound | RMFCS |
| UOM Class | Inbound | RMFCS |
| UOM Conversion | Inbound | RMFCS |
| Warehouse | Inbound | RMFCS |
| Warehouse Address | Inbound | RMFCS |

## Recurring BDI Transaction Batches

BDI Transaction batches are recurring batches to move bulk data between integrated applications. You can use BDI Scheduler to schedule selective transaction data batches. For details, please see *Oracle Retail Bulk Data Integration Implementation Guide*.

**Table 8-2    BDI Transaction Batch List**

| Batch Name | Source/ Destination |
| --- | --- |
| Clearance TX Import | RPCS |
| Price Change TX Import | RPCS |
| Promotion TX Import | RPCS |
| Inventory Available Warehouse TX Import | RPCS |

Transfer Transaction Import

Purchase Order Transaction Import

| | |
| --- | --- |
| Clearance TX | Import clearance price from Oracle Retail Pricing Cloud Service via BDI Process Flow. |
| Price Change TX | Import clearance price from Oracle Retail Pricing Cloud Service via BDI Process Flow. |
| Promotion TX | Import promotion from Oracle Retail Pricing Cloud Service via BDI Process Flow. |
| Inventory Available Warehouse TX | Import warehouse available inventory from third-party via BDI Process Flow. |

## BDI Transmitter CLI - Client Batch File Loader

The BDI CLI Transmitter is a standalone command line tool to transmit batch interface data files to a destination BDI receiver system.

It is particularly used where the source system is non-BDI (that is, the source system does not have or use BDI Batch Job Admin application) but needs to send interface data files to a receiver system running the BDI Job Admin application.

The tool uses the BDI Job Admin Receiver REST service URL to transmit the data to the destination system. So, it is necessary that the destination system runs the BDI Job Admin application to use the tool.

> **✏ Note:**
>
> When use BDI Transmitter CLI tool to integrate with EICS, the app name for EICS in BDI is called SIM (Store Inventory Management).

The following figure illustrates the basic process for importing data using the BDI Transmitter CLI and EICS importer.

**Figure 8-11    BDI Transmitter CLI**
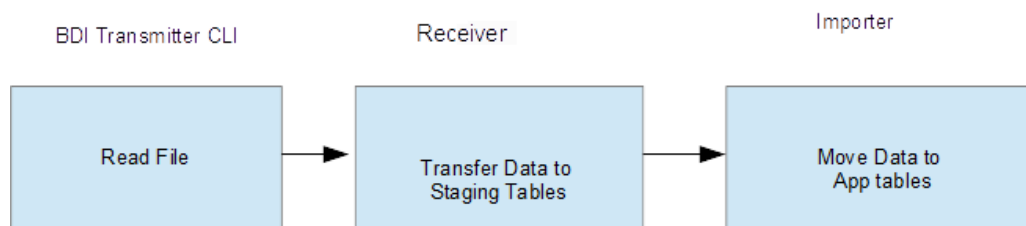


The BDI Transmitter CLI client tool process the data files from client machine, and load data into receiver application's staging inbound tables. The transmitter tool only load data into the Receiver's staging tables. To move data from receiver's staging tables into application's tables, you can use schedule EICS importer job to import the data from staging tables into EICS application tables.

Following is a list of batch file import implementations using BDI Transmitter CLI tool, alternatively the BDI External Process Flow can be chosen by integration implementation consultant, please see *Oracle Retail Bulk Data Integration Implementation Guide*.

**Table 8-3    BDI Transmitter CLI List**

| Batch Name | Description | Source/Destination |
|---|---|---|
| External Price TX Import | Import pricing data from external source into EICS, you can use BDI CLI Transmitter client tool to load CVS data file into BDI inbound interface tables, then setup SIOCS Batch job schedule to import data from inbound interface tables into EICS application tables. For additional detail, please see Appendix: BDI Transmitter CLI Procedure - Load External Data File. | Third Party |

**Table 8-3   (Cont.) BDI Transmitter CLI List**

| Batch Name | Description | Source/ Destination |
|---|---|---|
| External RFID TX Import | Import RFID data from external source, typically this can be implemented using BDI External Process Flow when external system implemented BDI Job Admin; alternatively, you can use BDI CLI Transmitter for CVS data file. For additional detail, please see Appendix: BDI Transmitter CLI Procedure - Load External Data File. | Third Party |
| Direct Delivery Transaction Import | Initial Seed Open Direct Delivery from External System.<br><br>See Appendix: Open Transaction Seeding via BDI for details. | Third Party |
| Purchase Transaction Import | Initial Seed Open Purchase Order from External System.<br><br>See Appendix: Open Transaction Seeding via BDI for details. | Third Party |
| Transfer Transaction Import | Initial Seed Open Transfer from External System.<br><br>See Appendix: Open Transaction Seeding via BDI for details. | Third Party |

# SFTP-Based Batch File Integration

In additional to BDI-Based Batch to move data between applications, EICS also supports SFTP-Based Batch File Integrations for list of batch modules. For additional detail, please see File Processing and SFTP Nightly Batch File Uploads.

SFTP-Based Batch File Integration Modules:

**Table 8-4   SFTP-Based Batch File Integration Modules**

| Object Name | Description | Source/Destination |
|---|---|---|
| POS Transaction Import | This batch imports POS Transaction records from the flat file that came from POS into the database staging table where polling timer framework will pick those staged requests and update the stock tables. | External POS |
| Retail Sale Audit Import | This batch program imports sales/order transaction data (SIM-ReSA file) that originated in Oracle Retail Xstore Point of Service. The batch also writes each failure record into a transaction log table. | ReSA |
| Store Sequence Import | This batch file import sequencing information like store sequence areas and items mapped to those areas from a flat file. | Third Party |
| Warehouse Item Stock Import | This batch imports the inventory of an items in a warehouse from flat files into warehouse item table. | Third Party |

**Table 8-4    (Cont.) SFTP-Based Batch File Integration Modules**

| Object Name | Description | Source/Destination |
|---|---|---|
| Inventory Extract | This batch extracts the inventory to a file that has been altered on or after the specified date for the specified store. The store id is not mandatory for the batch process. In case store id is not passed the files will be generated per store. The scheduled job will always generate files for all stores. | Order Broker |
| Stock Count Export | The stock count export files are generated when unit amount stock count is authorized. | Third Party |
| Clearance Price Change Import with On-prem RPM | This batch process imports the clearance price changes from flat files into item price table. Application uses this data to update price information of the items. For details, see Appendix: Pricing Import with On-premise RPM.This batch will be phased out in future release.For Clearance pricing integration with RPCS, see Clearance Transaction Import | On-prem RPM |
| Promotion Price Change Import with On-prem RPM | This batch process imports the promotion price changes from flat files into item price table. Application uses this data to update price information of the items.<br><br>For details, see Appendix: Pricing Import with On-premise RPM.<br><br>This batch will be phased out in future release.<br><br>For cloud promotion Pricing Integration, see Third Party Pricing Import | On-prem RPM |
| Regular Price Change Import with On-prem RPM | This batch process imports the price changes from flat files into item price table. Application uses this data to update price information of the items.<br><br>For details, see Appendix: Pricing Import with On-premise RPM.<br><br>This batch will be phased out in future release.<br><br>For cloud Pricing Integration, see Price Change Transaction Import | On-prem RPM |

# Web Services

EICS provides a large range of web services to manage the processing of information that is controlled within EICS. Each web service covers a topical area of functionality within EICS and contains numerous operations within to accomplish this functionality. This document is only meant as an outline or summary into using EICS web services and assumes the user has access to the fully documented APIs through the publishing of the web services themselves.

- Security Considerations

- Functionality
- Available Web Services
- Web Services Basic Design Principles
- Internally Managed vs Externally Managed
- Web Service Operation Basic Design Standards
- Interpreting Validation Errors

> **Note:**
>
> The WSDL files are available to download from My Oracle Support (MOS) Document 2614551.1.

## Security Considerations

Customers should create IDCS or OCI IAM user and the user should be assigned integration_users IDCS or OCI IAM application role to access the web-service endpoints.

See *Oracle Retail Enterprise Inventory Cloud Service Security Guide* and *Oracle Retail Enterprise Inventory Cloud Service User Guide -* Security chapter.

## Functionality

This document is intended to be used by someone who has read and understands all the functional areas and business functionality described in the *Oracle Retail EICS User Guide* and *Oracle Retail EICS Administration Guide.*

## Available Web Services

The following list contains a summary of the web services available in EICS.

| Web Service | Description |
| --- | --- |
| ActivityLock | This service is used to manage the locking of data within EICS. Data needs to be locked to be updated securely. |
| FulfillmentOrderDelivery | This service is used to manage fulfillment order deliveries (outgoing shipment to customers). It allows the creation, cancellation and dispatch of deliveries. |
| FulfillmentOrderPick | This service is used to manage fulfillment order picking within EICS. It allows the creation, deletion and confirmation of a pick to complete a fulfillment order. |
| FulfillmentOrderReversePick | This service is used to manage fulfillment order reverse picking within EICS. It allows the creation, update, deletion and confirmation of a reverse pick. |
| InventoryAdjustment | This service is used to manage inventory adjustments within EICS. It allows the creation, update, cancellation and confirmation of inventory adjustments. |
| ItemBasket | This service is used to manage item baskets within EICS. It allows the creation, update, and removal of item baskets. |

| Web Service | Description |
| --- | --- |
| OrderRequest | This service is used to create, read, update, approve, cancel and lookup store orders. |
| POSTransaction | This service processes external point-of-sale transactions updating the inventory accordingly. A point-of-sale is considered an externally managed transaction (internally and externally managed transaction are covered later in this document). |
| ProductGroup | This service is used to create or update a product group. |
| ProductGroupSchedule | This service is used to create, update, or cancel a product group schedule. |
| ReplenishmentGap | This service is used to create, update, or delete a replenishment gap. |
| RfidInventory | This service is used to create, update or delete a RFID facility zone. It is also used to refresh inventory and to process RFID events. |
| ShelfAdjustment | This service is used to create, update, cancel or confirm a shelf adjustment. |
| ShelfReplenishment | This service is used to create, update, cancel or confirm a shelf replenishment. |
| StockCount | This service is used to retrieve the details of a stock count or a stock count child (section of stock count). |
| Store | This service is used to retrieve information about stores such as store detail, associated stores, or transfer zones. |
| StoreFulfillmentOrder | This service is used to manage fulfillment orders within EICS. It allows for the cancellation and rejection of orders and items. |
| StoreInventory | This service is used to lookup information about inventory positions and has several different operations to do so. |
| StoreInventoryUIN | This service is used to create, update, generate or read a UINs. |
| StoreInventoryISN | This service is used to create, update or delete ISN data in EICS. |
| StoreItem | This service is used to lookup various information about an item within the store. |
| StoreItemPrice | This service is used to lookup prices about items within a store. |
| StoreNotification | This service is used to create new notifications within the system. |
| StoreShipmentManifest | This service is used to close documents based on shipped container information. |
| StoreShipmentReason | This service is used to retrieve shipment reasons codes to use when creating shipments. |
| StoreTicket | This service is used to create tickets and lookup ticket formats. |
| StoreTransfer | This service is used to create, update and request a transfer, which describes the intent to ship items to another store or to a warehouse. It is also used to approve or reject that request. It can be used to directly create, update, approve, cancel, or close an actual transfer. |
| TransferDelivery | This service is used to update, receive, or confirm a transfer delivery (delivery arriving from another store or warehouse). It is also used to create, update, receive, cancel, or confirm the containers on that delivery. |

| Web Service | Description |
| --- | --- |
| TransferShipment | This service is used to create, update, submit, or dispatch a transfer shipment (shipment going out to another store or warehouse). It is also used to create, update, cancel, or confirm the containers on that shipment. |
| VendorDelivery | This service is used to update, receive, reject, or confirm a vendor delivery (delivery arriving from a supplier). It is also used to create, update, cancel, or confirm the containers on that delivery. |
| VendorReturn | This service is used to create, update, approve, cancel, or close a vendor return document, which describes the intent to ship items to a supplier. |
| VendorShipment | This service is used to create, update, open, submit, cancel submit or dispatch a vendor shipment (outgoing shipment to a supplier). It is also used to create, update, cancel, submit or confirm the containers on that shipment. |

# Web Services Basic Design Principles

**Empty Response**

In the cast that a web service does not return any information (an empty list), the external system needs to understand that this is a valid response that indicates no item, transaction or queried information was found or retrieved. For example, performing a lookup in which the search criteria entered matched no input.

**Error Return Key**

Errors returned through a web service will be in the form of a key. This key should be translated into correct language and verbiage by the external system. EICS will not do this translation or provide English verbiage for the encountered web service error.

**Boolean Data Type**

If a Boolean is the data type on the interface to EICS, and no value is provided, EICS will default the value to False.

**Configured System Options in EICS**

Web services apply system configurations to the request that are coming in through the web service but assumes that all input validation that requires user interaction to confirm has been completed by the access or of the web service. This system configuration user-interaction option will be assumed to have been confirmed during the web service processing. In case the system option is a fixed restriction that does not require user interaction, and the input fails this restriction, the web service will return an error. For example:

- Shipping inventory when inventory is less than 0 can be allowed by the user of EICS. The web service assumes that the third-party application did prompt the user or that their business always allows the user to do this activity.

- Adding a non-ranged item requires both a system configuration option to be enabled and the user to confirm the process. If the system configuration does not allow it, the web service will block the transaction and return an error. If the system configuration does allow adding non-ranged items, it is automatically assumed that a user confirmed its addition, and the web service adds the item.

- Allowing Receiver Unit Adjustments are dependent on a period of time. If a receiver unit adjustment were to come into EICS after that period of time, it would automatically be rejected, and the web service would return an error regardless of presentation or confirmation of user done by the external system.

## Internally Managed vs Externally Managed

**Internally Initiated**

Internally initiated indicates the EICS was responsible for the original creation of the transaction being processed. A web service that creates a new transaction within EICS to be managed creates an internally initiated transaction.

**Externally Initiated**

Externally initiated indicates that another system created the transaction, has information about it, and notifies EICS of its creation through a notification system, not by requesting EICS create new information. EICS might manage the data after the notification but did not create the data.

**Internally Managed**

Internally managed data is information in which EICS is responsible for tracking its state and processing its life cycle. Our deliveries and shipments are primary examples of this. They may be externally initiated or internally initiated, but either way, they are internally managed. EICS is responsible for approving, picking, packing, manifesting, and dispatching the system and internally manages that process.

**Externally Managed**

Externally managed data is information that EICS does not process or track and is simply informed about after the externally managed data is complete. Point-of-sale transactions are a perfect example of this. We do not manage the sale, but once it is complete, EICS is notified and adjusts the inventory accordingly.

**Web Services**

EICS web services are intended for integration in order to allow a system using those services to control the flow and processing within EICS. Our web services are primarily designed (almost all of them) to internally manage the information. The services are intended to be used real time with the steps such as approving, picking, and dispatching occurring with real time access to EICS web services while the process is happening.

EICS web services are not designed for externally managed information. If a system is controlling the state managements itself and not informing EICS until later, this will produce out-of-sync inventory. For example, if you create a shipment, pack the shipment, and send it out and then a day later use the web service, to create, update, and dispatch the shipment, all dates and processing of inventory movements will be tagged with the later date as if they occurred real time when the web service is used.

The point-of-sale service is an externally managed service, where the timestamp on the service can be any date and EICS handles the logic of dating things according to that timestamp. Inventory Adjustment also has an "adjustment date" which represents the time the adjustment took place and so the movement of inventory can be controlled externally.

# Web Service Operation Basic Design Standards

This section discusses the general approach and design standards for naming and intent regarding operations within a web service.

**Lookup**

Lookup operations take either an identifier of a set of criteria and find all the relevant records associated to it. A thin or light view of the data being asked for is returned giving reference to information you can do further interrogation on.

**Read**

Read operations take an identifier and return all relevant information to it. It may only be one level, however. For example, reading a transfer shipment returns only all the information at the shipment level and does not read information at the container or item level. Usually, the entity that contains items will also retrieve the items. Reading a container will return the container information and the item information within.

**Create**

Create usually inserts and generates something new and returns an identifier, reference, or handler to that information. Create normally does not take a great deal of information, such as items or anything, but rather gives you a set of IDs that then lets you update the transaction with that reference.

**Save or Update**

Save or update is used to modify the data usually without changing state on the transaction. The save or update operation is used to add items, remove items, edit attributes, change quantities and all the other tasks one does during a process.

**Approve, Cancel, Confirm or Dispatch**

Activities that change state take in a simple identifier and then process that state change. To dispatch a shipment, you pass in a reference only to the shipment and it becomes shipped, updating the inventory. This means all changes are done through the save operations prior to making the state change.

# Interpreting Validation Errors

If some data could not be processed, the web service will return a fault or a validation fault. The general form that a fault will take is to be a series of problem detail nodes containing a key and value that describes the fault. The first problem detail node will have the key ERROR and the value will be a description of the error type such as INVALID_INPUT. This will be followed by a series of nodes where the KEY is an object class name (ex: Transfer) and the value is its identifier (ex: 123) describing the hierarchy of data the error took place in. For example, a transfer container fault would have two nodes (Transfer:123) and then (TransferCarton:456). If a specific attribute is known, the final node in any problem detail series is will have the key ATTRIBUTE and the value will be the name of the attribute of the error (ex: ITEM_ID:A5X).

| Problem Detail Name | Value |
| --- | --- |
| ERROR | This describes the error (for example: INVALID_INPUT) |

| Problem Detail Name | Value |
| --- | --- |
| ATTRIBUTE | Identifies the specific attribute that had an error. |

EICS follows the same business rules when processing information from a web service as it does from any of its clients, so the same business rules and functionality that exist in the User's Guide also exists for the web service. Understanding the basic functionality will help interpret why the validation or processing error occurred.

## Common Error Codes

The following codes are paired as values to the ERROR Key:

| Error Code | Description |
| --- | --- |
| ACTIVITY_LOCK_NOT_GRANTED | Indicates that a requested activity lock on a piece of data was not granted. |
| DUPLICATE_INPUT | Indicates the service would create a duplication of input that should be unique. |
| INVALID_DATE_RANGE | Indicates the end date of a date range is prior to the start date. |
| INVALID_INPUT | Indicates that the input is invalid. This error is usually followed by object and attribute information. |
| INVALID_ITEM | Indicates the item does not exist in the system. |
| INVALID_STATE_FOR_UPDATE | Indicates the transaction or data specified is not in a state that allows it to be updated (such as canceled). |
| INPUT_MISMATCH | Indicates the input to the web service has been altered incorrectly when compared to existing data. For example, the store identifier is different on the web service request than the currently existing transaction. |
| INPUT_TOO_LARGE | Indicates the input in the web service is larger than is allowed in the transaction date. |
| ITEM_NOT_RANGED | Indicates the item has not been activated in the location for which the request is made. |
| MULTIPLE_STORE | Indicates a batch of input data (such as a point-of-sale transaction) was for more than one store in a single web service call. |
| TIMEZONE_NOT_GMT | Indicates the time input of the web services was not in GMT. |
| UOM_MISMATCH | Indicates a mismatch of unit of measure information between the input and currently existing data that does not allow the information to be accurately merged. |

## Validation Error (Fault Example)

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Body>

<ns0:Fault xmlns:ns0="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://www.w3.org/2003/05/soap-envelope">

<faultcode>ns0:Server</faultcode>

```
<faultstring>VALIDATION_ERROR</faultstring>

<detail>

<ns0:ValidationWSFaultException xmlns:ns0="http://www.oracle.com/retail/integration/
services/exception/v1">

<ns0:shortErrorMessage>VALIDATION_ERROR</ns0:shortErrorMessage>

<ns0:BusinessProblemDetail>

<ns0:problemDescription>VALIDATION_ERROR</ns0:problemDescription>

<ns0:ProblemDetailEntry>

<ns0:name>ERROR</ns0:name>

<ns0:value>INVALID_INPUT</ns0:value>

</ns0:ProblemDetailEntry>

<ns0:ProblemDetailEntry>

<ns0:name>ShlfAdjRef</ns0:name>

<ns0:value>1</ns0:value>

</ns0:ProblemDetailEntry>

<ns0:ProblemDetailEntry>

<ns0:name>ATTRIBUTE</ns0:name>

<ns0:value>shelfAdjustmentId</ns0:value>

</ns0:ProblemDetailEntry>

</ns0:BusinessProblemDetail>

</ns0:ValidationWSFaultException>

</detail>

</ns0:Fault>

</S:Body>

</S:Envelope>
```

## Business Error (Fault Example)

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Body>

<ns0:Fault xmlns:ns0="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1="http://
www.w3.org/2003/05/soap-envelope">

<faultcode>ns0:Server</faultcode>

<faultstring>BUSINESS_ERROR</faultstring>

<detail>
```

&lt;ns0:ValidationWSFaultException xmlns:ns0="http://www.oracle.com/retail/integration/services/exception/v1"&gt;

&lt;ns0:shortErrorMessage&gt;BUSINESS_ERROR&lt;/ns0:shortErrorMessage&gt;

&lt;ns0:BusinessProblemDetail&gt;

&lt;ns0:problemDescription&gt;BUSINESS_ERROR&lt;/ns0:problemDescription&gt;

&lt;ns0:ProblemDetailEntry&gt;

&lt;ns0:name&gt;ERROR CODE&lt;/ns0:name&gt;

&lt;ns0:value&gt;ADJUSTMENT_NOT_FOUND&lt;/ns0:value&gt;

&lt;/ns0:ProblemDetailEntry&gt;

&lt;/ns0:BusinessProblemDetail&gt;

&lt;/ns0:ValidationWSFaultException&gt;

&lt;/detail&gt;

&lt;/ns0:Fault&gt;

&lt;/S:Body&gt;

&lt;/S:Envelope&gt;

# Web Services

Web services available in EICS:

## ActivityLock

The following operations are available within the ActivityLock web service.

| Operation | Description |
|-----------|-------------|
| lookupActivityLock | Retrieves information about one or more activity locks that match the input criteria. |
| readActivityLock | Retrieves detailed information about a single lock using its identifying reference. |
| createActivityLock | Created an activity lock on a transaction. |
| deleteActivityLock | Deletes an activity lock thereby releasing processing on a transaction. |

**Standard Usage**

An activity lock is a record indicating the user, time, and a piece of information (a transaction) that should be considered "locked". All server processing validates that the accessing user has a lock on the information before updating, notifying the current user if someone else has modified the information while they were locked and preventing the stale update.

Developers should create locks on information prior to performing update calls and delete locks when the update if finished. For example, create a lock on inventory adjustment with ID 123 with the ActivityLock service, then use

`saveInventoryAdjustment` in the Inventory Adjustment service with Adjustment 123, and then delete the activity lock using the ActivityLock service. If you do not gain the lock, you will receive an error when attempting to save an inventory adjustment.

# FulfillmentOrderDelivery

The following operations are available within the FulfillmentOrderDelivery web service.

| Operation | Description |
|---|---|
| lookupFulfillmentOrderDelivery Headers | Retrieves summary information for fulfillment order deliveries that match the search criteria input. |
| readFulfillmentOrderDeliveryD etail | Reads the complete detailed information about a fulfillment order including items and quantities. |
| createFulfillmentOrderDelivery | Creates a new fulfillment order delivery including items and quantities in an in-progress status to be further worked on. |
| cancelFulfillmentOrderDelivery | Cancels the fulfillment order delivery. |
| cancelFulfillmentOrderDelivery Submission | Cancels the fulfillment order review and moves it back into in-progress status for further work. |
| dispatchFulfillmentOrderDelive ry | Dispatches the fulfilment order delivery completing the delivery and updating the inventory. |
| submitFulfillmentOrderDelivery | Submits the fulfillment order delivery for review prior to dispatching. |
| updateFulfillmentOrderDelivery | Updates a fulfillment order delivery including items and quantities. This operation requires an activity lock. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the fulfillment order delivery. |

**Standard Usage**

A user can create a delivery by using `createFulfillmentOrderDelivery` references the fulfillment order to make a delivery for. The user can then use `updateFulfillmentOrderDelivery` to fill in all the quantities that are going to be shipped and finally use `dispatchFullfillmentOrderDelivery` to indicate that the order has been shipped out, which moves the inventory appropriately.

# FulfillmentOrderPick

The following operations are available within the FulfillmentOrderPick web service.

| Operation | Description |
|---|---|
| lookupFulfillmentOrderPickHea ders | Retrieves summary information for fulfillment order picks that match the search criteria input. |
| readFulfillmentOrderPick | Reads the complete detailed information about a fulfillment order pick including items and quantities. |
| confirmFulfillmentOrderPick | Confirm the fulfillment order pick which allows it to move on to the delivery cycle. |
| deleteFulfillmentOrderPick | Deletes a fulfillment order pick. |
| createFulfillmentOrderPickByF ulfillmentOrder | Generate a pick based on the information in a fulfillment order. |

| Operation | Description |
| --- | --- |
| createFulfillmentOrderPickByBin | Generate a pick based on a number of bins selecting orders as needed to fill the bins. |
| updateFulfillmentOrderPick | Update the item and quantity information about a pick. This operation requires an activity lock. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the fulfillment order pick. |

**Standard Usage**

Picking is used to reserve or set aside quantities for a later delivery. The user can create a pick for an order using `createFulfillmentOrderPickByFulfillmentOrder` or create a bin to places multiple orders in with `createFulfillmentOrderPickByBin`. The picked quantities can be updated through the `updateFullfillmentOrderPick` operation and when the pick is finished, it can be finalized with `confirmFulfillmentOrderPick` which sets assigned the goods as reserved in inventory.

# FulfillmentOrderReversePick

The following operations are available within the FulfillmentOrderReversePick web service.

| Operation | Description |
| --- | --- |
| lookupReversePickHeaders | Retrieves summary information for fulfillment order reverse picks that match the search criteria input. |
| lookupReversePickDetail | Reads the complete detailed information about a fulfillment order reverse pick including items and quantities. |
| createReversePick | Creates a new fulfillment order reverse pick for the specified fulfillment order. |
| deleteReversePick | Deletes a fulfillment order reverse pick. |
| updateFulfillmentOrderReversePick | Updates the items and quantities on a fulfillment order reverse pick. This operation requests an activity lock. |
| confirmReversePick | Confirms the fulfillment order reverse pick completing the process and assigning the inventory back to a location within the store system. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the fulfillment order reverse pick. |

**Standard Usage**

Reverse Picking is used to take reserved quantities and place them back into available inventory. The user can create a reverse pick with `createReversePick`. The quantities to return can be updated through the `updateFulfillmentOrderReversePick` operation and when the reverse pick is ready, it can be finalized with `confirmReversePick` which moves reserved inventory back into available inventory.

## InventoryAdjustment

The following operations are available within the InventoryAdjustment web service.

| Operation | Description |
| --- | --- |
| lookupInventoryAdjustmentReason | Retrieve a complete list of adjustment reasons that can be used when updating or saving an inventory adjustment. Reason codes are attached to each line item. |
| lookupNonSellableQuantityType | Retrieve a complete list of non-sellable quantity types. These codes indicate the reason that unavailable inventory is unavailable. |
| lookupInventoryAdjustmentHeader | Retrieve summary information about inventory adjustment transactions based on the search criteria sent. |
| readInventoryAdjustmentDetail | Retrieve the complete detailed information about an inventory adjustment, including its item information, based on a unique reference/id. |
| saveInventoryAdjustment | Creates or updates the information about an inventory adjustment in the data store. You can alter information about items and quantities using this operation. This operation requires having an activity lock. |
| confirmInventoryAdjustment | Confirms the inventory adjustment, updating all the inventory positions, and closing the adjustment. |
| saveAndConfirmInventoryAdjustment | Performs the functionality of saveInventoryAdjustment and immediately thereafter performs the confirmInventoryAdjustment functionality. See those operations. |
| cancelInventoryAdjustment | Cancel an inventory adjustment. This can only be done prior to the inventory adjustment being confirmed. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the inventory adjustment. |

**Standard Usage**

A new inventory adjustment can be created using the `saveInventoryAdjustment` operation. Alternatively, the user can `lookupInventoryAdjustmentHeader` to find a specific inventory adjustment to work on. Either way, `saveInventoryAdjustment` can be used to update the information on an open adjustment. The `lookupInventoryAdjustmentReasons` will retrieve the reasons codes that need to be assigned to items when you update an adjustment. When the adjustment contains all the information you need, the `confirmInventoryAdjustment` operation will finalize the inventory adjustment and shift the inventory appropriately.

## ItemBasket

The following operations are available within the Item Basket web service.

| Operation | Description |
| --- | --- |
| lookupItemBasketHeaders | Retrieve a list of item basket headers based on search criteria which contain summary information about the item basket. |
| lookupItemBasketTypes | Retrieve a complete list of item basket types to use when creating a new item basket. |
| createItemBasket | Creates a new item basket. |

| Operation | Description |
| --- | --- |
| readItemBasketDetail | Retrieve the complete detailed information about an item basket based on an identifier. |
| deleteItemBasket | Cancels an item basket. The basket will no longer be usable and will be marked for eventual purge from the data store. This operation requires an activity lock. |
| saveItemBasket | Updates an item basket. This operation requires an activity lock. |
| copyItemBasket | Creates a new item basket with the same information as an existing item basket. |
| confirmItemBasket | Moves the item basket to a completed state and allows it to be used within logic throughout the system. This operation requires an activity lock. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the item basket. |

**Standard Usage**

A new item basket can be created using the `saveItemBasket` operation. Alternatively, the user can `lookupItemBasketHeader` to find a specific item basket to work on. Either way, `saveItemBasket` can be used to update the information on an item basket. When the item basket contains all the information you need, the `confirmItemBasket` operation will finalize the item basket and make it available to use in other areas of the system.

# OrderRequest

The following operations are available within the Order Request web service.

| Operation | Description |
| --- | --- |
| lookupOrderRequestHeader | Retrieves store order request headers based on the query criteria. |
| readOrderRequest | Retrieves detailed information about a store order request. |
| createOrderRequest | Creates a new store order request. |
| updateOrderRequest | Updates an existing store order request. |
| approveOrderRequest | Approve a store order request. |
| cancelOrderRequest | Cancels a store order request. |
| lookupDeliveryTimeSlot | Retrieves delivery time slots. |
| lookupOrderContext | Retrieves contexts available for store order requests. |
| lookupOrderArea | Retrieves store order request areas that could be used for restriction. |
| lookupCustomAttributeAdmins | Retrieves all the custom attributes admins configured for store order requests. |

**Standard Usage**

A new store order can be created using the `createOrderRequest` operation. The information about store order can be read by `readOrderRequest`. The store order can be updated using `updateOrderRequest` and can be approved using

approveOrderRequest and can be canceled using cancelOrderRequest. The
lookupOrderRequestHeader is used to find the store orders.

## POSTransaction

The following operations are available within the StoreNotification web service.

| Operation | Description |
| --- | --- |
| processPOSTransactions | Processes a point-of-sale transaction or transactions through an asynchronous process. This is designed to optimize the processing of large numbers of point-of-sale transaction (up to a limit of 5,000 PosTrnItm details). |

**Standard Usage**

POS may integrate its transactions to EICS using this web service. The service processes point-of-sale transactions through an asynchronous process. This service is restricted to a maximum amount of overall PosTrnItms, though they may be distributed between any number of actual PosTrn transactions. The limit is determined by an operation only configuration value: POS Transaction Max Limit. Exceeding this limit causes a web service fault to occur. These transactions may belong to multiple store identifiers. The processing operation validates the input, parses the payload information, creates a POSTransaction object within EICS, and stores these records to be processed later. See Sales Integration for additional information.

## ProductGroup

The following operations are available within the ProductGroup web service.

| Operation | Description |
| --- | --- |
| lookupProductGroupHeader | Retrieves list of summary information about a product group that match the search criteria input. |
| readProductGroup | Retrieves the detailed information about a single product group based on its unique reference. |
| saveProductGroup | Creates or updates a product group. The input contains all the detailed information about the product group. An activity lock is needed for this operation. |

**Standard Usage**

With this web service, the user can create or update the contents of a product group, a collection of items associated with a certain type of grouping, such as stock counts. The user can find the product group with lookupProductGroupHeader, read in the entire product group with readProductGroup and then, if the group is still open, update the contents of the product group with saveProductGroup.

## ProductGroupSchedule

The following operations are available within the ProductGroupSchedule web service.

| Operation | Description |
| --- | --- |
| lookupProductGroupScheduleHeader | Retrieves list of summary information about a product group schedule that match the search criteria input. |
| readProductGroupSchedule | Retrieves the detailed information about a single product group schedule based on its unique reference. |
| saveProductGroupSchedule | Creates or updates a product group. The input contains all the detailed information about the product group schedule. An activity lock is needed for this operation. |
| cancelProductGroupSchedule | Cancels the product group schedule. |

**Standard Usage**

With this web service, the user can create or update the contents of schedule, which uses a product group to generate activity within EICS. The user can find the schedule with `lookupProductGroupScheduleHeader`, read in the entire schedule with `readProductScheduleGroup` and then, if the schedule is still open, update the contents of the schedule with `saveProductGroupSchedule`.

## ReplenishmentGap

The following operations are available within the ReplenishmentGap web service.

| Operation | Description |
| --- | --- |
| lookupReplenishmentGapHeaders | Retrieves list of summary information about replenishment gaps that match the search criteria input |
| readReplenishmentGap | Retrieves the detailed information about a single replenishment gap based on its unique reference. |
| saveReplenishmentGap | Creates a new replenishment gap or updates the detailed information about a replenishment gap. If update, this operation requires an activity lock. |
| deleteReplenishmentGap | Deletes a replenishment gap. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the replenishment gap. |

**Standard Usage**

With this web service, the user can create or update the contents of replenishment gap list which can then be used in creation of shelf replenishment within EICS. A new replenishment gap list can be created using `saveReplenishmentGap`. The user can update existing replenishment gap list with `saveReplenishmentGap`, find replenishment gap lists with `lookupReplenishmentGapHeaders`, read in the entire replenishment gap list with `readReplenishmentGap` and delete a replenishment gap list with `deleteReplenishmentGap`.

## RfidInventory

The following operations are available within the RfidInventory web service.

| Operation | Description |
| --- | --- |
| deleteRfidZone | Deletes a zone within a facility. A zone cannot be deleted if RFID tags still exist within the zone. |
| lookupRfidZones | Returns details about all the zones within a particular facility. |
| processRfidEvents | Processes Radio-Frequency-Identification based events. |
| saveRfidZone | Creates or updates the details of a facility zone. |

**Standard Usage**

With this web service, the user can create or update RFID zone within EICS. A new RFID zone can be created using `saveRfidZone`. The user can update existing RFID zone list with `saveRfidZone`, find RFID zome lists with `lookupRfidZones` and delete a RFID zone with `deleteRfidZone`. The user can process a RFID based events using `processRfidEvents` and refreshes the inventory for items with `refreshInventory`.

# ShelfAdjustment

The following operations are available within the ShelfAdjustment web service.

| Operation | Description |
| --- | --- |
| lookupShelfAdjustmentHeaders | Retrieves list of summary information about shelf adjustments that match the search criteria input. |
| readShelfAdjustment | Retrieves the detailed information about a single shelf adjustment gap based on its unique reference. |
| saveShelfAdjustment | Creates a new shelf adjustment or updates the detailed information about a current shelf adjustment. If update, this operation requires an activity lock. |
| confirmShelfAdjustment | Confirms a shelf adjustment completing the workflow and moving inventory positions. |
| cancelShelfAdjustment | Deletes a shelf adjustment. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the shelf adjustment. |

**Standard Usage**

Shelf adjustments are used to adjust the shop-floor or backroom stock in case of any discrepancy. A new shelf adjustment can be created using `saveShelfAdjustment`. The user can update existing shelf adjustment with `saveShelfAdjustment`, find shelf adjustment with `lookupShelfAdjustmentHeaders`, read in the entire shelf adjustment with `readShelfAdjustment`, cancel a shelf adjustment with `cancelShelfAdjustment` and confirm a shelf adjustment with `confirmShelfAdjustment`.

# ShelfReplenishment

The following operations are available within the ShelfReplenishment web service.

| Operation | Description |
|---|---|
| lookupShelfReplenishmentHeaders | Retrieves list of summary information about shelf replenishments that match the search criteria input. |
| readShelfReplenishment | Retrieves the detailed information about a single shelf replenishment gap based on its unique reference. |
| createShelfReplenishment | Creates a new shelf replenishment. |
| updateShelfReplenishment | Updates the detailed information about a current shelf replenishment. This operation requires an activity lock. |
| confirmShelfReplenishment | Confirms a shelf replenishment completing the workflow and moving inventory positions. |
| cancelShelfReplenishment | Deletes a shelf replenishment. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the shelf replenishment. |

**Standard Usage**

Shelf replenishment is used to replenish shop-floor stock from backroom or delivery bay. A new shelf replenishment can be created with `createShelfReplenishment`. The user can find shelf replenishments with `lookupShelfReplenishmentHeaders`, read in the entire shelf replenishment with `readShelfReplenishment`, update the shelf replenishment with `updateShelfReplenishment`, confirm the shelf replenishment with `confirmShelfReplenishment` and cancel the shelf replenishment with `cancelShelfReplenishment`.

# StockCount

The following operations are available within the StockCount web service.

| Operation | Description |
|---|---|
| lookupStockCountHeaders | Retrieves list of summary information about a stock count that match the search criteria input. |
| readStockCountDetail | Retrieves the detailed information about a single stock count based on its unique reference. This contains a list of summary information about the child counts. |
| readStockCountChild | Retrieves the detailed information about a single stock count child. |
| activateStockCount | This activates are starts the stock counting process including taking a snapshot of current inventory positions. |
| completeStockCountChild | Completes the counting or recounting of a stock count child, depending on which phase the stock count is in. This process will calculate discrepancies and move the child to the next phase. |
| updateCountQuantities | Updates the counted or recounted quantity fields for a stock count child based on the current phase of the stock count. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the stock count. |

**Standard Usage**

The stock count web services are design primarily to export information for third-party counting. You first lookup the headers, choose your head, retrieve all the details for the stock count, which does not contain item information but does contain a list of child count references. You can use these references to grab the full details of a child count which includes items and quantities.

# Store

The following operations are available within the Store web service.

| Operation | Description |
| --- | --- |
| lookupAutoReceiveStore | Retrieves all stores that allow auto-receiving of inventory from the input store. |
| lookupAssociatedStore | Retrieves all stores that are associated to the input store. They are sometimes called buddy stores. |
| lookupStoresInTransferZone | Retrieves all stores in the same transfer zone as the input store. |
| readStoreDetail | Retrieves the detailed information about a single store from the input unique reference. |

**Standard Usage**

The Store web service is used to retrieve information about stores. There are no updates. They are used to determine such information as whether you can ship to certain stores (such as those in transfer zones).

# StoreFulfillmentOrder

The following operations are available within the StoreFulfillmentOrder web service.

| Operation | Description |
| --- | --- |
| lookuFulfillmentOrdersHeaders | Retrieves summary information for fulfillment orders that match the search criteria input. |
| readFulfillmentOrderDetail | Reads the complete detailed information about a fulfillment order including items and quantities. |
| createFulfillmentOrderDetail | Creates a new fulfillment order with detailed information, including items and quantities. |
| cancelFulfillmentOrderDetail | Cancels quantities on a fulfillment order. This may cancel the entire order or just reduce or cancel quantities for specific items |
| rejectFulfillmentOrder | Rejects the fulfillment order indicating that the store will be unable to fulfill that particular order |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the fulfillment order. |

**Standard Usage**

Unlike some of the other web services, fulfillment order is not managed within EICS. Instead, EICs manages the picking and delivery, but the order itself is managed by an external order management system.

Oracle Retail Order Broker (OB) calls SIOCS for inventory availability.

Web services are supplied to find and read the details of a fulfillment order, but updates are not allowed. Instead, the external system uses `createFulfillmentOrderDetail` to notify EICS of a new order to ship, `cancelFulfillmentOrderDetail` to reduce or cancel quantities (note that they cannot be increased) or call `rejectFulfillmentOrder` to notify EICS that the order has been rejected.

## StoreInventory

The following operations are available within the StoreInventory web service.

| Operation | Description |
| --- | --- |
| lookupAvailableInventory | Retrieves basic availability information for multiple items at multiple locations. Only transaction-levels items are processed (UPCs are not allowed) and only current inventory is returned. The service supports up to 200 items at 150 locations. |
| lookupAvailableInventoryAll Stores | Retrieves basic availability information for a single item at all store locations. Only transaction-levels items are processed (UPCs are not allowed) and only current inventory is returned. |
| lookupInventoryInStore | Retrieves a broad set of inventory information for several items at several stores, broken down into various inventory groupings. |
| lookupInventoryInTransferZo ne | Retrieves a broad set of inventory information for items within the specific transfer zone, broken down into various inventory groupings. |
| lookupInventoryForBuddySt ores | Retrieves a broad set of inventory information for associated or buddy stores, broken down into various inventory groupings. |
| lookupFutureInventory | Retrieves the future inventory information (such as inbound, ordered quantities and expected dates) for an item and store location. |
| lookupAvailableInventoryAll Warehouses | Retrieves inventory information for a single item at multiple warehouses. Only transaction-level items are processed, and only current inventory is returned. |

**Standard Usage**

The StoreInventory is meant to retrieve inventory position information. Available inventory lookups are much smaller and quicker to respond than full inventory lookups. Future inventory is separated from current positions as it is much more time consuming to retrieve. Those who access the web services should consider the purpose before choosing which operation to use.

## StoreInventoryISN

The following operations are available within the StoreInventoryISN web service.

| Operation | Description |
| --- | --- |
| lookupIsnTypes | Returns a complete list of Item Scan Number types. |
| lookupIsn | Returns details about matching Item Scan Numbers in store inventory. |

| Operation | Description |
| --- | --- |
| createIsn | Create a new Item Scan Number without changing store inventory. |
| updateIsn | Updates an existing Item Scan Number without changing store inventory. |
| deleteIsn | Deletes an Item Scan Number without changing store inventory. |
| lookupCustomAttributeAdmins | Retrieves all the custom attribute admins configured for ISNs. |

**Standard Usage**

This web service is used to create, update or delete ISN in store inventory. An item scan number is any number meant to be scanned to find an item, and potentially a Unique Identification Number, that is not already an item, UPC, UIN, VPN, or other value. Items Scan Numbers are only used to find information and are not tracked as inventory.

## StoreInventoryUIN

The following operations are available within the StoreInventoryUIN web service.

| Operation | Description |
| --- | --- |
| createUIN | Create a new UIN without changing store inventory. |
| generateUIN | Generate new UINs without changing store inventory. |
| lookupUINDetails | Retrieves a collection of UINs matching a set of search criteria input. |
| readUINDetail | Returns details about a UIN in store inventory. A UIN reference is not unique, so this may return detailed information for UINs across multiple items. |
| updateUIN | Updates an existing UIN without changing store inventory. |

**Standard Usage**

This web service is used to create, generate, update or read UIN in store inventory.

## StoreItem

The following operations are available within the StoreItem web service.

| Operation | Description |
| --- | --- |
| lookupItemHeaderByItem | Retrieves list of summary information about an item that match the item-based search criteria input. |
| lookupItemHeaderBySource | Retrieves list of summary information about an item that match the source or location-based search criteria input. |
| lookupItemHeaderByUDA | Retrieves list of summary information about an item that match the UDA (User Defined Attribute)-based search criteria input. |
| lookupItemHeaderByInventory | Retrieves list of summary information about an item that match the inventory-based search criteria input. |
| lookupItemCfa | Retrieve a list of custom flexible attributes for the specified item and store. |

| Operation | Description |
| --- | --- |
| lookupItemUda | Retrieve a list of user defined attributes for the specified item and store. |
| readItemDetail | Retrieves the complete detailed information a single item based on its unique reference. |
| lookupRelatedItem | Retrieves a list of summary information about items related to the item used as input criteria. |
| saveItemImage | Inserts a new display image or QR code image for the specified item. The service returns immediately, and the information is processed asynchronously. |

**Standard Usage**

This web service is used to find items and retrieve information about items. The only exception is the ability to create new image-based information about an item.

# StoreItemPrice

The following operations are available within the StoreItemPrice web service.

| Operation | Description |
| --- | --- |
| lookupItemPriceHeader | Retrieve a summary list of item price information based on input criteria. This only retrieves information known to EICS and has no access to a pricing system. |
| readItemPrice | Retrieves the full details a single item price record based on its unique reference. |
| lookupItemPriceOnEffective Date | Retrieves the item price of an item for a specific date. |

**Standard Usage**

This web service is used to retrieve information about prices that are known to EICS. Integration with pricing systems updates EICS information about item prices on a continual basis. These web services give a view into EICS information only.

# StoreNotification

The following operations are available within the StoreNotification web service.

| Operation | Description |
| --- | --- |
| createNotification | Creates a new notification within the system. These notifications are displayed in the client applications. |

**Standard Usage**

This web service is designed for external system that handle related activities to EICS. With this web service, they can send notifications into EICS of activity that needs to take place based on something that has occurred in another system.

## StoreShipmentManifest

The following operations are available within the StoreShipmentManifest web service.

| Operation | Description |
| --- | --- |
| closeManifest | Closes the manifest shipments. |

**Standard Usage**

This web service is designed to close manifest shipments. All manifest shipments matching the input criteria like carrier code, carrier service code will be closed.

## StoreShipmentReason

The following operations are available within the StoreShipmentReason web service.

| Operation | Description |
| --- | --- |
| lookupAllShipmentReasons | Retrieves all the shipment reasons configured for store shipments. |

**Standard Usage**

This web service exists to allow customers to retrieve information about shipment reasons that can be assigned to line items on outgoing shipments. The shipment based web services taking the code identifier and thus, you will need to read in these shipment reasons to be able to select and apply valid reason codes.

## StoreTicket

The following operations are available within the StoreTicket web service.

| Operation | Description |
| --- | --- |
| createTickets | Create a new group of up to 999 tickets to be managed and printed. |
| lookupTicketFormats | Retrieves available ticket formats for the criteria specified. |

**Standard Usage**

The `createTickets` operation is used to create a new group up to 999 tickets to be managed and printed. The ticket formats can be retrieved using `lookupTicketFormats` operation based on the criteria specified.

## StoreTransfer

The following operations are available within the StoreTransfer web service.

| Operation | Description |
| --- | --- |
| lookupTransferHeader | Retrieve a summary list of transfers that matches the input criteria. |
| lookupTransferContext | Retrieves all the transfer context options available to assign to a transfer. |

| Operation | Description |
|---|---|
| readTransfer | Retrieves the detailed information about transfer, including its items and quantities, based on a unique reference. |
| createTransferRequest | Creates a brand-new transfer request (Location 1 requesting a transfer from Location 2). |
| saveTransferRequest | Updates a transfer request allowing user to change items and quantities. This must be done prior to requesting it, which finalizes the transfer request. This requires an activity lock. |
| createTransfer | Generates a new transfer that you can add details to. The saveTransfer method must be used to update details such as items and quantities of the transfer. |
| saveTransfer | Updates a previously approved transfer item and quantity details. This operation requires an activity lock. |
| saveTransferApproval | Updates items and quantities on a transfer in requested status that is currently in the process of being approved but has not yet been approved. This operation requires an activity lock. |
| requestTransfer | Updates the status to Requested, finally the transfer request. This allows the opposite location to view the new request for transfer of goods. This operation requires an activity lock. |
| approveTransfer | Approves a transfer request converted the transfer request into an approved transfer. This operation requires having an activity lock. |
| rejectTransfer | Rejects a transfer in request status which prevents the transfer request from becoming a transfer. This operation requires having an activity lock. |
| cancelTransfer | Cancels an approved transfer. This operation requires having an activity lock. |
| closeTransfer | Closes a processed or partially processed transfer finalizing the state of the transfer. This operation requires having an activity lock. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the transfer. |

**Standard Usage**

The process is started by one store creating a transfer request from a shipping store using `createTransferRequest`. The requesting store can continue modifying the transfer request using `aveTransferRequest` until it is ready to notify the shipping store, when it then uses the `requestTransfer` to the shipping store. The shipping store can then begin picking items for the transfer and updating the transfer using the `saveTransferApproval` operation. When all the quantities the shipping store are willing to ship are determined, the shipping store uses `approveTransfer` to finalize the approval of the transfer. Alternatively, they can choose to reject the transfer using `rejectTransfer`. It is possible for a shipping store to create a transfer document without going through the request and approval process by using `createTransfer` and `saveTransfer`.

# TransferDelivery

The following operations are available within the TransferDelivery web service.

| Operation | Description |
|---|---|
| lookupTransferDeliveryHeaders | Retrieves basic information about one or more transfer deliveries that match the criteria specified. This operation is used to find a delivery arriving at the store. |
| readTransferDeliveryDetail | Retrieves the entire set of information about a transfer delivery header based on the identifier you pass to it. |
| updateTransferDelivery | Updates the header information on a transfer delivery. This operation requires an activity lock. |
| receiveTransferDelivery | Receives all the currently open and active containers on a transfer delivery by defaulting quantities into all the unreceived items. This does not move inventory, only defaults quantities. This operation requires an activity lock. |
| confirmTransferDelivery | Confirms a transfer delivery receiving the goods into inventory and updating all the inventory positions. This moves the transfer delivery to a completed status. This operation requires an activity lock. |
| lookupTransferDeliveryContainerHeaders | Retrieves summary information about every container on a transfer delivery based on the unique delivery reference. |
| readTransferDeliveryContainerDetail | Reads the entire details of a container including items and quantities based on a unique container reference. |
| createTransferDeliveryContainer | Generates a new container on the transfer delivery and returns a reference to use so that items and quantity can be added later. |
| updateTransferDeliveryContainer | Updates the items and quantities on a transfer delivery container. This operation requires an activity lock. |
| receiveandConfirmTransferDeliveryContainer | It first defaults receiving quantity on the items within the container and then executes the same locking as the confirmTransferDeliveryContainer. This operation requires an activity lock. |
| confirmTransferDeliveryContainer | Confirms a transfer delivery container as received and updates all the inventory positions. This operation requires an activity lock. |
| cancelTranferDeliveryContainer | Cancels a transfer delivery container moving it to missing status. Changes cannot be made to a canceled container. |
| openTransferDeliveryContainer | Re-opens an already confirmed container moving it back into in-progress status. |
| lookupTransferDeliveryOrders | Retrieves any customer orders associated with the transfer delivery based on the delivery's unique reference. |
| lookupMisdirected TransferDeliveryContainers | Retrieves summary information about containers that may have been misdirected based on a set of search criteria as input into the operation. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the transfer delivery. |

**Standard Usage**

After reading a transfer delivery using `lookupTransferDeliveryHeader`, you can read the header detail with `readTransferDelivery` or container list with `lookupTransferDeliveryContainers`. You can then use `updateTransferDelivery` to update header attributes and `updateTransferDeliveryContainer` to update items and quantities in the container. To quickly receive the quantities, `receiveTransferDeliveryContainer` automatically fills in quantities, and when quantities are entered

`confirmTransferDeliveryContainer` finalizes the container (and if appropriate configurations and business rules apply) immediately updates the inventory. If `receiveTransferDelivery` or `confirmTransferDelivery` is used, then all containers will either be received or confirmed respectively.

## TransferShipment

The following operations are available within the TransferShipment web service.

| Operation | Description |
| --- | --- |
| lookupTransferShipmentHeader | Retrieves basic information about one or more transfer shipments that match the criteria specified. This operation is used to find a shipment. |
| readTransferShipmentDetail | Retrieves the entire set of information about a transfer shipment header based on a unique reference. |
| createTransferShipment | Creates a new and empty transfer shipment and returns a reference to the shipment. |
| saveTransferShipment | Updates the information on a transfer shipment header. |
| submitTransferShipment | Submits the transfer shipment for review before final dispatch. |
| cancelSubmittedTransferShipment | Cancels the submission of the transfer shipment for review. |
| dispatchTransferShipment | Dispatches a transfer shipment. This moves the shipment to dispatched state and updates the inventory. A transfer shipment cannot be modified after dispatch. Dispatch should occur only after all containers are confirmed. |
| cancelTransferShipment | Cancels a transfer shipment. |
| lookupTransferShipmentContainer | Finds all the containers on a specific shipment and retrieves basic identification information about each container. |
| readTransferShipmentContainer | Reads the specific and complete contents of a container. |
| createTransferShipmentContainer | Creates a new transfer shipment container on the shipment and returns a reference to it. |
| saveTransferShipmentContainer | Updates the information about a transfer shipment container including adding and removing items and quantities. |
| confirmTransferShipmentContainer | Confirms that a transfer shipment container is ready for shipment and marks the container as no longer editable. |
| cancelTransferShipmentContainer | Cancels a transfer shipment container on the shipment. |
| openTransferShipmentContainer | Re-opens a confirmed container on a shipment prior to the shipment being dispatched so that changes can be made to the container. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the transfer shipment. |

**Standard Usage**

To create a shipment for a transfer document, you look up the transfer shipment using `lookupTransferShipmentHeader`. If it does not exist, you could create one for the document using `createTransferShipment`. You then create a container on the shipment using `createTransferShipmentContainer`. You can update the container

with items and quantities using `saveTransferShipmentContainer`. You then confirm the container using `confirmTransferShipmentContainer`. Repeat the process for each container as needed. Once all containers are confirmed, you submit if configured using `submitTransferShipment` and finally dispatch the shipment using `dispatchTransferShipment`. Dispatching the shipment finalizes the shipment and relieves the inventory.

## VendorDelivery

The following operations are available within the VendorDelivery web service.

| Operation | Description |
| --- | --- |
| lookupVendorDeliveryHeaders | Retrieves basic information about one or more vendor deliveries that match the criteria specified. This operation is used to find a delivery from a supplier. |
| lookupPurchaseOrderHeaders | Retrieves basic information about one or more purchase orders that match the criteria specified. |
| readVendorDeliveryDetail | Retrieves the entire set of information about a vendor delivery header based on a unique reference. |
| createVendorDelivery | Generate a new vendor delivery heaver and returns a referenced to the delivery. |
| updateVendorDelivery | Updates the information on a vendor delivery header. This does not include containers, items, or quantities. This operation requires an activity lock. |
| receiveVendorDelivery | Updates the quantities on a vendor delivery filling in any unreceived items within the containers of the delivery with a default value. It "receives" missing quantities, but no inventory positions are updated. This operation requires an activity lock. |
| confirmVendorDelivery | Confirms the vendor delivery updating inventory positions and completing the delivery. This operation requires an activity lock. |
| rejectVendorDelivery | Rejects the vendor delivery placing it in rejected status. This operation requires an activity lock. |
| cancelVendorDelivery | Cancels the vendor delivery placing it in canceled status. This operation requires an activity lock. |
| lookupVendorDeliveryContainerHeaders | Retrieves summary information about every container on a vendor delivery based on the unique delivery reference. |
| readVendorDeliveryContainerDetail | Reads the entire details of a container including items and quantities based on a unique container reference. |
| createVendorDeliveryContainer | Generates a new container on the vendor delivery and returns a reference to use so that items and quantity can be added later. |
| updateVendorDeliveryContainer | Updates the items and quantities on a vendor delivery container. This operation requires an activity lock. |
| confirmVendorDeliveryContainer | Confirms a vendor delivery container as received and updates all the inventory positions. This operation requires an activity lock. |
| cancelVendorDeliveryContainer | Cancels a vendor delivery container moving it to missing status. Changes cannot be made to a canceled container. |
| openVendorDeliveryContainer | Open Vendor delivery container. This will re-open a container after receipt allowing it to be received again. |
| lookupVendorDeliveryOrders | Retrieves any customer orders associated with the vendor delivery based on the delivery's unique reference. |

| Operation | Description |
|---|---|
| lookupVendorDeliveryAdjustments | Retrieves any external receipt adjustments that exist for the delivery based on the specified unique reference. |
| cancelSubmitVendorDeliveryContainer | Opens a submitted container for further updates, moving the status to in-progress. |
| submitVendorDeliveryContainer | Moves the status of the container to submitted and prevents further updates. The container may still be confirmed. No inventory positions are updated via this operation. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the vendor delivery. |

**Standard Usage**

After reading a vendor delivery using `lookupVendorDeliveryHeader`, you can read the header detail with `readVendorDelivery` or container list with `lookupVendorDeliveryContainers`. You can then use `updateVendorDelivery` to update header attributes and `updateVendorDeliveryContainer` to update items and quantities in the container. To quickly receive the quantities, `receiveVendorDeliveryContainer` automatically fills in quantities, and when quantities are entered `confirmVendorDeliveryContainer` finalizes the container (and if appropriate configurations and business rules apply) immediately updates the inventory. If `receiveVendorDelivery` or `confirmVendorDelivery` is used, then all containers will either be received or confirmed respectively. You can re-open container using `openVendorDeliveryContainer`. To prevent further updates to the container can use `submitVendorDeliveryContainer`. Submitted container can be re-opened and moved to in-progress status for further updates using `cancelSubmitVendorDeliveryContainer`.

# VendorReturn

The following operations are available within the VendorReturn web service.

| Operation | Description |
|---|---|
| lookupVendorReturnHeader | Retrieves basic information about one or more vendor return documents that match the criteria specified. |
| readVendorReturnDetail | Retrieves the entire set of information about a vendor return, including items and quantities, based on a unique reference. |
| saveVendorReturn | Updates the entire set of information about a vendor return, including items and quantities. This operation requires an activity lock. |
| approveVendorReturn | This marks an in-progress vendor return as approve for shipment. This operation requires an activity lock. |
| cancelVendorReturn | Cancels a vendor return indicating no further items and quantities should be shipped for the return. |
| closeVendorReturn | Closes a vendor return document moving it from in-progress to canceled, rejected, or complete status depending on the state of shipped quantities. |

| Operation | Description |
|---|---|
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the vendor return. |

**Standard Usage**

The user may access `lookupVendorReturnHeader` to find vendor returns to deal with. Once the proper vendor return is found, `readVendorReturnDetail` will retrieve all the details of the vendor return including items and quantities. The `saveVendorReturn` operation is then used to update quantities that are expected to ship. Once the vendor return reaches its final state, the operation `approveVendorReturn` will approve the return and get it ready for shipment.

# VendorShipment

The following operations are available within the VendorShipment web service.

| Operation | Description |
|---|---|
| lookupVendorShipmentHeaders | Retrieves basic information about one or more vendor shipment headers that match the criteria specified. |
| lookupReturnContext | Retrieves all the context options that are available to assign to a vendor return shipment. |
| readVendorShipmentDetail | Retrieves the detailed information about a vendor return header based on a unique reference. It does not include information about containers or items. |
| saveVendorShipment | Creates a new vendor shipment header if not identifying reference is set or updates the vendor shipment header information if a unique reference is sent as part of the date. When used as an update, an activity lock is needed. |
| submitVendorShipment | Submits the vendor shipment for review before final dispatch. |
| cancelVendorShipmentSubmission | Cancels the submission of the vendor shipment for review. |
| cancelVendorShipment | Cancels a vendor shipment. This moves the shipment to canceled status. Changes cannot be made to a canceled shipment. |
| dispatchVendorShipment | Dispatches a vendor shipment. This moves the shipment to dispatched state and updates the inventory. A vendor shipment cannot be modified after dispatch. Dispatch should occur only after all containers are confirmed. This operation requires an activity lock. |
| closeVendorShipment | Closes a vendor shipment using business logic to determine its final state. It cancels the shipment of remaining quantities. Changes cannot be made after a shipment is closed. |
| lookupVendorShipmentContainerHeaders | Retrieves summary information about all containers within a vendor shipment based on the unique reference of the shipment. |
| readVendorShipment ContainerDetail | Reads the specific details, including items and quantities, about a container specified by its unique reference. |
| saveVendorShipmentContainer | Update the details of a container, including items and quantities. This operation requires an activity lock. |
| confirmVendorShipmentContainer | Confirms that the container is ready for shipment. A confirmed container cannot be modified. This operation requires an activity lock. |

| Operation | Description |
|---|---|
| cancelVendorShipmentContainer | Cancels a container on the shipment removing it from the shipment. |
| openVendorShipmentContainer | Opens a confirmed container placing it back into in-progress status so that items can be added or removed from the container. |
| lookupCustomAttributeAdmins | Retrieves the custom attribute administration information that describes what customized attributes are available on the vendor shipment. |

**Standard Usage**

To create a shipment for a vendor return document, you look up the vendor shipment using `lookupVendorShipmentHeader`. If it does not exist, you could create one using `createVendorShipment`. You then create a container on the shipment using `createVendorShipmentContainer`. You can update the container with items and quantities using `saveVendorShipmentContainer`. You then confirm the container using `confirmVendorShipmentContainer`. Repeat the process for each container as needed. Once all containers are confirmed, you submit if configured using `submitVendorShipment` and finally dispatch the shipment using `dispatchVendorShipment`. Dispatching the shipment finalizes the shipment and relieves the inventory.

# Enterprise Documentation

Full web service API documentation can be found at:

https://docs.oracle.com/cd/E82085_01/160/RIB%20Integration%20Guide/Output/ServiceTOC.html

# Sales Integration

EICS integrates with POS systems and Sales Audit systems to ensure that the inventory positions are accurate. This is especially important where accurate up-to-date inventory positions are required to reduce customer disappointment when trying to locate items that appear in inventory or delays in filling customer orders.

POS is the primary source of sales, returns, void, and some customer order transaction information to EICS.

ReSA sends only modified or new POS transaction records to EICS.

POS systems integrated with EICS can do the transaction notifications using a web service.

Sales Audit systems can only communicate through a file import process.

**Figure 8-12    POS and Sales Audit Integration**
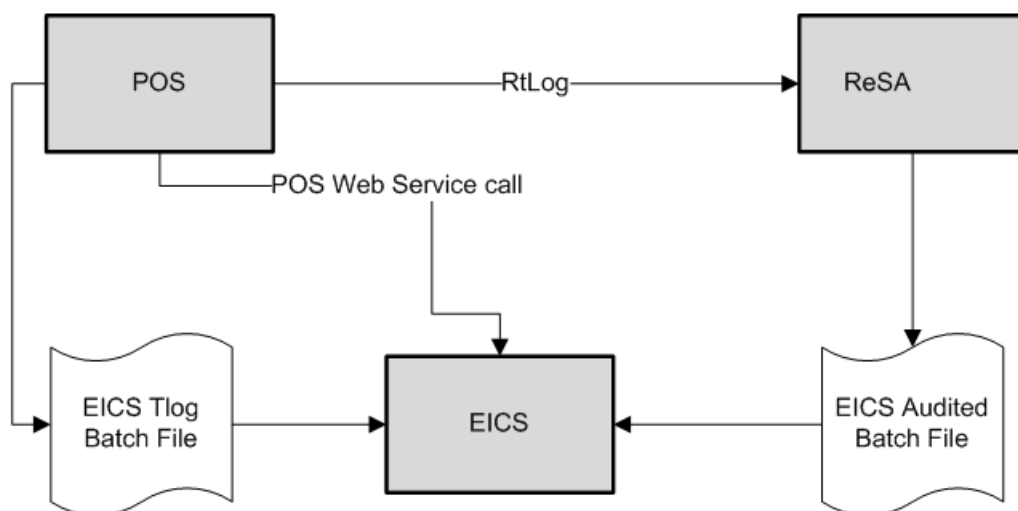


The following features are part of this integration:

- Real-time web service integration
- Batch integration
- Audited sales data integration
- Automatic disposition processing for returns

Batch processing and ReSA processing are discussed elsewhere as are the store and system configurations that might determine how the sale is processes.

## POS and Sales Audit Process Flow

The following figure shows how a POS, Retail Sales Audit, and EICS are integrated. A POS generates an RTLog containing all the POS transactions and sends it to the Oracle Retail Sales Audit system (ReSA). ReSA sends the audited modified or new transactions to EICS. ReSA also sends the POS transaction upload file to merchandising to update inventory.

Please note that Oracle Retail Xstore is interfaced with EICS to update the inventory transactions near real time only through web service. It does not use batch.

Non-Oracle POS systems can use a batch to import transactions directly into EICS. EICS also processes the POS transactions that have been changed or entered the sales audit system and updates the inventory based on the delta.

**Figure 8-13    POS and Sales Audit Process Flow**
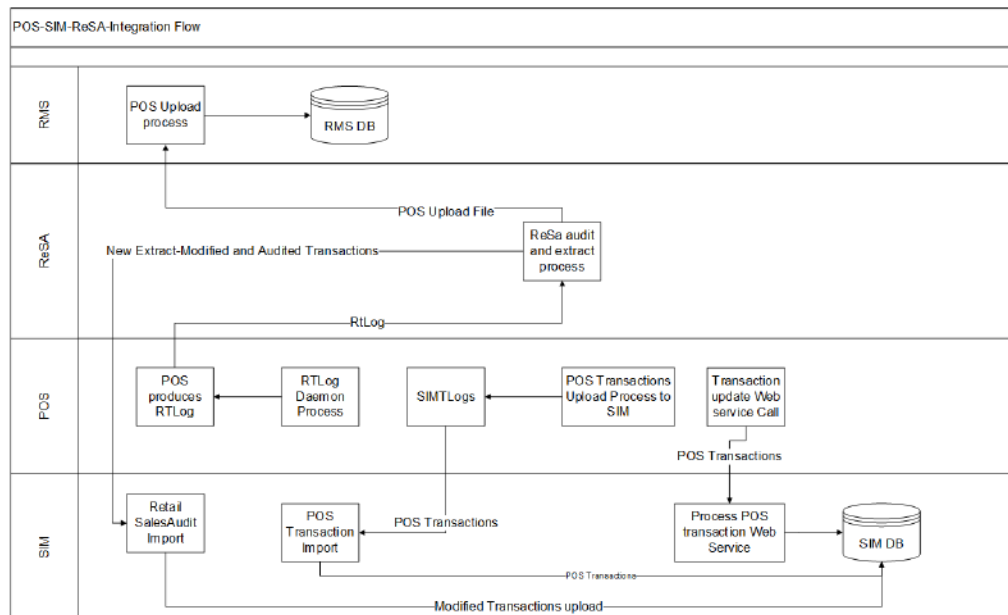


There are two reasons for POS to send sales data directly to EICS and not to the auditing system:

- Real-time inventory updates to support Commerce Anywhere are critical. A possible round trip from POS to ReSA to EICS takes too long in the dynamic inventory environment of today.

- POS is the application that owns sales data and ReSA owns audited data. Architecturally, it makes more sense to have data supplied by the owner of that data. POS sends sales data and ReSA sends audit changes to EICS.

## Sales and Return Processing

As part of the sales processing, EICS updates the inventory depending on the nature of the transaction. The following are the supported transaction types for the sales processing: Sale, Return, and Post Void of these transactions. The audit system should not modify the post void transactions. A change to a void is not supported by EICS.

## Customer Order Processing

In EICS, the Retail Sales Audit import process, POS Transaction import process, and POS Transaction web service process support the following types of customer orders.

- For layaway and on hold, EICS supports create, update, cancel, and pickup/delivery. For external web order type, only pickup transactions performed in POS are sent to EICS.

- Pickup transactions, both in-store and external, cannot be voided or modified by sales audit and if these transactions are modified by sales audit system, EICS just drops the transaction and does not process.

> ✏️ **Note:**
>
> Current Xstore functionality is limited to only layaway and on hold orders. Web order processing is not supported in this release.

## Item Disposition

POS can move inventory for return and post void transactions to 'unavailable' or 'out of stock'. This is especially useful in some environments where items returned must be disposed of or have to be reprocessed.

The external sale transaction coming into EICS may include a reason code that is mapped to the inventory adjustment reason codes in EICS. Point of Service maps the EICS reason codes, and the reason codes are sent to EICS in the web service or file extract for the return and post void transactions. EICS first processes the return or post void and updates stock on hand. Next, if the reason code exists, EICS checks this reason code with the one in inventory adjustment reason code table. If a valid match is found, EICS generates an inventory adjustment to notify external systems and execute the disposition instructions tied to the inventory adjustment reason code. Based on the disposition mapped to the reason code, EICS moves the returned inventory to not for sale or out of stock and updates the history trail. If sub-buckets are used, they are also updated if the movement is to not for sale.

If the reason code received is invalid/not present/mapped incorrectly, the system writes an error log and continues to process the stock on hand part of the transaction.

## Drop Ship

When the sales records indicate the record is a drop ship, EICS does not perform any processing of this record since the drop ship process implies the inventory is shipped from a third-party location and not from the store.

## Item Types

EICS only processes SKU or UPC numbers. GS1 databars, or any other smart barcodes such as VPLUs or Type-E barcodes, should have been extracted to their SKU or UPC number by the POS system.

In addition, EICS only updates inventory for stock holding items. Non-inventory items do not update any stock on hand and are not processed.

Items with the store pack inventory indicator turned off are automatically broken down and the inventory of the component items is updated.

## RFID

If the point-of-sale record for an item includes an RFID tag, the tag will be moved to a SOLD status indicating it should be out-of-store.

# Integration with Manifesting Systems

For access to an external manifesting system to take place, the customer must first setup Carrier Type as "Third Party" and the Carrier Service (Manifest Type) must be Parcel (P).

Configuration controls whether manifesting is done for a transfer to store, finisher, or warehouse. In addition, configuration controls manifesting for a return to vendor shipment or a customer order delivery.

Carrier services with manifest type of "O" (Other) and "H" (Home Fleet) do not go through the manifesting system. When Manifest Type is "O," EICS prompts the user to enter the carrier address where the shipment is to be sent for fulfillment. Manifest Type of "H" is within the company and therefore, does not prompt the user for an address.

Some carriers require weight, dimension, or both values to be sent in the manifest payload. If so, the carrier's service should have either the weight indicator or carton dimension indicate set to active (or both) during their carrier service setup.

EICS supplies an outbound and inbound Shipment Manifest SOAP web service. The following are supported service operations:

A web service is used to send all the shipment information to the external manifesting system and to receive close shipment requests from external systems.

A web service accepts requests from external systems to close shipments. It is used to find those "Submitted" shipments for the provided tracking ID, carrier, service and date, and dispatch those shipments.

> **Note:**
>
> EICS supplies a WSDL and XSD that defines the web service, operation, and data content. This web service will need to be implemented either for the manifesting system or a plug-in set up.

**ShipmentManifestService**

This web service notifies an external manifesting system that a manifest needs to be created.

| Service Operation | Description |
| --- | --- |
| createManifest | Requests the external manifesting system to create a new parcel manifest for an input transaction. |

**StoreShipmentManifestService**

This web service receives a message from an external manifesting system that the items on the manifest have been picked up.

| Service Operation | Description |
| --- | --- |
| closeManifest | Instructs EICS that submitted shipments have been picked up by the carrier. |

# Customer Order

### CustomerOrderAddressService

When shipping to customer during the fulfillment order workflow, EICS retrieves the address for the order delivery from an external order managements system. When viewing delivery address information within the client application, it also retrieves it from an external system. The web service is defined to connect to an OrderManagementService.

| Service Operation | Description |
| --- | --- |
| queryCustomerOrderAddress | Retrieves detailed address information for the order and customer information passed to it. |

### CustomerOrderService

This service connects to OrderManagementService to manage customer orders. It includes operations to create a customer order, query for customer orders, pickup/cancel items from a customer order and return items from customer orders.

| Service Operation | Description |
| --- | --- |
| requestNewCustomerOrderId | Requests new customer order Id. |
| cancelNewCustomerOrderId | Cancels the new customer order id. |
| createCustomerOrder | Creates customer order. |
| queryCustomerOrder | Queries the customer order present in the system. |
| PickupCustomerOrderItems | Pickup items from the customer order. |
| ReturnCustomerOrderItems | Returns items from the customer order. |
| UpdateReceipt | Updates the receipt of customer order. |

# Integration for Ticket Printing

### TicketPrintService

When printing tickets, EICS sends ticket information to an external system for printing. This web service needs to be implemented for printing tickets to a physical printer.

| Service Operation | Description |
| --- | --- |
| printTickets | Sends item tickets to an external system to be printed. It must be implemented by the external system in order to receive the tickets. |

# Integration for Sales Forecast

### SalesForecastService

If implemented by a third-party sales analytic system, EICS can retrieve sales forecasting information from this system. It retrieves the specific date and quantity for that date extending out to up to 30 days.

| Service Operation | Description |
| --- | --- |
| retrieveSalesForecast | Retrieves the sales forecast data for a store and item over the next 30 days. |

# Integration for Store Order

**OrderApproveNotificationService**

When store order is approved, EICS sends notification to a third-party item management system.

This notification will be sent out for store orders that are created manually or system generated.

It is not applicable to store orders created by external system.

| Service Operation | Description |
| --- | --- |
| orderRequestApproved | Sends notification to external item management system that the order request is approved. |

**StoreExtNotificationService**

When store order with external ID is approved, EICS sends notification to the external system.

This service is applicable only for externally created store orders.

| Service Operation | Description |
| --- | --- |
| createNotification | Sends notification to external system on approving the externally created store orders with its items information. |

# Retail Home Integration

EICS now supports following integration scenarios with Retail Home:

- Launch SIOCS web client from Retail Home
- Launch SIOCS favorites from Retail Home
- Display a tile report for items that are out of stock on shop floor
- Display a tile report for stock counts that are pending authorization
- Launch detailed operational views in SIOCS web client from related tile reports in Retail Home

# Launch SIOCS from Retail Home

Launching SIOCS client requires an entry to be made under the application navigator section of Retail Home. It enables the user to launch SIOCS web client in a new browser tab from within Retail Home. Please refer to *Oracle Retail Home*

*Administration Guide* for information on how to work with application navigator in Retail Home.

The SIOCS application configuration should look like this:

**Figure 8-14    Add Application Info**



- **Seeded**: Disabled and set to No.
- **Application Navigator**: Enable it to launch SIOCS client from Retail Home.
- **Application Name**: The name of the application that is, Store Inventory Operations Cloud.
- **Color Set**: Any color that you want to allocate to SIOCS.
- **Application Code**: Select SIOCS from the drop down.

- **Application Link**: The URL of SIOCS web client.
- **Platform Service**: Enable it to use Favorites feature.
  - **URL**: The base URL of the platform services. The URL would be of the form

    https://<SIOCS-HOST>/RetailAppsPlatformServices

    <SIOCS-HOST> is the same host in Application Link.
  - **Supported Features**: Check only the favorites feature.

The user needs to be part of RETAIL_HOME_ADMIN security group in order to access Application Navigator in Retail Home.

## Tile Reports

EICS supports following two types of two metric reports:

- Shop Floor Out of Stock Items
- Stock Counts - Ready to Authorize

Adding an application navigator entry for SIOCS will automatically configure EICS tiles on Retail Home.

The data seed features do the following:

1. Creates a custom report for EICS tiles on Retail Home.

2. Creates two tiles from the custom report and maps them to retail_home_users IDCS or OCI IAM application role.

3. The data seed features will also configure tile states for the two tiles and hook them up with EICS end points.

After all the configuration, you should be able to see EICS tiles on the dashboard. They should look like the ones below:

**Figure 8-15    Example EICS Tiles**
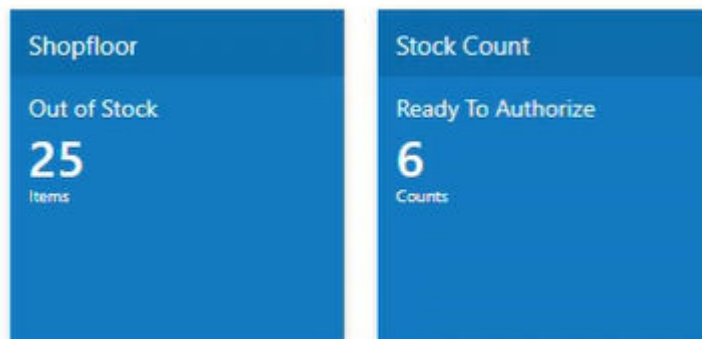


## EICS Endpoints

EICS exposes following two endpoints:

## Shop Floor Out of Stock Items

This endpoint can be used as a data source for **Shop floor Out of Stock** tile state.

The response contains information on number of items that are out of stock across all the stores that are accessible to the user.

If the percentage of out of stock items to total items is greater than the **Shopfloor Out of Stock Items Critical Percentage** system configuration, EICS marks the response as important which displays a '!' mark next to the number on the tile report.

**Table 8-5    Shop Floor Out of Stock**

| Endpoint | Operational View |
| --- | --- |
| https://<EICS-HOST>:<EICS -PORT>/sim-mobile/internal/rhreports/ outofstock/shopfloor/tile | Shopfloor Out of Stock |

## Stock Counts - Ready to Authorize

This endpoint can be used as a data source for **Stock Count - Ready to Authorize** tile state.

The response contains information on number of stock counts that are pending authorization across all stores that are accessible to the user.

**Table 8-6    Stock Counts - Ready to Authorize**

| Endpoint | Operational View |
| --- | --- |
| https://<EICS-HOST>:<EICS -PORT>/sim-mobile/internal/rhreports/ readytoauthorize/tile | Stock Count - Ready To Authorize |

The response payloads of both these endpoints confirm to the two metric payload specifications of Retail Home.

User should be a part of retail_home_users IDCS or OCI IAM application role to access these endpoints.

For convenience, EICS also provides a RETAIL HOME security role that captures security permissions required to access these operational views. The user still needs appropriate functional area permissions to navigate to transaction detail screens.

## SIOCS Operational Views

EICS has added following operational views that can be hooked with related tiles:

* **Shopfloor Out of Stock Items**

  This view gives a store and item level breakdown of the information that is displayed on the tile. The user can look at item level records for each store and navigate to the item detail screen for any store/item combination provided he or she has the required permissions.

  This view is available under Operations - Operational Views - Shopfloor Out of Stock menu.

- **Stock Count - Ready to Authorize**

  This view gives a store and stock count level breakdown of the information that is displayed on the tile. The user can look at stock count level records for each store and navigate to the stock count detail for any store/count combination provided he or she has the required permissions.

  This view is available under Operations - Operational Views - Stock Count - Ready to Authorize menu.

## Launch SIOCS Operational Views from Tile Report

Launching SIOCS operational views from related tile report requires the tile report to be configured with the URL of the related operational view. Once that is done, clicking on tile report header should open the related EICS operational view in a new browser tab.

# 9
# Configuration

This chapter describes how you can configure functionality usage.

Configuration can be achieved by adjusting:

1. System Admin Parameters
2. Store Admin Parameters
3. Feature Permissions

## System Admin Parameters

**Table 9-1    System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Allow Non-Range Item | Values: Yes/No<br>This parameter is to determine to give stores the ability to add non ranged items to functional areas in the application. | Yes | Admin | Boolean |
| Auto Default UIN Attributes | Auto default UIN attributes is used during ItemLocCre to default UIN attributes for an item taking values from the store defaults (STORE_UIN_ADMIN_DEPT table). It is also checked whenever we range an item to a store. | No | Admin | Boolean |
| Barcode Scan/ Entry Log - Receiving | Values: Yes/No<br>Yes: Captures, on the mobile, all container and item scans or manual entries by user, location, and time at the point of receiving deliveries. Containers are captured in the quick receiving dialog when item level is captured in the Item Receiving function of the Container Summary for DSD and Transfer Receiving.<br>No: Does not capture any container or item information in the mobile receiving dialogs. | Yes | Admin | Boolean |
| Default UOM | Values: Standard UOM / Cases<br>Standard UOM the system will default to the standard UOM until the UOM is changed. Cases the system will default to cases until the UOM is changed. | 2 | Admin | Integer |
| Disable Pack Size | Values: Yes/No<br>Yes: Pack size is disabled and cannot be changed.<br>No: Pack size is editable and can be updated. | No | Admin | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Enable Extended Attributes | Values: Yes/No<br><br>This is a master switch that will turn on and off the feature for collecting extended attributes.<br><br>If set to 'Yes', the system will capture the extended attributes when scanning a GS1 DataBar<br><br>If set to 'No', the Ext Attributes are not captured when scanning a GS1 Databar and the button/menu option will not be available on either the EICS or SOCS regardless of the store parameter. | No | Admin | Boolean |
| Enable Sub-buckets | Values: Yes/No<br><br>Yes: Sub-buckets will be used throughout the application.<br><br>No: Sub-buckets will not be used in the application. | Yes | Admin | Boolean |
| Item Description Translation Preference | Values: System/User<br><br>System: The Item description displayed in the application would be the item description from STORE_ITEM table (store description) or from ITEM table if the item is not ranged. This is existing functionality.<br><br>User: The Item description will be displayed in the user's preferred language if translation is available.<br><br>If the translation is not available in the user's preferred language, then the item description will be displayed in the store locale language, if it is available.<br><br>If the translation is not available in the store locale language, then the item description displayed will be item description from STORE_ITEM (store description). If the item is a non ranged item, then the item description displayed will be from the ITEM table. | System | Admin | Boolean |
| Maintain RFID History | Values: Yes/No<br><br>Yes: will create history records in the history table for every transaction occurred for the RFID Tag.<br><br>No: will not create the history records however the integration with the RFID solution and RFID tag tracking could still be on. | Yes | Admin | Boolean |
| Shopfloor Out of Stock Items Critical Percentage | Values: 0.01 - 100%<br><br>If the percentage of out of stock items on shop floor to total items is greater than this percentage, then there will be an '!' alert with the number of items out of stock on the tile report. | 0.01% | Admin | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| System Code | Code identifying the application for integration with an external system. This often is a company ID. This is only used for outbound integration. | INV | Admin | String |
| Audit Direct Store Delivery | Audit Records are log of activities and usage information in the system. This parameter is to determine whether activity records for actions (confirm/submit/update and so on) performed on vendor delivery and vendor delivery carton will be created. | Yes | Audit | Boolean |
| Audit Publish Message | Audit Records are log of activities and usage information in the system. This parameter is to determine whether activity records for outbound messages will be created. | Yes | Audit | Boolean |
| Audit Receive Message | Audit Records are log of activities and usage information in the system. This parameter is to determine whether activity records for inbound messages will be created. | Yes | Audit | Boolean |
| Audit RTV Update | Values: Yes/No<br><br>Yes: Enables activity logging for rtv request and rtv shipments.<br><br>No: Disables activity logging for rtv request and rtv shipments. | Yes | Audit | Boolean |
| Audit Security | Values: Yes/No<br><br>Yes: Enables activity logging for security events. It includes login success/failure, security management changes (roles, user assignments, and so on).<br><br>No: Disables activity logging for security events. | Yes | Audit | Boolean |
| Audit Stock Count Completed | Audit Records are log of activities and usage information in the system. This parameter is to determine whether activity records will be created for count or recount complete for stock count child. | Yes | Audit | Boolean |
| Audit Transfer Dispatch | Values: Yes/No<br><br>Yes: Enables activity logging for transfer shipments.<br><br>No: Disables activity logging for transfer shipments. | Yes | Audit | Boolean |
| Audit Transfer Receiving | Values: Yes/No<br><br>Yes: Enables activity logging for transfer receiving.<br><br>No: Disables activity logging for transfer receiving. | Yes | Audit | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Audit Transfer Update | Values: Yes/No<br><br>Yes: Enables activity logging for transfer requests.<br><br>No: Disables activity logging for transfer requests. | Yes | Audit | Boolean |
| Batch max files per job run | Batch max files per job run. | 20 | Batch | Integer |
| Batch Maximum Threads | Gives the maximum number of batch partitions for a batch to run. | 5 | Batch | Integer |
| Batch Scheduler Execution Interval | Gives the batch scheduler execution delay interval duration to throttle batch execution calls. | 300 | Batch | Integer |
| Data seed by store | Values: Yes/No<br>Enables/Disables data seeding per store. | No | Batch | Boolean |
| Data seed chunk limit | Chunk limit for records processed in data seeding. | 10000 | Batch | Integer |
| Data seed foundation data | Values: Yes/No<br>Enables/Disables the foundation data seeding. | No | Batch | Boolean |
| Data seed store data | Values: Yes/No<br>Enables/Disables the store data seeding. | No | Batch | Boolean |
| Days to Hold Archived and Rejected Files | Gives the days to hold rejected and archived batch files before deleting them from the batch directory. | 30 | Batch | Integer |
| Maximum Job Instances Per Scheduler Execution | Gives the maximum number of jobs allowed per run of the scheduler. | 100 | Batch | Integer |
| Maximum Processing Hours For Scheduled Job | Determines the maximum processing time for a batch job. After this time, the batch schedule record is updated to failed status. | 5 | Batch | Integer |
| Procedure Log Purging Info Enabled | Determines whether the procedure log purging info is enabled. | Yes | Batch | Boolean |
| Search Limit Default for Batch Job Days | Gives the default number of days in past for batch job records to be displayed on Batch Job Admin screen on EICS. | 7 | Batch | Integer |
| Display Item Image For Container Lookup - Operations | Values: Yes/No<br><br>Yes: Indicates item image will be displayed in Container Lookup on EICS.<br><br>No: Image will not be displayed in Container Lookup on EICS. | No | Container Lookup | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Customer Order Fulfillment Restriction | Values: Restricted/Transaction Controlled/Line Controlled<br><br>Transaction Controlled: The Allow Partial Delivery indicator that comes in on the customer order will be used as it was sent.<br><br>Restricted: The Allow Partial Delivery Indicator will be updated to 'No' on the Customer Order or Transfer Request upon coming into the system.<br><br>- Customer Order Deliveries and Transfer Request/Shipment will validate the Allow Partial Delivery indicator as usual, however, it will be set to 'No' and force the user to have a full delivery (with the exception of a user override in customer order deliveries).<br><br>- Customer Order Picking: When creating a pick, the user will not be able to create the pick if there is not enough available to pick. When confirming a pick, everything must be picked on the customer order.<br><br>- Reverse Picking: when creating a reverse pick, the user must reverse pick everything that was picked.<br><br>Line Controlled: If an item is getting delivered, it must be delivered in its entirety.<br><br>- Customer Order Deliveries and Transfer Request/Shipment will validate the Allow Partial Delivery indicator as usual. When Allow Partial Delivery indicator is set to 'Yes', the system will force the user to approve/ship an item fully, if it is getting approved/shipped. When Allow Partial Delivery indicator is 'No', the system will force the user to ship the full order to the customer (with the exception of a user override in customer order deliveries). In case the user has override permission, the customer order can be shipped partially however an item getting shipped should be shipped fully.<br><br>- Customer Order Picking: When creating a pick, the user will not be able to create the pick if there is not enough available to pick at least one item. When confirming a pick, if an item is getting picked, it must be picked fully. | Transaction Controlled | Customer Order | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Customer Order Reauthorization Notification | Values: Yes/No<br><br>Yes: User will get a notification if a customer order has been reauthorized successfully, that is, when SIOCS successfully consumes and processes a RIB message from OBCS that a customer order has been released from on hold, a notification will be sent to the user<br><br>No: User will not get a notification if a customer order has been reauthorized successfully. | No | Customer Order | Boolean |
| Customer Order Receipt Notification | Values: Yes/No<br><br>Yes: A receipt notification will be generated when a customer order related transfer receipt or Direct Store Delivery (DSD) has been confirmed.<br><br>No: The notification will not be generated. | No | Customer Order | Boolean |
| Customer Order Tracking ID Required | Yes: If the store parameter Manifest Customer Order Deliveries is set to No, the Tracking ID must be captured before dispatching the Customer Order Delivery. If 'Manifest Customer Order Deliveries' is set to Yes, then it is not required.<br><br>No: Capturing Tracking ID becomes optional while dispatching the Customer Order Delivery. | No | Customer Order | Boolean |
| Display Item Image for Customer Order Delivery | Values: Yes/No<br><br>Yes: This parameter indicates that item image will be displayed in Customer Order Deliveries.<br><br>No: Images will not be displayed in Customer Order Deliveries | No | Customer Order | Boolean |
| Display Item Image for Customer Order Picking | Values: Yes/No<br><br>Yes: This parameter indicates that item image will be displayed in Customer Order Picking.<br><br>No: Images will not be displayed in Customer Order Picking. | No | Customer Order | Boolean |
| Display Item Image for Customer Order Reverse Picking | Values: Yes/No<br><br>Yes: This parameter indicates that item image will be displayed in Customer Order Reverse Picking.<br><br>No: Images will not be displayed in Customer Order Reverse Picking. | No | Customer Order | Boolean |
| Display Item Image for Customer Orders | Values: Yes/No<br><br>Yes: Indicates item image will be displayed in Customer Orders.<br><br>No: Image will not be displayed in Customer Orders. | No | Customer Order | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---------|-------------|---------------|-------|------|
| Minutes To Hold New Customer Order Before Sending Notification | Values: 0-999<br><br>This system parameter indicates the time interval in minutes to send a follow-up message to a store associated after a customer order (web order) has arrived, but no user has accessed the customer order. | 5 | Customer Order | Integer |
| Minutes To Hold Open Customer Order Pick Before Sending Notification | Values: 0-999<br><br>This system parameter dictates the time interval in minutes to send a follow-up message to a store associate after a pick list has been created but no one has started the pick list. | 15 | Customer Order | Integer |
| New Customer Order Notification | Values: Yes/No<br><br>Yes: This parameter generates a notification for a new cross channel (web order) customer order.<br><br>No: No notification is generated. | No | Customer Order | Boolean |
| New Customer Order Reverse Pick Notification | Values: Yes/No<br><br>Yes: A notification alert is generated when a new system generated reverse pick comes into the system.<br><br>No: A notification is not generated upon getting a new system generated reverse pick. | No | Customer Order | Boolean |
| Always Send DSD Receipt Cost | Values: Yes/No<br><br>Yes: When the receipt is published, the unit cost will be sent if there is not an override cost.<br><br>No: When the receipt is published, only the override cost will be sent if it exists. | No | DSD Receiving | Boolean |
| Days Beyond PO Not After Date | This parameter is used to determine the Purchase Orders returned in the deliveries on Item Detail as well as calculating the On Order Qty.<br><br>1. Planned Deliveries<br><br>Ordered Qty needs to be taking the Not After Date into account.<br><br>Purchase Order – only include those PO's where Not After Date >= Today + Days Beyond PO Not After Date<br><br>Transfer – only include those transfers where Not After Date >= Today<br><br>2. Deliveries – Only display deliveries where:<br><br>Purchase Order – only include those PO's where Not After Date >= Today + Days Beyond PO Not After Date<br><br>Transfer – only include those transfers where Not After Date >= Today | 50 | DSD Receiving | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Display Unit Cost for Direct Deliveries | Values: Yes/No<br><br>Yes: Display Unit Cost and allow editing when receiving. If On the system displays the original cost and allows entering the new cost for the on-the-fly and Dex/Nex deliveries. For the delivery with PO and ASN, it displays the unit cost. Display the Unit Cost on the Direct Delivery Report when printing.<br><br>No: Do not display this data to the user in the DSD Receiving Containers screen. Do not display the unit cost on the Direct Delivery Report. If No the system does not display the unit cost and does not allow editing or entering new cost. | Yes | DSD Receiving | Boolean |
| Displays Item Image for DSD Receiving | Values: Yes/No<br><br>Yes: This parameter indicates whether the item image will be displayed in Container Items and Item detail screens.<br><br>No: Image will not be displayed in that functional area. | No | DSD Receiving | Boolean |
| Displays Item Image for Purchase Order | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in Purchase Order Items screen.<br><br>No: Image will not be displayed in that functional area. | No | DSD Receiving | Boolean |
| DSD Receiving Preferred Currency | Values: Store Currency/Supplier Currency<br><br>This parameter will default the store or supplier currency to newly created POs depending on preference. | Store Currency | DSD Receiving | Integer |
| Ignore the Supplier DSD indicator to create a PO on the fly | Values: Yes/No<br><br>Allows the system to ignore the supplier level indicator when creating a PO in the system.<br><br>Yes: The system ignores the supplier level flag and will always allow stores to create purchase orders for any supplier based on the receipt.<br><br>No: The system will verify creating a purchase order on the fly is allowed based on the supplier level flag. | Yes | DSD Receiving | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Number of days received direct deliveries can be adjusted | Values: 0-999<br><br>0: no adjustment<br><br>1: allowed to adjust until the end of today<br><br>2: allowed to adjust until the end of tomorrow<br><br>X: allowed to adjust until X number of days starting with today as day 1<br><br>This parameter specifies the number of days received direct deliveries can be reopened and adjusted. If a direct delivery falls within the number of days, the system allows to adjust the received delivery. The user will be allowed to edit values and confirm the delivery. | 0 | DSD Receiving | Integer |
| Over Received Quantity Notification | Values: Yes/No<br><br>This parameter will generate a notification when more than the expected quantity has been received and the store parameter Direct Delivery Auto Remove Over Received Quantity is set to yes. | Yes | DSD Receiving | Boolean |
| Display Item Image for Inventory Adjustments - Execution | Values: Yes/No<br><br>Yes: The item image is displayed within Inventory Adjustments in SOCS.<br><br>No: The item image is not displayed in Inventory Adjustments. | No | Inventory Adjustment | Boolean |
| Display Item Image for Inventory Adjustments - Operations | Values: Yes/No<br><br>Yes: The item image is displayed within Inventory Adjustments in EICS.<br><br>No: The item image is not displayed in Inventory Adjustments. | No | Inventory Adjustment | Boolean |
| Display Item Image for Item Baskets - Execution | Values: Yes/No<br><br>Yes: The item image is displayed within Item Basket on the mobile.<br><br>No: The item image is not displayed in Item Basket on the mobile. | No | Item Basket | Boolean |
| Display Item Image for Item Baskets - Operations | Values: Yes/No<br><br>Yes: The item image is displayed within Item Basket on the desktop.<br><br>No: The item image is not displayed in Item Basket on the desktop. | No | Item Basket | Boolean |
| Display Item Image for Item Lookup - Execution | Values: Yes/No<br><br>Yes: The item image is displayed within Item Lookup on SOCS.<br><br>No: The item image is not displayed in Item Lookup. | No | Item Lookup | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Display Item Image for Item Lookup - Operations | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in Item Lookup in EICS. It is in the list and the details.<br><br>No: Image will not be displayed in Item Lookup. | No | Item Lookup | Boolean |
| Display Price in Search Result - Operations | Values: Yes/No<br><br>This parameter decides whether Price and Pricing Type will be displayed in the search results in the Item Lookup screen in the desktop application.<br><br>Yes: Price and Price Type will be displayed in the search results.<br><br>No: Price and Price Type will not be displayed in the search results. | Yes | Item Lookup | Boolean |
| Display SOH/ Price in Search Result - Execution | Values: Yes/No<br><br>This parameter decides whether Price and Pricing Type will be displayed in the search results in the Item Lookup screen in the mobile application.<br><br>Yes: Price and Price Type will be displayed in the search results.<br><br>No: Price and Price Type will not be displayed in the search results. | Yes | Item Lookup | Boolean |
| Background Thread Count | Gives the thread count for background tasks in SOCS. | 1 | Mobile | Integer |
| Barcode Attribute Refresh Rate Milliseconds | Determines the cache refresh rate for barcode attribute labels in milliseconds. | 3600000 | Mobile | Integer |
| Configuration Refresh Rate Milliseconds | Determines the cache refresh rate for system configurations, store configurations, batch scheduler and MPS work type in milliseconds. | 3600000 | Mobile | Integer |
| Date Output Chinese China | This is to determine the date format based on the locale. | yy-MM-dd | Mobile | String |
| Date Output Chinese Hong Kong | This is to determine the date format based on the locale. | yy-MM-dd | Mobile | String |
| Date Output Chinese Taiwan | This is to determine the date format based on the locale. | yy-MM-dd | Mobile | String |
| Date Output English Australia | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output English Canada | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output English India | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |

**Table 9-1 (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---------|-------------|---------------|-------|------|
| Date Output English Ireland | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output English South Africa | This is to determine the date format based on the locale. | yy-MM-dd | Mobile | String |
| Date Output English United Kingdom | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output English United States | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |
| Date Output French Belgium | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output French Canada | This is to determine the date format based on the locale. | yy-MM-dd | Mobile | String |
| Date Output French France | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output French Luxembourg | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output French Switzerland | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output German Austria | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output German Germany | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output German Luxembourg | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output German Switzerland | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Italian Italy | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Japanese Japan | This is to determine the date format based on the locale. | yy-MM-dd | Mobile | String |
| Date Output Korean South Korea | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output New Zealand | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Date Output Portuguese Brazil | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Portuguese Portugal | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Russian Russia | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Argentina | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Bolivia | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Chile | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Columbia | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Costa Rica | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Dominican Republic | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |
| Date Output Spanish Ecuador | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish El Salvador | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |
| Date Output Spanish Guatemala | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Honduras | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |
| Date Output Spanish Mexico | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Nicaragua | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |
| Date Output Spanish Panama | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Date Output Spanish Paraguay | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Peru | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Puerto Rico | This is to determine the date format based on the locale. | MM-dd-yy | Mobile | String |
| Date Output Spanish Spain | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Uruguay | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Date Output Spanish Venezuela | This is to determine the date format based on the locale. | dd-MM-yy | Mobile | String |
| Enable Device Camera Barcode Scan | Used for enabling device camera for scanning on SOCS. | No | Mobile | Boolean |
| External Scanner Refresh Rate Milliseconds | Determines the cache refresh rate for external scanner in millisecond. | 3600000 | Mobile | Integer |
| Inventory Adjustment Reason Refresh Rate Milliseconds | Determines the cache refresh rate for inventory adjustment reason in milliseconds on SOCS. | 3600000 | Mobile | Integer |
| Item Image Refresh Rate Milliseconds | Determines the cache refresh rate for item image in milliseconds on SOCS. | 3600000 | Mobile | Integer |
| Manual Quantity Entry Default Mode | Values: Scan Mode/Override<br><br>Scan Mode - the numeric entry popup on mobile will have its mode defaulted per the scan mode (as it has always done)<br><br>Override - the numeric entry popup on mobile will have its mode always defaulted to override, and it will not look at the scan mode. | Scan Mode | Mobile | Integer |
| Non-Sellable Quantity Type Refresh Rate Milliseconds | Determines the cache refresh rate for non-sellable quantity types in milliseconds on SOCS. | 3600000 | Mobile | Integer |
| Notification Count Refresh Rate Milliseconds | Determines the cache refresh rate for notifications in milliseconds on SOCS. | 300000 | Mobile | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| RFID Zone Refresh Rate Milliseconds | Determines the cache refresh rate for RFID zones in milliseconds in the system. | 3600000 | Mobile | Integer |
| Scan Focus Item Detail | Determines if sticky focus is enabled on SOCS screens. | No | Mobile | Boolean |
| | Yes: Focus will automatically be in the Scan field when on a screen with the scan bar. The system will retain focus in the scan bar field until an error arises or until the user/system moves focus to somewhere else.The keyboard will display on the mobile device while scanning/ entering the scan bar field. This is because the system believes you are typing into the field. | | | |
| | No: Focus will not stay in the scan bar. | | | |
| Sound Error Enabled | Determines if severe error sound will be played in case of severe errors on SOCS. | Yes | Mobile | Boolean |
| Sound Information Enabled | Determines if information sound effect will be played on SOCS. | Yes | Mobile | Boolean |
| Sound Scan Enabled | Determines if beep sound will be played on scan on SOCS. | Yes | Mobile | Boolean |
| Sound Warning Enabled | Determines if a business error sound will be played on business errors on SOCS. | Yes | Mobile | Boolean |
| Store Printer Refresh Rate Milliseconds | Determines the cache refresh rate for store printer in milliseconds on SOCS. | 3600000 | Mobile | Integer |
| Store Refresh Rate Milliseconds | Determines the cache refresh rate for notifications in milliseconds on EICS and SOCS. | 3600000 | Mobile | Integer |
| Tablet Mode Screen Size | Determines the screen size for tablet mode for SOCS. | 16,5 | Mobile | .Double |
| UOM Conversion Refresh Rate Milliseconds | Determines the cache refresh rate for UOM conversion in milliseconds | 3600000 | Mobile | Integer |
| Vibration Enabled | Determines if vibration is enabled on errors on SOCS. | No | Mobile | Boolean |
| MPS Enabled | Determines if MPS is enabled which in turn determines if MPS work types can be enabled. | Yes | MPS | Boolean |
| MPS Increment Threads | Determines the allowed thread increment factor for MPS work types. | 2 | MPS | Integer |
| MPS Maximum Queue Age Seconds | Determines the maximum seconds before MPS work queue needs to be refreshed. | 180 | MPS | Integer |
| MPS Maximum Queue Size | Determines the maximum size limit for generating MPS work queues. | 1000 | MPS | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| MPS Maximum Threads | Determines the maximum thread count for MPS work types. | 8 | MPS | Integer |
| MPS Refresh Rate Seconds | Determines the MPS work queue refresh rate after checking for the system parameter MPS Maximum Queue Age Seconds. If the MPS Maximum Queue Age Seconds has not exceeded, then this parameter is checked to determine if MPS work queue needs to be refreshed. | 15 | MPS | Integer |
| Seconds to Check for Notifications | Defines how many seconds the system will check for new notifications. This applies to any notification inserted into the system. | 300 | Notification | Integer |
| Days to Hold Areas | Values: 1-999<br><br>Purge all areas that are greater than or equal today's date minus the days to hold value. | 60 | Purge | Integer |
| Days to Hold Audit Records | Audit Records are log of activities and usage information in the system.<br><br>This parameter is to determine the number of days to hold the audit records.<br><br>The batch will delete all records where the create date is less than or equal to current date minus the days to hold. | 45 | Purge | Integer |
| Days To Hold Batch Logs | Delete all logs where the log date is less than or equal to the current date minus the days to hold for any records. | 30 | Purge | Integer |
| Days to Hold Completed Inventory Adjustments | Delete records in 'Complete' Status where the inventory complete date is less than or equal to the current date minus the days to hold. | 30 | Purge | Integer |
| Days to Hold Completed Purchase Orders | Purge all records in 'Closed' status after 'x' number of days defined by user, where the complete date (the date of when all items were received on the order) is less than or equal to the current date minus the days to hold. | 30 | Purge | Integer |
| Days to Hold Completed Staging Records | Delete all records that have been processed successfully or deleted where the update date is less than or equal to the current date minus the days to hold for any records. | 3 | Purge | Integer |
| Days to Hold Completed Stock Counts | Purges any records 'x' days after the last stock count event has occurred. In other words, when the schedule date is less than or equal to the current date the system will subtract the days to hold completed stock counts from the date and delete when this date is reached. The purging will occur when the stock count has a status of 'Complete'. | 30 | Purge | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Days to Hold Completed UINs | Indicates how long completed UINs are kept in the system. Completed UINs are defined as any UIN that is in one of the following statuses:<br>Sold<br>Shipped to Warehouse<br>Shipped to Vendor<br>Shipped to Finisher<br>Removed from Inventory<br>Customer Fulfilled | 120 | Purge | Integer |
| Days to Hold Customer Orders | Indicates the number of days that Cancelled and Fulfilled Customer Orders will be held in the system before being purged. | 60 | Purge | Integer |
| Days to Hold Expired item price | Indicates the number of days to hold the expired price changes in the system before being purged. | 30 | Purge | Integer |
| Days to Hold Expired User Roles | Values: 0-999<br>This will determine the number of days after which the expired roles will be purged from the system | 30 | Purge | Integer |
| Days to Hold In Progress Ad Hoc Stock Counts | Ad hoc stock counts that are In Progress will be deleted through the purge process. Any ad hoc count with a creation date/time stamp older than this parameter value will be deleted. For example, the default value of 1 would delete all in progress counts more than 24 hours old when the batch is run. | 1 | Purge | Integer |
| Days to Hold Item Basket | Values: 1-999<br>This will determine the number of days to hold 'Canceled' and 'Completed' Item Baskets. | 60 | Purge | Integer |
| Days to Hold Locking Records | Locking records will be purged through a batch process and the batch process will account for all locking activity across all functional areas. This is to determine the number of days to hold the locking records. The batch will delete all locking behavior around all functional areas where the lock date is less than or equal to the current business date minus the days to hold. | 3 | Purge | Integer |
| Days to Hold Notifications | This parameter is used to purge notifications which are greater than or equal to this value. | 14 | Purge | Integer |
| Days to Hold Price Change Worksheet Records | Records in the price change staging / worksheet table will be purged based upon this parameter. | 30 | Purge | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Days to Hold Price History | The 'Days to Hold Price History' parameter allows the user to keep records beyond the 4 most recent historical prices for 'x' number of days if desired. Prices in the future will not be deleted and will not be included as part of the four historical prices that will remain on the database. | 90 | Purge | Integer |
| Days to Hold Received Shipment Records | Values: 0 - 999<br><br>Purge all PO and DSD Delivery records in 'Received' and 'Cancelled' status after 'x' number of days defined by user, where the inventory completed date is less than the current date minus the days to hold.<br><br>There is a receipt record that will be deleted along with any record that is in 'Received' status. | 30 | Purge | Integer |
| Days to Hold Related Items | To determine when a related item should be purged. Program will purge Related items that have an end date in the past. The system will purge the related items after 'x' number of days defined by user, where the related items End Date is less than the current date minus the days to hold. | 60 | Purge | Integer |
| Days to Hold Resolved UIN Exceptions | Indicates how long resolved UIN exceptions are kept in the system.<br><br>The date the exception was resolved is the date the system uses to determine if the exception is ready to be purged. | 120 | Purge | Integer |
| Days to Hold RFID | Indicates how long the RFID data that are not present in the store is kept in the system. | 999 | Purge | Integer |
| Days to Hold RFID History | Values: 0 - 999<br><br>This parameter will purge RFID history that is greater than or equal today's date minus the days to hold value. | 120 | Purge | Integer |
| Days to Hold RTV | Values: 0 - 999<br><br>This parameter will decide that which RTV documents and Shipments need to be purged. The value in this parameter will decide the number of days after a RTV document or shipment gets into cancelled or completed status for document and cancelled or shipped for shipment | 30 | Purge | Integer |
| Days to Hold Sales Posting | The audit trail for the sales posting will be purged on a periodic basis based on the specified parameter. The system will purge all records from the database after the configurable number of days, where the processed date is less than or equal to current date minus the days to hold | 120 | Purge | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Days to Hold Scan Lists | Purge any records in 'Complete' or 'Cancelled' status where the post date is less than or equal to the current date minus the days to hold | 1 | Purge | Integer |
| Days to Hold Shelf Adjustment Lists | Purge any records in 'Complete' or 'Cancelled' status where the post date is less than or equal to the current date minus the days to hold | 30 | Purge | Integer |
| Days to Hold Shelf Replenishments | Purge any records in 'Complete' or 'Cancelled' status where the post date is less than or equal to the current date minus the days to hold. | 1 | Purge | Integer |
| Days to Hold Store Orders | Purge any records in 'Approved' or 'Canceled' status where the post date is less than or equal to the current date minus the days to hold. | 60 | Purge | Integer |
| Days to Hold Temporary UINs | To indicate how long the temporary UINs must stay in the system. | 120 | Purge | Integer |
| Days to Hold Ticket History | To indicate how long the tickets that printed and persisted in the history table must stay. | 60 | Purge | Integer |
| Days to Hold Transaction History | Determines the number of days after which store_item_stock_history records can be purged. | 30 | Purge | Integer |
| Days to Hold Transfer Documents | Values: 0 -999. This parameter would decide the number of days after which a Transfer document, shipments, and deliveries can be purged. | 30 | Purge | Integer |
| Days to Hold UIN Audit Information | Indicates how long UIN audit information is kept in the system. Audit information can be purged for a UIN within the system. The date the audit transaction was captured is used to determine if the record needs to be purged. | 120 | Purge | Integer |
| Number of days to keep the SFTP Log file | To determine the number days to hold the SFTP log files. | 7 | Purge | Integer |
| Display Item Image for RFID Locator | Values: Yes/No. Yes: This parameter indicates if the item image will be displayed in the RFID Locator dialog in mobile application. No: The image will not be displayed. | No | RFID Locator | Boolean |
| Days to send Notification before not after date for return requests | Values: 0-999. RTV requests generated in an external system sometimes require the RTV to be dispatched to supplier before a certain date. This option prompts the recipient of the e-mail the specified number of days before the not after date is reached, if the RTV was not dispatched. | 2 | RTV | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Displays Item Image for RTV Requests | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in that transaction. It is in the item list and the details of the transaction.<br><br>No: Image will not be displayed in that functional area. | No | RTV | Boolean |
| DSD delivery supplier for RTV | Values: Yes/No<br><br>This indicator will check to see if the DSD allowed indicator needs to be set in addition to the return allowed values when creating a supplier return.<br><br>Yes: If the "DSD delivery supplier for RTV" system option is set to 'On', then the system needs to check both the DSD indicator (Indicator on Supplier table which determines whether a supplier can create a new Purchase Order) and the return allowed indicator (also an indicator on the supplier table).<br><br>No: If the "DSD delivery supplier for RTV" system option is set to 'No', then only the return allowed indicator needs to be validated for supplier returns.<br><br>Note: regardless of the indicator, the system should always be able to dispatch the RTV if it was created in an external system. | Yes | RTV | Boolean |
| RTV Unavailable Request Qty Notification | Values: Yes/No<br><br>This system parameter will generate notification when "Auto Approve RTV request" parameter is set to On, and the request has unavailable quantity greater than the stock. | Yes | RTV | Boolean |
| Displays Item Image for RTV Shipment | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in that transaction. It is in the item list and the details of the transaction.<br><br>No: Image will not be displayed in that functional area. | No | RTV Shipment | Boolean |
| Display Item Image for Replenishment Pick | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in the replenishment pick. It is in the item list and the details of the transaction.<br><br>No: Image will not be displayed in that functional area. | No | Shelf Replenishment | Boolean |
| Display Item Image for Scan List | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in the scan list. It is in the item list and the details of the transaction.<br><br>No: Image will not be displayed in that functional area. | No | Shelf Replenishment | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Display Item Image for Shelf Adjustment | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in the shelf adjustment. It is in the item list and the details of the transaction.<br><br>No: Image will not be displayed in that functional area. | No | Shelf Replenishment | Boolean |
| Display Item Image for Stock Counts - Execution | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in the stock counts. It is in the item list and the details of the transaction.<br><br>No: The image will not be displayed. | No | Stock Counts | Boolean |
| Display Item Image for Stock Counts - Operations | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in the stock counts functionality in EICS.<br><br>No: The image will not be displayed. | No | Stock Counts | Boolean |
| Stock Count Display Default Timeframe | This parameter is to determine whether the system must prompt the user to select to whether it is performed before store open or after store close. | No | Stock Counts | Boolean |
| Stock Count Lockout Days | Stock Count Lockout Days is used to determine when a Unit and Amount Stock Count can be generated. The system will take this value plus the system date and enforce a start date of the schedule to be greater than or equal to that date.<br><br>Note: If the system is integrated with the merchandising system, the values in the two systems must be the same. | 1 | Stock Counts | Integer |
| Stock Count Null Count Quantity = 0 | Values: Yes/No<br><br>Yes: The system changes the quantity to zero for items not counted (null quantity to zero), which makes the items appear as though they were counted.<br><br>No: The system does not change the quantity to zero for items not counted, but rather leaves the value as null. These items will still appear as though they were not counted.<br><br>Note This parameter does not apply to Unit and Amount stock counts. | No | Stock Counts | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Unguided Stock Count Allow Multiple Users | Values: Yes/No<br><br>This parameter controls whether more than one user can scan simultaneously again the same child stock count for an unguided count.<br><br>Yes: The system will allow more than one user to access the same stock count, child count.<br><br>No: The system will allow more than one user to access the same stock count, but only one user may access a child stock count at a time. | No | Stock Counts | Boolean |
| Unguided Stock Counts Automatic Save | Values: Yes/No<br><br>Yes: The physical count timestamp and item count quantity are automatically saved when the next item on the count is scanned.<br><br>No: The physical count timestamp and item count quantity are saved when the user manually saves the count. It is assumed with this option the user frequently saves.<br><br>Note: The physical count timestamp is taken when the user scans the item for the first time. | No | Stock Counts | Boolean |
| Unit and Amount Stock Count Sales Processing | Values: Timestamp Processing, Daily Sales Processing<br><br>Timestamp Processing: This option is used when sales data is available near real-time with a date and time available on the transaction. The user is not prompted to select Before Store Open or After Store Close when starting the stock count since the sales timestamp will be used to compare with the timestamps taken during the stock count.<br><br>Daily Sales Processing: This option is used when sales data is only available with a date and no time is provided and/or when integrated with RMFCS. The user is either prompted or the store parameter determines when the stock count is performed, (before store opens or after store close). The date is used to determine if a sale is late or not.<br><br>Note:<br><br>Unit and Amount stock counts require some dual processing in RMFCS for capturing the financial value. RMS is only capable of processing sales data on a daily basis and disregards the time value if included. | Timestamp | Stock Counts | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Unit Stock Count Sales Processing | Values: Timestamp Processing, Daily Sales Processing<br><br>Timestamp Processing: This option is used when sales data is available near real-time with a date and time available on the transaction. The user is not prompted to select Before Store Open or After Store Close when starting the stock count since the sales timestamp will be used to compare with the timestamps taken during the stock count.<br><br>Daily Sales Processing: This option is used when sales data is only available with a date and no time is provided. The date is used to determine if a sale is late or not. | Timestamp | Stock Counts | Integer |
| Update Stock On Hand | Values: Yes/No<br><br>Yes will update SOH.<br><br>No will not update SOH. | No | Stock Counts | Boolean |
| Update Stock On Hand | Values: All/Discrepant<br><br>Discrepant Items only: The system will update only items identified as discrepant when the Update Auth Qty button is selected and when the stock count has been authorized, only the SOH is updated for the discrepant items only.<br><br>All Items: The system will update all items regardless of if they are discrepant or not when the Update Auth Qty button is selected and when the stock count has been authorized, the SOH is updated for all items, including the non-discrepant.<br><br>Note: Discrepant items are defined as items having a counted to actual variance greater than the pre-configured allowed variance. Non-discrepant items have a difference between the counted and actual qty, but they are within a tolerated variance. Unit and Amount stock counts will disregard this setting since all items will always be updated for that type of stock count. | 1 | Stock Counts | Integer |
| Auto Approve Store Orders | Values: Yes/No<br><br>Yes: external store orders will be auto approved based upon the Days before auto approving Store Orders parameter<br><br>No: external store orders will not be auto approved. | Yes | Store Order | Boolean |
| Days before auto approving Store Orders | Values: 0-999<br><br>The number of days before setting external Store Orders to approved status. | 0 | Store Order | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Days to hold before Auto Canceling Store Orders | Values: 0-999<br><br>The number of days before setting store orders to canceled status. | 0 | Store Order | Integer |
| Display Item Image Store Orders - Execution | Values: Yes/No<br><br>Yes: This parameter indicates that item image will be displayed in Store Orders.<br><br>No: Images will not be displayed in Store Orders. | No | Store Order | Boolean |
| Carrier Service Refresh Rate Milliseconds | Determines the cache refresh rate for carrier service in milliseconds | 3600000 | System Settings | Integer |
| Carton Type Refresh Rate Milliseconds | Determines the cache refresh rate for carton type in milliseconds. | 3600000 | System Settings | Integer |
| Context Type Refresh Rate Milliseconds | Determines the cache refresh rate for context type in milliseconds | 3600000 | System Settings | Integer |
| Data seed on error continue | Values: Yes/No<br><br>Indicating if terminate data Boolean seeding process on data failures. | No | System Settings | Boolean |
| Database Clock Refresh Rate Milliseconds | Determines the cache refresh rate for database clock in milliseconds. | 10800000 | System Settings | Integer |
| Default POS transaction max size | Determines the max size of pos transaction records that can be processed in one pos transaction web service call. | 1000 | System Settings | Integer |
| Item Image Request Timeout Milliseconds | When loading an item image from an external server, the amount of time before we timeout the request. | 25000 | System Settings | Integer |
| JNDI Context Refresh Rate Milliseconds | Determines the cache refresh rate for JNDI context for RIB message publisher in milliseconds. | 900000 | System Settings | Integer |
| Merchandise Hierarchy Refresh Rate Milliseconds | Determines the cache refresh rate for merchandise hierarchy in milliseconds. | 3600000 | System Settings | Integer |
| POS Transaction Max. Size for Synchronous Processing | Determines the maximum processing size for pos transaction records. All pos transaction records above this value are staged first. | 10 | System Settings | Integer |
| Price default extract size | Determines the default extract size for price change import file for Regular, Promotion and Clearance price change batch jobs. | 1000 | System Settings | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Price History Refresh Rate Milliseconds | Determines the cache refresh rate for item price history in milliseconds. | 3600000 | System Settings | Integer |
| Print Format Refresh Rate Milliseconds | Determines the cache refresh rate for print format types in milliseconds. | 3600000 | System Settings | Integer |
| Publish Non Inventory Items | Values: Yes/No<br><br>This parameter indicates whether the non-inventory items will be published in the outgoing messages or not. | Yes | System Settings | Boolean |
| Serialization Label Refresh Rate Milliseconds | Indicates serialization label refresh rate in milliseconds. | 3600000 | System Settings | Integer |
| Shipment Reason Refresh Rate Milliseconds | Determines the cache refresh rate for finisher shipment reason in milliseconds. | 3600000 | System Settings | Integer |
| Store Shipment Reason Refresh Rate Milliseconds | Determines the cache refresh rate for store shipment reason in milliseconds. | 3600000 | System Settings | Integer |
| Supplier Refresh Rate Milliseconds | Determines the cache refresh rate for supplier in milliseconds. | 3600000 | System Settings | Integer |
| Supplier Shipment Reason Refresh Rate Milliseconds | Determines the cache refresh rate for supplier shipment reason in milliseconds. | 3600000 | System Settings | Integer |
| Translation Refresh Rate Milliseconds | Determines the cache refresh rate for locale and translations in milliseconds. | 3600000 | System Settings | Integer |
| Uda Details Refresh Rate Milliseconds | Determines the cache refresh rate for UDA details in milliseconds. | 3600000 | System Settings | Integer |
| User Authorization Cache Refresh Rate Milliseconds | Determines the cache refresh rate for user authorization cache in milliseconds. | 600000 | System Settings | Integer |
| Warehouse Refresh Rate Milliseconds | Determines the cache refresh rate for warehouse in milliseconds. | 3600000 | System Settings | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Warehouse Shipment Reason Refresh Rate Milliseconds | Determines the cache refresh rate for warehouse shipment reason in milliseconds. | 3600000 | System Settings | Integer |
| Display Item Image for Ticket - Execution | This indicates whether the item image must be displayed in the ticketing dialog in mobile application. | No | Ticketing | Integer |
| Display Item Image for Ticket - Operations | This indicates whether the item image must be displayed in the ticketing dialog in desktop application. | No | Ticketing | Integer |
| Daily GMT Batch Run | Values: Yes/No<br><br>Yes: Program will only be run once a day. The server is running on GMT and the dates for processing are still stored in GMT. The batch will process everything for the GMT day the server is in, including future records for the same day as the server GMT day. | Yes | Time Zone | Boolean |
| Enable GMT for Customer Orders | Values: Yes/No<br><br>This is to determine whether the customer orders uploaded in the system are in GMT. | No | Time Zone | Boolean |
| Enable GMT for Dex/Nex | Values: Yes/No<br><br>This parameter will dictate whether the DEX/NEX data being loaded into the system is in GMT. | No | Time Zone | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Enable GMT for Direct Deliveries | Values: Yes/No<br><br>This parameter will dictate whether the Direct Delivery messages published by an external system should have dates in GMT or not.<br><br>Yes: When publishing the Direct Delivery messages, it means that the dates in the message should be written in GMT. When subscribing to the purchase order messages, it means that the dates are coming in, in GMT time and no conversion needs to occur. When publishing a purchase order message, it means that the dates in the message should be written in GMT.<br><br>No: When publishing the direct delivery messages, it means that the dates in the message should be converted from GMT and written in the store's local date/time. When subscribing to the purchase order message, it means that the dates are coming in, in are in the store's local date/time and must be converted to GMT prior to persisting the date in the database.<br><br>When publishing the purchase order message, it means that the dates in the message should be converted from GMT and written in the store's local date/time. | No | Time Zone | Boolean |
| Enable GMT for Foundation Data | Values: Yes/No<br><br>This parameter will dictate whether any foundation data messages being loaded into the system are in GMT. | No | Time Zone | Boolean |
| Enable GMT for Initial Inventory Import | Values: Yes/No<br><br>This parameter will determine whether the date/time stamp in the Initial Inventory Import file (.DAT file) is in GMT or not. | No | Time Zone | Boolean |
| Enable GMT for Inventory Adjustments | Values: Yes/No<br><br>This new system parameter will determine which date/time stamp is used in the inventory adjustment message when it is being published.<br><br>Yes: When publishing the inventory adjustment message, it means that the dates in the message should be written in GMT.<br><br>No: When publishing the inventory adjustment message, it means that the dates in the message should be converted from GMT and written in the store's local date/time. | No | Time Zone | Boolean |
| Enable GMT for POS sale import process | Values: Yes/No<br><br>This parameter will dictate whether the POS data being loaded into the system are in GMT. | No | Time Zone | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Enable GMT for Price Changes | Values: Yes/No<br><br>This parameter will dictate whether the price changes being subscribed to by the system are time zone sensitive.<br><br>Yes: When subscribing to a price change, it means that the effective date is coming in GMT time and no conversion needs to occur.<br><br>No: The effective date must be converted prior to storing the price change in the system. | No | Time Zone | Boolean |
| Enable GMT for Receiving | Values: Yes/No<br><br>This parameter will dictate whether receiving messages need to be published in GMT or not. | No | Time Zone | Boolean |
| Enable GMT for ReSA sale import process | Values: Yes/No<br><br>This parameter will dictate whether the ReSA data being loaded into the system are in GMT. | No | Time Zone | Boolean |
| Enable GMT for RTVs | Values: Yes/No<br><br>This system parameter will dictate whether the RTV message being loaded into the system is in GMT. Likewise, if the system publishes any RTV message this will determine which date/time stamp is used on the message as well<br><br>Yes: When subscribing to the RTV message, it means that the dates are coming in, in GMT time and no conversion needs to occur.<br><br>When publishing the RTV message, it means that the dates in the message should be written in GMT.<br><br>No: When subscribing to the RTV message, it means that the dates are in the store's local date/time and must be converted to GMT prior to persisting the date in the database.<br><br>When publishing the RTV message, it means that the dates in the message should be converted from GMT and written in the store's local date/time | No | Time Zone | Boolean |
| Enable GMT for Stock Counts | Values: Yes/No<br><br>This parameter will determine which date/time stamp is used in the stock count message when it is being published.<br><br>Yes: When publishing the stock count message, it means that the dates in the message should be written in GMT.<br><br>No: When publishing the stock count message, it means that the dates in the message should be converted from GMT and written in the store's local date/time. | No | Time Zone | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Enable GMT for Store Orders | Values: Yes/No | No | Time Zone | Boolean |
| | This parameter will determine which date/time stamp is used in the store order message when it is being published. | | | |
| | Yes: When publishing the store order message, it means that the dates in the message should be written in GMT. | | | |
| | No: When publishing the store order message, it means that the dates in the message should be converted from GMT and written in the store's local date/time. | | | |
| Enable GMT for Store Transfers | Values: Yes/No | No | Time Zone | Boolean |
| | This new system parameter will dictate whether the Transfer messages being loaded into the system from an external system has dates in GMT or not. Likewise, if the system publishes any Transfer messages to an external system this will determine which date/time stamp is used on the message as well. | | | |
| | Yes: When subscribing to the Transfer messages, it means that the dates are coming in, in GMT time and no conversion needs to occur. | | | |
| | When publishing a transfer message, it means that the dates in the message should be written in GMT. | | | |
| | No: When subscribing to the transfer message, it means that the dates are coming in, in are in the store's local date/time and must be converted to GMT prior to persisting the date in the database. | | | |
| | When publishing the transfer message, it means that the dates in the message should be converted from GMT and written in the store's local date/time. | | | |
| Enable GMT for Third Party Stock Counts | Values: Yes/No his parameter will determine whether the date/time stamp in the Third party stock count file (DSLDAT) is in GMT or not. | No | Time Zone | Boolean |
| Enable GMT for Vendor ASN | Values: Yes/No | No | Time Zone | Boolean |
| | This parameter will dictate whether the Vendor ASN messages being loaded into the system have dates in GMT or not. | | | |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Enable GMT for Warehouse Transfers | Values: Yes/No<br><br>This new system parameter will dictate whether the transfer messages being loaded into the system have GMT dates or not. Likewise, if the system publishes any transfer message to an external system this will determine which date/time stamp is used on the message as well.<br><br>Yes: When subscribing to the transfer messages, it means that the dates are coming in, in GMT time and no conversion needs to occur.<br><br>When retrieving transfer data, it means that the dates are in GMT time and no conversions needs to occur. This includes both reading and writing data.<br><br>When publishing a transfer message, it means that the dates in the message should be written in GMT.<br><br>No: When subscribing to the transfer message, it means that the dates are coming in, in are in the store's local date/time and must be converted to GMT prior to persisting the date in the database.<br><br>When retrieving transfer data, it means that the dates are not in GMT time and must be converted to GMT. This includes both reading and writing data.<br><br>When publishing the transfer message, it means that the dates in the message should be converted from GMT and written in the store's local date/time. | No | Time Zone | Boolean |
| Damaged Delivery Notification | Values: Yes/No<br><br>Yes: Sends a notification to the receiving store when damaged items are received on the delivery.<br><br>No: No alert is sent. This parameter generates a notification for transfers with items marked as damaged (Warehouse, Store, Finisher). | Yes | Transfer Receiving | Boolean |
| Days Shipped Delivery Overdue Notification | Values: 1-999<br><br>This parameter generates a notification when the delivery is overdue. The delivery will be considered overdue when the create date of the delivery plus the days from this parameter have passed. This will include only deliveries from source type store. | 7 | Transfer Receiving | Integer |

ORACLE®

**Table 9-1 (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Display Item Image for Transfer Receiving | Values: Yes/No<br>Yes: This parameter indicates if the item image will be displayed in that transaction. It is in the item list and the details of the transaction.<br>No: Image will not be displayed in that functional area. | No | Transfer Receiving | Boolean |
| External Finisher UIN Qty Discrep Notification | Values: Yes/No<br>This system parameter will generate notification when there is a discrepancy with the number of UINs on the ASN and the UINs received when auto receiving with a Source Type of 'Finisher'.<br>Yes: Whenever the transaction cannot be auto received, the system generates a notification when there is a discrepancy with the number of UINs on the Finisher Return and the UINs received.<br>Auto Received by batch<br>Auto Received thru RIB Injector.<br>No: No alert is generated. | Yes | Transfer Receiving | Boolean |
| Misdirected Container Notification | Values: Yes / No<br>Yes: Sends a notification when a location receives a container belonging to another location.<br>No: No notification is sent.<br>This system parameter will generate a notification when there is a misdirected container that has been received in another location. | Yes | Transfer Receiving | Boolean |
| Number of Days Received Transfer Deliveries can be Adjusted | Values: 0-999<br>This parameter controls the number of days a container can be adjusted within a receipt after (Warehouse, Store, Finisher) are received.<br>0: no adjustment allowed<br>1: allowed to adjust until the end of today<br>2: allowed to adjust until the end of tomorrow<br>X: allowed to adjust until x days starting from today | 0 | Transfer Receiving | Integer |
| Quick Receiving - Receive misdirected containers | Values: Yes/No<br>Yes: Enables the ability to receive misdirected containers.<br>No: Disables the ability to receive misdirected containers. | Not Allowed | Transfer Receiving | Integer |
| Quick Receiving - Receive missing containers | Values: Yes/No<br>Yes: Enables the ability to receive missing containers.<br>No: Disables the ability to receive missing containers. | Yes | Transfer Receiving | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Receive Entire Transfer | Values: Yes/No<br><br>Yes: User is ONLY allowed to receive the entire delivery. It is not allowed to add any items, only confirmation of the receipt is allowed (Warehouse, Store, Finisher).<br><br>No: The user is not limited to only receiving the entire delivery. | No | Transfer Receiving | Boolean |
| Store Receiving Force Close Indicator | Values: RL / SL / NL This parameter applies to deliveries with a Source Type of 'Store'.<br><br>RL: (Receiver Loss) Any shipped quantity that was not received is a loss at the receiving store.<br><br>SL: (Sending Loss) Any shipped quantity that was not received is a loss at the sending store.<br><br>NL: (No Loss) Any shipped quantity that was not received does not affect the receiving or the sending store. | Receiving Loss | Transfer Receiving | Integer |
| Store Receiving Over/Under Notification | Values: Yes / No<br><br>This parameter generates a notification to the receiving store when items on a transfer receipt with a Source Type of 'Store' is greater than or less than the expected quantity or if the expected quantity is null or zero. A notification should not be sent in the case of a manually created container or a copied misdirected container.<br><br>On: Sends a notification when the receiving store over or under receives goods.<br><br>No: No alert is sent. | Yes | Transfer Receiving | Boolean |
| Warehouse/ Store UIN Qty Discrep Notification | Values: Yes/No<br><br>This system parameter will generate notification when there is a discrepancy with the number of UINs on the ASN and the UINs received when auto receiving a warehouse or store delivery with a Source Type of 'Warehouse' or 'Store'.<br><br>Yes: Sends a notification when there is a discrepancy with UINs on the ASN. The number of UINs on the ASN and the Qty received do not match and cannot be auto received.<br><br>Auto Received by batch (Store), (Warehouse) Auto Received thru RIB Injector<br><br>No: No notification is generated. | Yes | Transfer Receiving | Boolean |
| Display Item Image for Transfer Shipment | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in that transaction. It is in the item list and the details of the transaction.<br><br>No; Image will not be displayed in that functional area. | No | Transfer Shipment | Boolean |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Days to send Notification before not after date for transfer requests | Values: 0-999<br><br>For transfer requests generated in an external system (warehouse, store or finisher), his option sends a notification the specified number of days before the not after date is reached, if the transfer was not dispatched. | 2 | Transfers | Integer |
| Display Item Image for Transfer | Values: Yes/No<br><br>Yes: This parameter indicates if the item image will be displayed in that transaction. It is in the item list and the details of the transaction.<br><br>No: Image will not be displayed in that functional area. | No | Transfers | Boolean |
| Transfer Request Approve Notification | Values: Yes/No<br><br>Yes: A notification will be generated when a requested transfer is approved.<br><br>No: No notification will be generated.<br><br>Note: The notification will only be generated for SIOCS initiated store to store requests. | No | Transfers | Boolean |
| Transfer Request Notification | Values: Yes/No<br><br>Yes: A notification will be generated when a transfer is requested.<br><br>No: No notification will be generated.<br><br>Note: The notification will only be generated for SIOCS initiated store to store requests. | No | Transfers | Boolean |
| Transfer Request Reject Notification | Values: Yes/No<br><br>Yes: A notification will be generated when a transfer is rejected.<br><br>No: No notification will be generated.<br><br>Note: The notification will only be generated for SIOCS initiated store to store requests. | No | Transfers | Boolean |
| Unavailable Qty Discrepancy Notification | Values: Yes/No<br><br>Yes: A notification will be generated when a transfer request fails auto-approval.<br><br>No: No notification will be generated.<br><br>Note: Auto Accept External Generated Request is set to On and Unavailable quantity requested is more than the sending store has. | No | Transfers | Boolean |
| Currency Default Type | Gives the default currency for EICS and SOCS. | USD | UI | String |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Display Item Description | Values: Short Description / Long Description<br><br>On Mobile the description will be short or long based upon this configuration.<br><br>Short Item Description: The description displayed everywhere will be the short item description.<br><br>Note that when integrated with Oracle Retail's merchandising system, the short description of an item is a product of the first 20 characters of the long description.<br><br>Long Item Description: The description displayed everywhere will be the long item description.<br><br>Note that when integrated with Oracle Retail's merchandising system, the Merchandising system concatenates the diff descriptions with the long item description so the user can view all diff information. The diffs display in order from Diff 1 to Diff 4.<br><br>On the desktop the system will always use the long description. | Short Description | UI | Integer |
| Display Item Description Diffs | Values: Yes/No<br><br>Yes: A second item description line for diffs will be used wherever an item description is displayed where applicable.<br><br>No: The second line will not be displayed for the item description and the diffs will not be displayed separately. | Yes | UI | Boolean |
| Maximum Manual Quantity Entry | Values: 1-100,000,000<br><br>The value set here will be the maximum value a user can enter for a quantity via the Numeric Entry on the mobile. | 999 | UI | Integer |
| Problem Line UI Limit | Gives the recommended item count in product group component screen for problem line stock count product group. | 1500 | UI | Integer |
| Search Date Range Default for Transaction History | Values: 0-99<br><br>This holds the default number of days for which the transaction history records need to be listed in the Transaction History List screen. | 0 | UI | Integer |
| Search Limit Default for Area - Operations | Values: 1-999<br><br>This parameter indicates the default search limit for the Area. | 50 | UI | Integer |
| Search Limit Default for Container Lookup - Execution | Values: 1-999<br><br>Indicates the default search limit for Container Lookup. | 50 | UI | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Search Limit Default for Container Lookup - Operations | Values will be 1-999<br><br>This parameter will determine the number of records to be displayed on container lookup list screen.<br><br>The default value on container lookup list screen should be set to the value for the system parameter. | 50 | UI | Integer |
| Search Limit Default for Customer Order Picking | Values: 1-999<br><br>Indicates the default search limit for Customer Order Picking. | 50 | UI | Integer |
| Search Limit Default for Customer Orders | Values: 1-999<br><br>Indicates the default search limit for Customer Orders. | 50 | UI | Integer |
| Search Limit Default for DSD Receiving | This is to determine the default search limit for DSD receiving list. | 50 | UI | Integer |
| Search Limit Default for Finisher Lookup - Operations | Values: 1-999<br><br>Indicates the default search limit for Finisher Lookup. | 50 | UI | Integer |
| Search Limit Default for Format Assignment - Operations | Values: 1-999<br><br>This parameter will determine the number of records to be displayed on the Format Assignment List screen in desktop application.<br><br>The default value on the ticketing search screen should be set to the value from this parameter. | 50 | UI | Integer |
| Search Limit Default for Inventory Adjustments - Execution | Values: 1-999<br><br>Indicates the default search limit for Inventory Adjustments on SOCS. | 50 | UI | Integer |
| Search Limit Default for Inventory Adjustments - Operations | Values: 1-999<br><br>Indicates the default search limit for Inventory Adjustments on EICS. | 50 | UI | Integer |
| Search Limit Default for Item Baskets - Execution | Values: 0-999<br><br>Indicates the default search limit for Item Baskets on SOCS. | 50 | UI | Integer |
| Search Limit Default for Item Baskets - Operations | Values: 0-999<br><br>Indicates the default search limit for Item Baskets on EICS. | 50 | UI | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Search Limit Default for Item Lookup - Execution | Values: 1-999<br><br>Indicates the default search limit for Item Lookup on SOCS. | 50 | UI | Integer |
| Search Limit Default for Item Lookup - Operations | Values: 1-999<br><br>Indicates the default search limit for Item Lookup on EICS. | 50 | UI | Integer |
| Search Limit Default for Item Scan Number Lookup | Values: 1-999<br><br>This parameter will determine the number of records to be displayed on the Item Scan Number Lookup screen.<br><br>The default value on the ISN Lookup search screen should be set to the value from this parameter. | 500 | UI | Integer |
| Search Limit Default for MPS Staged Messages | Values: 1-999<br><br>Indicates the default search limit for MPS staged messages on MPS staged message screen in EICS. | 50 | UI | Integer |
| Search Limit Default for Notifications | Values: 1-999<br><br>Indicates the default search limit for Notifications. | 50 | UI | Integer |
| Search Limit Default for Open Transaction | Values: 1-999<br><br>Indicates the default search limit for Open Transactions. | 50 | UI | Integer |
| Search Limit Default for Operational Views - Operations | Values: 1-999<br><br>This parameter will determine the number of records to be displayed on various operational view screens.<br><br>The default value on various operational views screens should be set to the value from the system parameter. | 50 | UI | Integer |
| Search Limit Default for POS Transaction Resolution | Values: 1-999<br><br>This parameter will determine the number of records to be displayed on the POS Transaction resolution dialog in desktop application.<br><br>The default value on the ticketing search screen should be set to the value from this parameter. | 50 | UI | Integer |
| Search Limit Default for Purchase Order | This parameter will determine the default number of records to be displayed on PO list screen. | 50 | UI | Integer |
| Search Limit Default for Replenishment Pick | Values: 1-999<br><br>Indicates the default search limit for shelf replenishment. | 50 | UI | Integer |

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---------|-------------|---------------|-------|------|
| Search Limit Default for RTV | Values: 1-999 <br><br> Indicates the default search limit for Returns. | 50 | UI | Integer |
| Search Limit Default for Scan List | Values: 1-999 <br><br> Indicates the default search limit for scan lists. | 50 | UI | Integer |
| Search Limit Default for Shelf Adjustment | Values: 1-999 <br><br> Indicates the default search limit for shelf adjustments. | 50 | UI | Integer |
| Search Limit Default for Stock Count | Values: 1-999 <br><br> Indicates the default search limit for Stock Counts. | 50 | UI | Integer |
| Search Limit Default for Store Order | Values: 1-999 <br><br> Indicates the default search limit for Store Orders. | 50 | UI | Integer |
| Search Limit Default for Supplier Lookup | Values: 1-999 <br><br> Indicates the default search limit for Supplier Lookup. | 50 | UI | Integer |
| Search Limit Default for Supplier Lookup - Operations | Values: 1-999 <br><br> This parameter will determine the number of records to be displayed on supplier lookup list screen. <br><br> The default value on supplier lookup list screen should be set to the value for the system parameter. | 50 | UI | Integer |
| Search Limit Default for Ticket - Operations | Values: 1-999 <br><br> This parameter will determine the number of records to be displayed on the ticketing dialog in desktop application. <br><br> The default value on the ticketing search screen should be set to the value from this parameter. | 50 | UI | Integer |
| Search Limit Default for Transaction History | Values: 1-999 <br><br> Indicates the default search limit for Transaction History. | 50 | UI | Integer |
| Search Limit Default for Transfer Receipts | Values: 1-999 <br><br> Indicates the default search limit for Transfer receipts. | 50 | UI | Integer |
| Search Limit Default for Transfer Shipment | Values: 1-999 <br><br> Indicates the default search limit for Transfer shipments. | 50 | UI | Integer |
| Search Limit Default for Transfers | Values: 1-999 <br><br> Indicates the default search limit for Transfer documents. | 50 | UI | Integer |

ORACLE®

**Table 9-1    (Cont.) System Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Search Limit Default for Troubled Transaction List | Values: 1-999 <br><br> Indicates the default search limit for Troubled Transactions | 50 | UI | Integer |
| Search Limit Default for UIN Lookup | Values: 1-999 <br><br> Indicates the default search limit for UIN Lookup. | 50 | UI | Integer |
| Shelf Replenishment UI Limit | Gives recommended item count in product group component screen for shelf replenishment pick product groups. | 1500 | UI | Integer |
| Store Order UI Limit | This parameter indicates the UI limit for store orders, used in generation of the store orders. It is also used in the Recommended # of Items in Product Group Components. | 1500 | UI | Integer |
| Ticketing UI Limit | This parameter indicates the UI limit for ticketing, used in generation of the auto ticket print. It is also used in the Recommended # of Items in Product Group Components. | 1500 | UI | Integer |
| Unit and Amount Count UI Limit | Gives the recommended item count in product group component screen for unit and amount stock count product group. | 1500 | UI | Integer |
| Unit Count UI Limit | Gives the recommended item count in product group component screen for unit stock count product group. | 5000 | UI | Integer |
| Allow Store UIN Relocation | Values: Yes/No <br><br> Indicates whether UIN can be relocated from one store to another. | Yes | UIN | Boolean |

# Store Admin Parameters

**Table 9-2    Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Display Shopfloor/ Backroom Quantity | Values: Yes/No <br><br> Yes: This parameter indicates if the shop floor and back room SOH should be displayed in various areas of the system including item lookup as well as transactions. <br><br> No: Shop floor and back room SOH will not be displayed in various areas of the system. | No | Admin | Boolean |
| Manifest Weight UOM | Values: List of UOMs from the Weight UOM table <br><br> The UOM selected for this store admin will be used as the Weight UOM for the weight on the BOL in store to store transfer shipments, customer order deliveries and returns. | LBS | Admin | String |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| SSCC Shipping Label ID Generation | Values: Yes/No<br><br>Yes: The system will generate an identifier for printing on the shipping label.<br><br>No: The user will need to enter an identifier for printing on the shipping label.<br><br>This store parameter will be used for RTV Shipping and Transfer Shipping. | Yes | Admin | Boolean |
| UIN Processing Enabled | Values: Yes/No<br><br>Yes: Enables UIN processing for the store.<br><br>No: UIN functionality is disabled for the store. | No | Admin | Boolean |
| Use Extended Attribute Entry | This will turn on and off the feature for editing transaction item level attributes, the capturing of Extended Attributes will be skipped.<br><br>Values: Yes/No<br><br>Yes: Extended Attributes will be captured in the functional areas in EICS and SOCS.<br><br>No: Extended Attributes will be captured in the functional areas in EICS and SOCS. The Ext Attributes data entry screen will not be available. | No | Admin | Boolean |
| Allow Picking By Area | Values: Yes/No<br><br>Yes: Picking by Area is allowed. The user will be able to select an Area when creating a pick if the Customer Order Fulfillment Restriction is set to 'Transaction Controlled'. The system will narrow down the customer order to those items within an area when creating the pick.<br><br>No: Picking by Area is not allowed, and the system will always look at all items on the customer order when creating a pick. | Yes | Customer Order | Boolean |
| Auto Pick Mixed Containers | Values: Yes/No<br><br>Yes: If a container has items in it that are for both customer orders and non-customer orders, the system will auto pick the container. It will mark those items that exist on the customer order as picked.<br><br>No: If there is a mixed container of customer order and non-customer order items, it will not get auto picked and the picked quantities will not be updated. | No | Customer Order | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Auto Pick On Receive - Direct Delivery | Values: Yes/No<br><br>Yes: The system will automatically fill in the pick quantities on the customer order when receiving. This can only happen if the customer order record has already come into the system. If there is no customer order, the auto picking will not happen at the time of receiving, rather it will occur when the customer order comes in.<br><br>No: The system will not pick when receiving goods. | No | Customer Order | Boolean |
| Auto Pick On Receive - Transfer Receiving | Values: Yes/No<br><br>Yes: The system will automatically fill in the pick quantities on the customer order when receiving. This can only happen if the customer order record has already come into the system. If there is no customer order, the auto picking will not happen at the time of receiving, rather it will occur when the customer order comes in.<br><br>No: The system will not pick when receiving goods in transfer receiving. | No | Customer Order | Boolean |
| Default Customer Order Picking Method | Values: Bin / Store Customer Order<br><br>This parameter is used to define the default picking method when creating a customer order pick, bin or store customer order.<br><br>Note this is just a default and the user can still switch the picking method. | Store Customer Order | Customer Order | Integer |
| Default Number of Bins | Values: 1-999<br><br>This parameter will determine the number of bins to default into the 'Bin Qty' field if the user selects 'Bin' as the pick type when creating the pick. | 1 | Customer Order | Integer |
| Dispatch Validate | Values: Ship Direct, Ship Submit<br><br>Ship Direct: System will control all processes. The user will be able to go from create/edit directly to dispatch. There will not be a submit option.<br><br>Ship Submit: This option will require the user to press the Submit option and require a specific press of the dispatch button. | Ship Direct | Customer Order | Integer |
| Generate Bins | Values: System / Manual<br><br>System: The system will automatically generate the bin IDs when the pick is created.<br><br>Manual: The system will require the user to enter the bin IDs upon acting on the created pick. This option removes the need for printing out labels since the labels already exist on the bin. For both System and Manual, the user will still have the option to print labels for the bins. | Manual | Customer Order | Integer |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Item Substitution - Store Discretion | Values: Yes/No<br><br>This store parameter is used to determine if the user can use their own discretion when doing substitute items in the picking process.<br><br>Yes: The user can choose any item to be used as a substitute item.<br><br>No: Only those items that are defined as substitutes can be substituted. | No | Customer Order | Boolean |
| Override Bin Quantity | Values: Yes/No<br><br>This store parameter determines whether the user is allowed to override the default bin quantity when creating a pick by bin. The Bin Quantity is defaulted based upon the store parameter for Default Number of Bins. | No | Customer Order | Boolean |
| Picking Required for Customer Orders | Values: Yes/No<br><br>Yes: Requires that manual picking be performed on the customer order prior to being able to create a delivery for it.<br><br>No: Picking is not necessary in order to create a delivery. | Yes | Customer Order | Boolean |
| Pre-shipment Notification | Values: Yes / No<br><br>This parameter will drive the following functionality:<br><br>Yes: The system will publish a pre-shipment message<br><br>No: The system will not publish a pre-shipment message | No | Customer Order | Boolean |
| Reserve Customer Order Inventory Upon Receiving | Values: Yes/No<br><br>This store parameter will dictate when inventory for a web order customer order should be reserved.<br><br>Yes: Inventory will be reserved upon receiving a delivery in the store which contains the customer order (Store to store transfer, DSD, Warehouse to Store transfer). If the retailer wishes to fulfill the customer orders from deliveries, for example getting the goods from the warehouse, then this option would be set to Yes to reserve upon receiving.<br><br>No: Inventory will be reserved upon getting the customer order into the store. If the retailer chooses to mainly fulfill customer orders from within the stock in the store, this parameter would be set to No, thus reserving inventory right away when the customer order is received. | No | Customer Order | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Allow Multiple Deliveries against PO with No ASN | Values: Yes/No<br><br>Yes: The user can create more than one delivery for the same PO when the PO does not have an associated ASN.<br><br>No: The user can only create a single delivery against a PO when the PO does not have an associated ASN. The PO will be closed when the delivery is confirmed. | Yes | DSD Receiving | Boolean |
| Auto close days after expected date | Values: 0-999<br><br>Number of days after the expected delivery date the ASN will be closed. | 5 | DSD Receiving | Integer |
| Direct Delivery Auto Remove Over Received Quantity | Values: Yes/No<br><br>Yes: If set to Yes, the user is allowed to add any quantity for the DSD, but any quantity above the expected quantity will be removed from the transaction. After the user confirms the transaction, they are prompted that any over received quantities will be removed. The user can create more than one delivery for the same PO when the PO does not have an associated ASN.<br><br>No: The over received quantities will stay in the delivery transaction. | No | DSD Receiving | Boolean |
| Direct Delivery Default to ShopFloor Receiving | Values: Yes/No<br><br>This parameter determines whether the DSD receiving dialog will default to receive inventory into the shop floor instead of automatically receiving into the back room or delivery bay.<br><br>Yes: The DSD Receiving Container workflow will default the option to receive inventory into the shop floor. The shop floor inventory bucket will be incremented instead of the backroom or delivery bay bucket. If a capacity is defined for the item and Direct Delivery Receive Item Capacity is set to yes, the maximum shop floor quantity will equal the capacity; otherwise, the shop floor will be updated to the entire receipt amount. If the capacity is used and if the receiving quantity is excess, the balance is incremented to delivery bay or back room depending on the Replenishment - Delivery Bay Inventory parameter.<br><br>No: The option will be defaulted to back room or delivery bay depending on the Replenishment - Delivery Bay Inventory parameter. If this parameter is on, the system will increment the delivery bay bucket instead of back room. | No | DSD Receiving | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Direct Delivery Invoice Entry | Values: Enabled/Disabled/Unique<br><br>Enabled: This option allows the user to enter any value for the invoice number, including duplicates.<br><br>Disabled: The Invoice Number and date fields are disabled. Unique: The Invoice Number field will allow entry; however, the user will not be able to enter a duplicate invoice number based upon the supplier.<br><br>Upon entering an Invoice Number, the system will validate if the invoice already exists for the supplier defined on the DSD. If a duplicate exists, there will be an error. If no duplicate exists, the invoice number will be accepted. | Enabled | DSD Receiving | Integer |
| Direct Delivery Receive Item Capacity | Values: Yes/No<br><br>This parameter determines whether the capacity will be considered while receiving the deliveries.<br><br>Yes: While receiving, the capacity will be considered. For example: If capacity is 50, delivery is for 100, and Available SOH is 10 on the shop floor, if this parameter is on and receive in shop floor is checked, 40 (un-damaged) will be moved to shop floor and the rest to the backroom.<br><br>No: While receiving, the capacity will not be considered. Damaged inventory will not move to the shop floor. | No | DSD Receiving | Boolean |
| DSD Receiving Auto Remove Damaged Quantity | Values: Yes/No<br><br>Yes: All damaged items on the delivery are removed automatically when confirming the transaction.<br><br>No: All damaged items remain on the delivery when confirming the transaction. | No | DSD Receiving | Boolean |
| Days Before Item Basket Expiration | Values: 0-999<br><br>This parameter will be used to add to the system date when defaulting the expiration date on an item basket. A value of 0 would set the expiration date to today. A value of 1 will set the expiration date to tomorrow (today + 1). | 1 | Item Basket | Integer |
| Auto Accept External Generated RTV Request | Values: Yes/No<br><br>This parameter determines whether system automatically approves the return request and defaults the requested quantity to the accepted quantity for externally generated RTV requests. | No | RTV | Boolean |
| Not After Date Default days | Values: 0-999<br><br>This parameter would decide the number of days after which a RTV document can be closed After not after date is passed. All status documents would be marked cancelled once this criterion is met. | 30 | RTV | Integer |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Dispatch Validate | Values: Ship Direct, Ship Submit<br><br>Ship Direct: SIOCS will control all processes. The user will be able to go from create/edit directly to dispatch.<br><br>Ship Submit: This option will require the user to press the Submit button and require a specific press of the dispatch button. An additional option is that an external system will generate a dispatch message through a standard web service. | Ship Direct | RTV Shipment | Integer |
| Pre-shipment Notification | Values: Yes/No<br><br>Yes: The system will publish a pre-shipment message.<br><br>No: The system will not publish a pre-shipment message. | No | RTV Shipment | Boolean |
| RTV Shipment Carrier Default | Values: Sender / Receiver / Third Party<br><br>When creating a return, the Carrier Type on the BOL will default initially based upon this parameter. The user can still change this value and if so, that will be the value used on the return.<br><br>Sender: Sender will be selected for Carrier Type on BOL<br><br>Receiver: Receiver will be selected for the Carrier type on BOL.<br><br>Third Party: Third Party will be selected for the Carrier type on the BOL. The type (drop down) will be defaulted to "Other". | Third Party | RTV Shipment | Integer |
| Display Sequence Fields | Values: Yes/No<br><br>Yes: Will display sequencing information throughout the application including guided stock count option, capacity, and an item's locations including primary location.<br><br>No: Sequence information will not be displayed in the system. | No | Sequencing | Boolean |
| Replenishment - At Case Level | Yes: The standard UOM will default to Cases on the shelf replenishment screens.<br><br>No: The standard UOM will default to Units on the shelf replenishment screens.<br><br>Note: This parameter will determine whether to replenish shelves at the Case or standard unit of measure. This will override the 'Default UOM' system parameter. | Yes | Shelf Replenishment | Boolean |
| Replenishment - Delivery Bay Inventory | Values: Yes/No<br><br>Yes: The delivery bay will be used for replenishment.<br><br>No: The delivery bay will not be used. | Yes | Shelf Replenishment | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---------|-------------|---------------|-------|------|
| Replenishment - End of Day max. fill % | This parameter will determine the percentage the stock can fall to before creating the end of day replenishment list. | 100 | Shelf Replenishment | Double |
| Replenishment - Item Substitution Store Discretion | Values: Yes/No<br><br>Yes: The user is allowed to choose any item to substitute. An item lookup feature will allow the user to search for an item to select.<br><br>No: The user is restricted to scanning/entering an item that exists on the list of approved substitute items defined by the merchandising system.<br><br>Note: If there are no items defined for item substitution, the dialogue will be displayed with the original item in the header with no substitutes. | No | Shelf Replenishment | Boolean |
| Replenishment - Within Day Max. fill % | This parameter will determine the percentage the stock can fall to before creating the within day replenishment list. | 75 | Shelf Replenishment | Double |
| Display Late Inventory Adjustment Message | Values: Yes/No<br><br>Yes: When the user is confirming a Stock Count in the Authorization phase and there are items on the stock count with In Progress Inventory Adjustments, the user may return to the stock count to complete the inventory adjustments or to continue and ignore the adjustments.<br><br>When the user is approving an inventory adjustment with items on an open stock count, a message is displayed allowing the user to determine if further processing should be undertaken. Similar logic to how late sales is processed will be used for these adjustments.<br><br>No: When the user is confirming a Stock Count in the Authorization phase and there are items on the stock count with In Progress Inventory Adjustments the system ignores the inventory adjustments and allows the user to complete the count.<br><br>When the user is approving an inventory adjustment, there is no additional processing.<br><br>Note: The system will process the inventory adjustment like how late sales are processed and determine if the stock count should be adjusted or not. The update to the stock count is not immediate but rather is at the time of authorization if there are reversing entries created. | No | Stock Counts | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Stock Count Default Timeframe | Values: Before Store Open/After Store Close<br><br>This parameter defines when the stock count is performed in relation to the store opening hours for Daily Sales Processing. This value may be overridden at the time of the stock count, if the system is configured to allow the override. If an override is allowed, this setting will determine the default value displayed.<br><br>Before Store Open: The stock count is performed before the opening of the store. All sales on the day of the stock count will only update SOH. It will not update any counted quantities.<br><br>After Store Close: The stock count is performed after the closing of the store. All sales on the day of the stock count will update both SOH and any counted quantities. If using RMS, After Store Close must be selected.<br><br>Note: Timestamp processing does not use this parameter. | Before Store Open | Stock Counts | Integer |
| Display Delivery Timeslot | Values: Yes/No<br><br>Yes: The Delivery Timeslot fields will display throughout Store Orders as well as the Admin screen Delivery Timeslots.<br><br>No: The Delivery Timeslot fields will NOT display throughout Store Orders as well as the Admin screen Delivery Timeslots. | Yes | Store Order | Boolean |
| DSD Delivery Supplier for Store Order | Values: Yes/No<br><br>This indicator will check to see if the DSD allowed indicator needs to be yes when adding a supplier restriction when creating a store order.<br><br>Yes: The system needs to check the DSD indicator (Indicator on Supplier table which determines whether a supplier can create a new Purchase Order) when creating a new Store Order. If the indicator is set to 'Yes' the supplier can be added. If it is set to 'No', the supplier cannot be added.<br><br>No: The DSD indicator on the supplier does not need to be checked. | Yes | Store Order | Integer |
| Enable Area for Store Order | Values: Yes/No<br><br>Yes: The Area will display throughout Store Orders.<br><br>No: The Area field will NOT display throughout Store Orders. | Yes | Store Order | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Supplier Restriction for Store Order | Values: Enabled / Required / Disabled<br><br>Enabled: Supplier will be available as a restriction when creating and searching for a store order.<br><br>Required: Supplier will be available as a restriction when creating and searching for a store order. When creating it will also be required.<br><br>Disabled: Supplier will not be available as a restriction when creating a store order. | Enabled | Store Order | Integer |
| Warehouse Restriction for Store Order | Values: Enabled / Required / Disabled<br><br>Enabled: Warehouse will be available as a restriction when creating and searching for a store order.<br><br>Required: Warehouse will be available as a restriction when creating and searching for a store order. When creating it will also be required.<br><br>Disabled: Warehouse will not be available as a restriction when creating a store order. | Enabled | Store Order | Integer |
| Auto Generate Item Tickets for Clearance Price Changes | Values: Yes/No<br><br>Yes: When a clearance price event comes from the pricing system, a new item ticket is sent to the ticketing dialogue.<br><br>No: When a clearance pricing event comes from the pricing system, the system does not generate an item ticket.<br><br>This determines whether the system must auto generate item tickets in the system when there is a clearance price event coming in from the pricing system. | No | Ticketing | Boolean |
| Auto Generate Item Tickets for Description Changes | Values: Yes/No<br><br>Yes: When a new description comes from the merchandising system, a new item ticket is sent to the ticketing dialogue.<br><br>No: When a new description comes from the merchandising system, the system does not generate an item ticket.<br><br>This configuration will be used to auto send item tickets to ticketing when an item description is updated and sent to EICS. | No | Ticketing | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Auto Generate Item Tickets for Promotion Price Changes | Values: Yes/No<br><br>Yes: When a promotion price event comes from the pricing system, a new item ticket is sent to the ticketing dialogue.<br><br>No: When a promotion pricing event comes from the pricing system, the system does not generate an item ticket.<br><br>This determines whether the system must auto generate item tickets in EICS when there is a promotion price event coming in from the pricing system. | No | Ticketing | Boolean |
| Auto Generate Item Tickets for Regular Price Changes | Values: Yes/No<br><br>Yes: When a regular price change comes from the pricing system, a new item ticket is sent to the ticketing dialogue.<br><br>No: When a regular price change comes from the pricing system, the system does not generate an item ticket.<br><br>This determines whether the system must auto generate item tickets in EICS when there is a regular price change event coming in from the pricing system. | No | Ticketing | Boolean |
| Auto Generate Shelf Edge Labels for Clearance Price Changes | Values: Yes/No<br><br>Yes: When a clearance price event comes from the pricing system, a shelf edge label is sent to the ticketing dialogue.<br><br>No: When a clearance pricing event comes from the pricing system, the system does not generate a label.<br><br>This determines whether the system must auto generate item tickets in the system when there is a clearance price event coming in from the pricing system. | No | Ticketing | Boolean |
| Auto Generate Shelf Edge Labels for Description Changes | Values: Yes/No<br><br>Yes: When a new description comes from the merchandising system, a shelf edge label is sent to the ticketing dialogue.<br><br>No: When a new description comes from the merchandising system, the system does not generate a label.<br><br>This configuration will be used to auto generate labels when an item description is updated and to send to EICS. | No | Ticketing | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Auto Generate Shelf Edge Labels for Promotion Price Changes | Values: Yes/No<br><br>Yes: When a promotion price event comes from the pricing system, a new shelf edge label is sent to the ticketing dialogue.<br><br>No: When a promotion pricing event comes from the pricing system, the system does not generate a label.<br><br>This determines whether the system must auto generate labels in EICS when there is a promotion price event coming in from the pricing system. | No | Ticketing | Boolean |
| Auto Generate Shelf Edge Labels for Regular Price Changes | Values: Yes/No<br><br>Yes: When a regular price change comes from the pricing system, a new shelf edge label is sent to the ticketing dialogue.<br><br>No: When a regular price change comes from the pricing system, the system does not generate a label.<br><br>This determines whether the system must auto generate labels in EICS when there is a regular price change event coming in from the pricing system. | No | Ticketing | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Item Print Events | This is to determine the default item price events for the ticket printing.<br><br>Values: Always, Clearance, Promotion, Permanent, Clearance or Permanent and Any Price Event<br><br>Always: This option will always print a ticket regardless of if there is a price change.<br><br>Clearance: Only print a ticket if on the specific date any clearance event is effective. So, for Clearance 2, that means today's date + 2 days, if the item on that day has a clearance going on, print the clearance ticket.<br><br>Promotion: Only print a ticket if on the specific date any promotion event is effective.<br><br>Regular or Clearance: Only print a ticket if on the specific date any Regular or Clearance event is getting effective.<br><br>Permanent: Only print a ticket if on the specific date any Regular even of getting effective.<br><br>Any Price Event (Promotion, Clearance or Permanent (Regular)): Based on the date selected, if any price event goes into effect (clearance, promotion or regular price a ticket. If no price event goes into effect, print nothing).<br><br>This is to determine the default item price events for the ticket printing.<br><br>Values: Always, Clearance, Promotion, Permanent, Clearance or Permanent and Any Price Event<br><br>Always: This option will always print a ticket regardless of if there is a price change.<br><br>Clearance: Only print a ticket if on the specific date any clearance event is effective. So, for Clearance 2, that means today's date + 2 days, if the item on that day has a clearance going on, print the clearance ticket.<br><br>Promotion: Only print a ticket if on the specific date any promotion event is effective.<br><br>Regular or Clearance: Only print a ticket if on the specific date any Regular or Clearance event is getting effective.<br><br>Permanent: Only print a ticket if on the specific date any Regular even of getting effective.<br><br>Any Price Event (Promotion, Clearance or Permanent (Regular)): Based on the date selected, if any price event goes into effect (clearance, promotion or regular price a ticket. If no price event goes into effect, print nothing). | Always | Ticketing | String |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Maximum Ticket Quantity to Print | This is to determine the maximum ticket size to print in one command. This is used in auto ticket printing batch and ticketing dialogue. | 500 | Ticketing | Boolean |
| Auto Close Receipt | Values: 1-99<br><br>0: close the receipt immediately<br><br>1: close the receipt the end of day today<br><br>2: close the receipt end of day tomorrow<br><br>X: close end of day x days starting from today.<br><br>The batch program will auto close any transfer receipts, and marks all non-received containers to missing. Partially received containers will be marked as damaged. | 1 | Transfer Receiving | Integer |
| External Finisher Auto Receive | Values: Not Allowed, External Message, Date Driven<br><br>Not allowed will make the system work as today.<br><br>External message will receive the full external finisher delivery with a Source Type of 'Finisher', the moment an ASN transaction arrives that indicates that the delivery needs to be auto received.<br><br>Date Driven will look at a secondary store option (External Finisher Auto Receive number of Days) to determine how many days the transaction stays open before it is fully received. If it is set to 0, it will auto-receive on the ETA date. | Not Allowed | Transfer Receiving | Integer |
| External Finisher Auto Receive Number of Days | Values: 0-999<br><br>A batch program will auto receive any external finisher deliveries with a Source Type of 'Finisher', that have not been closed x-days after the ETA date or the create date depending on if the ETA date is set or not and if the auto receive external delivery parameter is set 0 means immediate receiving 1 means today (EOD) 2 means EOD tomorrow x means EOD x days starting from today | 0 | Transfer Receiving | Integer |

**Table 9-2 (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Store Auto Receive | Values: Not allowed / External message / Date Driven<br><br>Not allowed: Auto receiving is not allowed for the store.<br><br>External message: Receives the full store delivery the moment an ASN transaction arrives when the indicator on the ASN identifies this as an auto receive delivery and the Source Type is 'Store'. This parameter works with the Store Auto Receive screen.<br><br>Date Driven: Receives the delivery automatically when the date is reached. A second options, 'Store Auto Receive Number of Days' is used to determine how many days the transaction stays open before it is fully received. If it is set to 0, it will receive immediately when the transfer is shipped. This parameter works with the Store Auto Receive screen. | Not Allowed | Transfer Receiving | Integer |
| Store Auto Receive Number of Days | Values: 0-99<br><br>0: immediate receiving<br><br>1: end of day today<br><br>2: end of day tomorrow<br><br>X: end of day x days starting from today<br><br>The batch program will auto receive any transfers with a Source Type of 'Store' not previously closed x-days after they have been shipped. This parameter is only used when the Store Auto Receive parameter is enabled. | 0 | Transfer Receiving | Integer |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Store Transfer Default to ShopFloor Receive | Values: Yes/No<br><br>This parameter determines whether the receiving will default to receive inventory into the shop floor instead of automatically receiving into the back room or delivery bay when the source type is 'Store'.<br><br>Yes: The Transfer workflow will default to receive inventory into the shop floor when the Source Type is 'Store'. The shop floor inventory bucket will be incremented instead of the backroom or delivery bay bucket. If a capacity is defined for the item, the maximum shop floor quantity will equal the capacity; otherwise, the shop floor will be updated to the entire receipt amount.<br><br>If the capacity is used and if the receiving quantity is excess, the balance is incremented to delivery bay or back room depending on the Replenishment.<br><br>No: This will default to back room or delivery bay depending on the Replenishment -Delivery Bay Inventory parameter and all inventory will be automatically received into the back room or delivery bay when Source Type is 'Store'. Note: With the proper permissions, the user will still have the option to receive onto the shop floor while receiving the delivery by changing the default to Shop Floor.<br><br>Damaged inventory will not move to shop floor. | No | Transfer Receiving | Boolean |
| Store Transfer Receive Item Capacity | Values: Yes/No<br><br>This parameter will determine whether the capacity will be considered while receiving the deliveries.<br><br>Yes: If the value is set to YES, then while receiving and Source Type of 'Store', the capacity will be considered. For example: If capacity is 50, receipt is for 100 and Available SOH is 10 on shopfloor, then if this parameter is on and receive in shop floor is checked then 40 (un-damaged) will be moved to shop floor and rest to back room.<br><br>No: If the value is set to NO, then while receiving the capacity will not be considered. Damaged inventory will not move to shop floor. | No | Transfer Receiving | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Warehouse Auto Receive | Values: Not Allowed, External Message, Date Driven<br><br>This parameter will drive the following functionality.<br><br>Not Allowed will make the system work as today.<br><br>External message will receive the full warehouse delivery the moment an ASN transaction arrives that indicates that the delivery needs to be auto received and the Source Type is 'Warehouse'.<br><br>Date Driven will look at a secondary store option (Warehouse Auto Receive number of Days) to determine how many days the transaction stays open before it is fully received. If it is set to 0, it will auto-receive on the ETA date. | Not Allowed | Transfer Receiving | Integer |
| Warehouse Auto Receive Number of Days | Values: 0-99<br><br>0: immediate receiving<br><br>1: end of day today<br><br>2: end of day tomorrow<br><br>X: end of day x days starting from today<br><br>The batch program will auto receive any transfers with a Source Type of 'Warehouse' not previously closed x-days after they have been shipped. This parameter is only used when the Warehouse Auto Receive parameter is enabled. | 0 | Transfer Receiving | Integer |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Warehouse Default to ShopFloor Receive | Values: Yes/No<br><br>This parameter determines whether the receiving will default to receive inventory into the shop floor when source type is 'Warehouse' instead of automatically receiving into the back room or delivery bay when the source type is 'Warehouse'.<br><br>Yes: The Transfer receiving workflow will default to receive inventory into the shop floor when the Source Type is 'Warehouse'. The shop floor inventory bucket will be incremented instead of the backroom or delivery bay bucket. If a capacity is defined for the item, the maximum shop floor quantity will equal the capacity; otherwise, the shop floor will be updated to the entire receipt amount.<br><br>If the capacity is used and if the receiving quantity is excess, the balance is incremented to delivery bay or back room depending on the Replenishment - Delivery Bay Inventory parameter.<br><br>No: This will default to back room or delivery bay depending on the Replenishment -Delivery Bay Inventory parameter and all inventory will be automatically received into the back room or delivery bay when Source Type is 'Warehouse'. Note: With the proper permissions, the user will still have the option to receive onto the shop floor while receiving the delivery by changing the default to Shop Floor.<br><br>Damaged inventory will not move to shop floor. | No | Transfer Receiving | Boolean |
| Warehouse Receive Item Capacity | Values: Yes/No<br><br>This parameter will determine whether the capacity will be considered while receiving the deliveries.<br><br>Yes: If the value is set to YES, then while receiving and Source Type of 'Warehouse', the capacity will be considered. For example: If capacity is 50, receipt is for 100 and Available SOH is 10 on shopfloor, then if this parameter is on and receive in shop floor is checked then 40 (un-damaged) will be moved to shop floor and rest to back room.<br><br>No: If the value is set to NO, then while receiving the capacity will not be considered. Damaged inventory will not move to shop floor | No | Transfer Receiving | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Dispatch Validate | Values: Ship Direct, Ship Submit<br><br>Ship Direct: SIOCS will control all processes. The user will be able to go from create/edit directly to dispatch.<br><br>Ship Submit: This option will require the user to press the Submit button and require a specific press of the dispatch button. An additional option is that an external system will generate a dispatch message through a standard web service. | Ship Direct | Transfer Receiving | Integer |
| Pre-shipment Notification | Values: Yes/No<br><br>This parameter will drive the following functionality:<br><br>Yes: The system will publish a pre-shipment message.<br><br>No: The system will not publish a pre-shipment message. | No | Transfer Shipment | Boolean |
| Ship to Finisher Carrier Default | Values: Sender / Receiver / Third Party<br><br>Sender: Sender will be selected for Carrier Type on BOL<br><br>Receiver: Receiver will be selected for the Carrier type on BOL.<br><br>Third Party: Third Party will be selected for the Carrier type on the BOL. The type (drop down) will be defaulted to "Other".<br><br>When creating a store to Finisher transfer the Carrier Type on the BOL will default initially based upon this parameter. The user can still change this value and if so, that will be the value used on the transfer. | Third Party | Transfer Shipment | Integer |
| Ship to Store Carrier Default | Values: Sender / Receiver / Third Party<br><br>Sender: Sender will be selected for Carrier Type on BOL<br><br>Receiver: Receiver will be selected for the Carrier type on BOL.<br><br>Third Party: Third Party will be selected for the Carrier type on the BOL. The type (drop down) will be defaulted to "Other".<br><br>When creating a store to store transfer the Carrier Type on the BOL will default initially based upon this parameter. The user can still change this value and if so, that will be the value used on the transfer. | Third Party | Transfer Shipment | Integer |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
| --- | --- | --- | --- | --- |
| Ship to Warehouse Carrier Default | Values: Sender / Receiver / Third Party<br><br>Sender: Sender will be selected for Carrier Type on BOL<br><br>Receiver: Receiver will be selected for the Carrier type on BOL.<br><br>Third Party: Third Party will be selected for the Carrier type on the BOL. The type (drop down) will be defaulted to "Other".<br><br>When creating a store to WH transfer the Carrier Type on the BOL will default initially based upon this parameter. The user can still change this value and if so, that will be the value used on the transfer. | Third Party | Transfer Shipment | Integer |
| Auto Accept External Generated Request | Values: Yes/No<br><br>This parameter automatically approves the requested transfer and defaults the requested quantity to the accepted quantity for externally generated requests. | No | Transfers | Boolean |
| Auto Accept Store Transfer Request | Values: Yes/No<br><br>This parameter automatically approves the requested transfer and defaults the requested quantity to the accepted quantity for store to store requests. | No | Transfers | Boolean |
| Not After Date Default Days | Values: 0-999<br><br>This parameter adds a value to the current date and uses the value to default the 'Not After Date' when creating a transfer or requesting a transfer. Documents will be closed via batch when this date is reached. | 30 | Transfers | Integer |
| Manifest Customer Order Deliveries | Values: Yes/No<br><br>Yes: The Manifesting system will be called.<br><br>No: The Manifesting system will not be called. | No | Web Service Enablement | Boolean |
| Manifest RTV to Supplier | Values: Yes/No<br><br>Yes: The Manifesting system will be called for return to supplier.<br><br>No: The Manifesting system will not be called.<br><br>Note: The interface will still need to be implemented; this just is to determine if it will be called. | No | Web Service Enablement | Boolean |
| Manifest Transfer to Finisher | Values: Yes/No<br><br>Yes: The Manifesting system will be called for transfer to Finisher.<br><br>No: The Manifesting system will not be called.<br><br>Note: The interface will still need to be implemented; this just is to determine if it will be called. | No | Web Service Enablement | Boolean |

**Table 9-2    (Cont.) Store Admin Parameters**

| Options | Description | Default Value | Topic | Type |
|---|---|---|---|---|
| Manifest Transfer to Store | Values: Yes/No<br>Yes: The Manifesting system will be called for transfer to store.<br>No: The Manifesting system will not be called.<br>Note: The interface will still need to be implemented; this just is to determine if it will be called. | No | Web Service Enablement | Boolean |
| Manifest Transfer to Warehouse | Values: Yes/No<br>Yes: The Manifesting system will be called for transfer to warehouse<br>No: The Manifesting system will not be called.<br>Note: The interface will still need to be implemented; this just is to determine if it will be called. | No | Web Service Enablement | Boolean |
| OMS Customer Order Delivery Validation | Values: Yes/No<br>Yes: When confirming the delivery, the system will make a call out to an external system (such as an OMS) to validate the delivery status and delivery quantities before completing the dispatch.<br>No: external system will not be called. | No | Web Service Enablement | Boolean |
| OBCS Customer Order Delivery Validation | Values: Yes/No<br>Yes: SIOCS-OBCS Integration will be enabled<br>No: SIOCS-OBCS Integration will not be enabled | No | Web Service Enablement | Boolean |
| Sales Forecast Data | Yes: The web service for Sales Forecast Data will be called.<br>No: The web service for Sales Forecast Data will NOT be called. | No | Web Service Enablement | Boolean |
| Send Event Alert External | Values: Yes / No<br>Yes: Notification event alerts (non ad hoc notifications) will be sent externally via web service.<br>No: Notification event alerts (non ad hoc notifications) will not be sent externally. Web service is not called. | No | Web Service Enablement | Boolean |

# Permissions

**Table 9-3    Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access Ad Hoc Stock Count Tolerances | Admin | With this permission, the user will have access to the Ad hoc Stock Count Tolerance dialog. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access Admin | Admin | With this permission, the user will have access to the Admin menu. |
| Access Auto-Receive Stores | Admin | With this permission, the user will have access to the Auto Receive Stores admin dialog. |
| Access Barcode Processor | Admin | With this permission, the user will have access to the Barcode Processor dialog. |
| Access Buddy Stores | Admin | With this permission, the user will have access to the Buddy Store dialog. |
| Access Carrier Services | Admin | With this permission, the user can access the Carrier Service dialog to add and edit the carrier service data. |
| Access Carriers | Admin | With this permission, the user can access the Carrier dialog to add or edit the carrier data. |
| Access Code Info | Admin | With this permission, the user can access the Code Info dialog to add, edit and delete code information. |
| Access Container Lookup | Admin | With this permission, the user will have access to the Container Lookups dialog. |
| Access Credential Administration | Admin | With this permission, the user will have access to the Credential Administration Screen.<br><br>Without this permission, the user will not have access to the Credential Administration Screen. |
| Access Customer Order Picking Tolerances | Admin | With this permission, the user will have access to the Customer Order Picking Tolerance dialog. |
| Access Data Seed | Admin | With this permission the user will have the ability to start the data seeding job via the batch job admin. |
| Access Delivery Timeslot | Admin | User must have this permission for the Delivery Timeslot menu option to be available within the Data Setup menu.<br><br>With this permission the user will be able to do all operations on this screen. |
| Access Extended Attribute | Admin | With this permission, the Extended Attributes Menu option is displayed under Admin/Configuration and the user gets the ability to setup and assign extended attributes. |
| Access Extended Attribute Dept Assign | Admin | With this permission, the Assign Extended Attributes Menu option is displayed under Admin/Configuration and the user gets the ability to assign new extended attributes and remove the existing assignments. |
| Access Extended Attribute Setup | Admin | With this permission, the Setup Extended Attributes Menu option is displayed under Admin/Configuration and the user gets the ability to view and edit extended attributes. |
| Access Finisher Lookup | Admin | With this permission, the user will have access to the Finisher Lookup functionality. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Access Flexible Attributes | Admin | With this permission, the user will have access to the Custom Flexible Attributes dialog and can setup (Create) and Delete flexible attributes. Note that CFAs are not editable once created due to data integrity. |
| Access Future Price Events | Admin | With this permission, the user will have access to Future Price Events. |
| Access Inventory Adjustment Reasons | Admin | With this permission, the user will have access to the Inventory Adjustment Reason admin dialog and the ability to setup and maintain inventory adjustment reason codes. |
| Access Inventory Management | Admin | With this permission, the user will have access to the Inventory Management. |
| Access ISN Types | Admin | With this permission, user will be able to access the ISN Types dialog in the desktop application. |
| Access Item CFAs | Admin | With this permission, the CFAs will be available on Item Detail in Item Lookup |
| Access Item Lookup | Admin | With this permission, the user will have access to the Item Lookup functionality. |
| Access Item Scan Number Lookup | Admin | With this permission, the user can access the Item Scan Number Lookup. |
| Access Job Admin | Admin | With this permission, the user can access the Job admin dialog. |
| Access Job Scheduler | Admin | With this permission, the user can access the Job scheduler dialog. |
| Access Lookup | Admin | With this permission, the user will have access to functionality within Lookups. |
| Access MPS Staged Messages | Admin | User must have this permission in order to access the MPS Staged Messages screen to view or edit the inbound and outbound messages. |
| Access MPS Work Types | Admin | User must have this permission in order to Access or Edit the Worker Type settings. |
| Access Operational Issues | Admin | With this permission, the user can access the Operational Issues dialog. |
| Access Operational Views menu | Admin | On EICS, with this permission, the operational views main menu is displayed, and user is allowed to view the different operational views. |
| Access Package Size | Admin | With this permission the user will have access to the Package Size admin dialog. |
| Access POS Transaction Resolution List | Admin | On desktop application, with this permission, the user can access the Transaction Resolution dialogue. |
| Access Price Events | Admin | With this permission, 'View Price Events' button will be available in the Item Detail screen hence the user will be able to access to the Price History screen. |
| Access Printer Setup | Admin | With this permission, the user can access the printer setup dialog. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access Product Group Schedules | Admin | With this permission the user will have access to the Product Group Schedule dialog. |
| Access Product Groups | Admin | With this permission, the user will have access to the admin Product Group and Product Group Component functionality |
| Access Reports | Admin | With this permission, the user can access the Reports dialog. |
| Access RFID Locator | Admin | With this permission, the user can access the RFID Locator dialog in mobile. |
| Access Sequence Admin | Admin | With this permission the user will have the Sequence Admin menu option under Technical Maintenance menu. |
| Access Shipment Reasons | Admin | With this permission, the user will have access to the Shipment Reason admin dialog and the ability to setup and maintain shipment reason codes. |
| Access Shipping Receiving | Admin | With this permission, the user will have access to functionality within Shipping/Receiving. |
| Access SIOCS Managed Stores | Admin | With this permission, the user will have access to the SIOCS Managed Stores screen in EICS. |
| Access Store Administration | Admin | With this permission, the user will have access to Store Administration dialog. |
| Access Store Administration Default | Admin | With this permission, the user will have access to Store Administration default dialog. |
| Access Store Shipping Network | Admin | With this permission, the user will have access to the Store Shipping Network screen in the desktop application. |
| Access Sub Buckets | Admin | With this permission, the user can access the Sub bucket dialog. |
| Access Supplier CFAs | Admin | With this permission, the CFAs will be available on Supplier Detail in Supplier Lookup. |
| Access Supplier Lookup | Admin | With this permission the user will have access to Supplier Lookup and the Supplier Lookup menu option will appear in the drawer (main menu). |
| Access System Administration | Admin | With this permission, the user will have access to the 'System Administration' dialog. |
| Access Technical Maintenance | Admin | With this permission, the user will have access to Technical Maintenance dialog. |
| Access Transaction History | Admin | With this permission the user will have access to the Transaction History dialog. |
| Access Translation Setup | Admin | With this permission, the user can access the Translation Setup dialog. |
| Access Troubled Transaction | Admin | With this permission, the user can select the Troubled Transaction List from the Inventory Management menu. |
| Access UDA Print Setup | Admin | With this permission, the user can access the UDA Print Setup dialog in the desktop application. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access UDAs | Admin | With this permission, in Item Lookup the user will be able to search for an item by a search type of UDA.<br><br>Also, the UDAs will be available on Item Detail in Item Lookup. |
| Access UIN Label Setup | Admin | With this permission, the user can access the UIN Label Setup dialog in the desktop application. |
| Access Unit of Measure | Admin | User must have this permission for the Unit of Measure menu option to be available within the Configuration menu.<br><br>With this permission the user will be able to do all operations on this screen. (Add ""+"", Remove ""-"" and Edit). |
| All Stores Product Groups | Admin | With this permission, Product Groups for All Stores can be created as well as edited.<br><br>Without this permission, Product Groups cannot be created for all stores and they can only be viewed for all stores. |
| Allow Bulk Scan | Admin | With this permission, the user will have access to the Bulk Scan dialog.<br><br>The Bulk Scan menu option will be in the footer menu of applicable transaction item list screens.<br><br>The Bulk scan type will be available as a scan mode in the mode bar. |
| Batch Execution Delete | Admin | With this permission, the user will be able to delete the issues in bulk Import, Transaction Execution and Data Purge tasks via Operational Issues Dialog. |
| Create Item Scan Number | Admin | With this permission, the user is allowed to create a new item scan number in the system. |
| Create Notes | Admin | With this permission, the user will be able to add notes within the notes dialog. |
| Create Product Group Schedules | Admin | With this permission, the user can create new Product Group Schedules. |
| Create Product Groups | Admin | With this permission, the user will be able to create a new Product Group. |
| Create Translations | Admin | With this permission, the user can create new translations. |
| Delete Item Scan Number | Admin | With this permission, the user is allowed to delete an existing Item Scan Number. |
| Delete MPS Staged Messages | Admin | User must have this permission in order to delete the inbound and outbound messages. |
| Delete Product Group Schedules | Admin | With this permission, the user can delete Product Group Schedules. |
| Delete Product Groups | Admin | With this permission, the user can delete a Product Group. |
| Display Stock Locator | Admin | With this permission, the user will have access to Stock Locator within Item Lookup. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Edit Item Scan Number | Admin | With this permission, the user is allowed to edit an existing Item Scan Number. Applicable for webservice operation. |
| Edit Item Scan Number CDA | Admin | This is required for web service action to edit the Item Scan number CDAs. |
| Edit Job Schedules | Admin | With this permission the user can edit Job Schedules. |
| Edit POS Transaction | Admin | On desktop application, with this permission, the user can edit the troubled pos transaction message. |
| Edit Product Group Schedules | Admin | With this permission, the Product Group Schedule will be editable. |
| Edit Product Groups | Admin | With this permission, the user can Edit an existing Product Group which also includes adding and removing Product Group Components. |
| Edit UDA Print Setup | Admin | With this permission, the user can edit the UDA Print Setup values in the desktop application. |
| Update Resolution Status | Admin | With this privilege the Resolve/Reset button on the Troubled Transactions List screen will be displayed and enabled. |
| Update UIN Status | Admin | With this permission, the user can update the status of the UIN from the history screen. |
| View UIN History | Admin | With this permission, the user will be able to access the UIN Lookup feature and view the history in EICS. |
| Warehouse Inventory Access | Admin | With this permission, the warehouse inventory details will be included in the Stock Locator Item Lookup section. |
| Access Area | Area | With this permission, the user will have access to the Area dialog. |
| Confirm Area | Area | With this permission, the user will be able to Confirm an Area. |
| Create Area | Area | With this permission, the user will be able to create Areas. |
| Delete Area | Area | With this permission, the user will be able to delete an Area. |
| Edit Area | Area | With this permission, the user will be able to edit active Areas. |
| Access Customer Details | Customer Order | With this permission the user will have access to the Customer Details (name, address, and so on) associated with the customer order. |
| Access Customer Order | Customer Order | With this permission, the user will have access to Customer Orders dialog. |
| Access Customer Order Delivery | Customer Order | With this permission, a user can access the Customer Order Delivery dialog. |
| Access Customer Order Delivery Attribute | Customer Order | With this permission, the user will have access to Customer Order Delivery Attributes. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access Customer Order Management | Customer Order | With this permission, the user will have access to the Customer Order Management operations within the drawer/menu. |
| Access Customer Order Pick | Customer Order | With this permission, the user will have access to the Customer Order Picking dialog. |
| Access Customer Order Reverse Pick | Customer Order | With this permission, the user will have access to the Customer Order Reverse Picking dialog. |
| Cancel Submit Customer Order Delivery | Customer Order | With this permission, the user will be able to Cancel Submit a "Submitted" status Customer Order Delivery that is a "web order. |
| Confirm Customer Order Pick | Customer Order | With this permission, the user can Confirm a customer order pick. |
| Confirm Customer Order Reverse Pick | Customer Order | With this permission, the user will be able to Confirm a customer order reverse pick. |
| Create Customer Order Delivery | Customer Order | With this permission, the user can create a new delivery for a Customer Order that is a "web order". Used in conjunction with Create Customer Order Delivery for Shipment or Create Customer Order Delivery for Pickup permission. This permission must also exist in order to create a delivery that is not a web order from an external system. |
| Create Customer Order Delivery for Pickup | Customer Order | With this permission, the user will be able to create Customer Order Deliveries which are of type store pickup. Used in conjunction with the Create Customer Order Delivery permission. |
| Create Customer Order Delivery for Shipment | Customer Order | With this permission, the user will be able to create Customer Order Deliveries which are of type store shipment. Used in conjunction with the Create Customer Order Delivery permission. |
| Create Customer Order Pick | Customer Order | With this permission, the user will be able to create customer order picks. Picks can be created from within a Customer Order (for a single customer order pick). Picks can be created from within Customer Order Picking. |
| Create Customer Order Reverse Pick | Customer Order | With this permission, the user will be able to create customer order reverse picks. |
| Delete Customer Order Delivery | Customer Order | With this permission, the user will be able to delete a customer order delivery that is a "web order". |
| Delete Customer Order Pick | Customer Order | With this permission, the user will be able to delete a customer order pick. |
| Delete Customer Order Reverse Pick | Customer Order | With this permission, the user will be able to delete a customer order reverse pick. |
| Dispatch Customer Order Delivery | Customer Order | With this permission, the user can Dispatch a Customer Order Delivery that is a "web order". |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Dispatch Incomplete Customer Order Delivery | Customer Order | When dispatching a delivery that requires full delivery (Allow Partial Delivery = 'No'), the entire delivery must be delivered in full.<br><br>With this permission, the user will get a warning message and will be able to continue the dispatch without full delivery.<br><br>Without this permission the delivery must be in full. |
| Edit Customer Order BOL | Customer Order | With this permission, the user will be able to edit the details of the Bill of Lading associated with a customer order delivery. This is done in the Edit Delivery screen.<br><br>User must also have Edit Customer Order Delivery permission. |
| Edit Customer Order CFA | Customer Order | With this permission, the user will be able to capture CFAs for a customer order. |
| Edit Customer Order Delivery | Customer Order | With this permission, the user can edit an existing delivery for a Customer Order that is a "web order". |
| Edit Customer Order Delivery Attribute | Customer Order | With this permission, the user can edit the Customer Order Delivery Attributes. |
| Edit Customer Order Delivery CFA | Customer Order | With this permission, the user will be able to capture CFAs for a Customer Order Delivery |
| Edit Customer Order Pick | Customer Order | With this permission, the user will be able to edit active customer order picks. |
| Edit Customer Order Pick CFA | Customer Order | With this permission, the user will be able to capture CFAs for a Customer Order Pick. |
| Edit Customer Order Reverse Pick | Customer Order | With this permission, the user will be able to edit active reverse picks |
| Edit Customer Order Reverse Pick CFA | Customer Order | With this permission, the user will be able to capture CFAs for a Customer Order Reverse Pick |
| Edit Quantity Delivery | Customer Order | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity.User must also have Edit Customer Order Delivery permission as well.Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit Quantity Picking | Customer Order | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity.User must also have Edit Customer Order Pick permission. Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit Quantity Reverse Picking | Customer Order | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity.User must also have Edit Customer Order Reverse Pick permission. Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Item Substitution For Picking | Customer Order | With this permission, the user will have access to the Item Substitution dialog within Customer Order Picking. |
| Reject Customer Order | Customer Order | With this permission, the user will be able to Reject a Customer Order. |
| Submit Customer Order Delivery | Customer Order | With this permission, the user will be able to Submit a Customer Order Delivery that is a "web order". Used in conjunction with Modify Customer Order Delivery permission. |
| View Customer Order BOL | Customer Order | With this permission, the user can access and view the details of the Bill of Lading associated with a customer order delivery. |
| Container Items Limited To | Data | With this permission, the user can access data criteria associated with adding items to a container. This is selected by each individual criterion. |
| Counting Method | Data | With this permission, the user can access specific counting methods. This is selected by each individual counting method. |
| Display List Diff Types | Data | With this permission, the user can access the display of diff types. This is selected by each individual diff type. |
| Inventory Adjustment Reason Code | Data | With this permission, the user can access specific inventory adjustment reason codes. This is selected by each individual reason code. |
| Item Basket Types | Data | With this permission, the user can access specific item basket types. This is selected by each individual basket types. |
| Location Types | Data | With this permission, the user can access specific location types. This is selected by each individual location type. |
| Print Format Type | Data | With this permission, the user can access specific print format types. This is selected by each individual format type. |
| Product Group Type | Data | With this permission, the user can access specific product group types. This is selected by each individual product group type. |
| Role Type | Data | With this permission, the user can access specific role types. This is selected by each individual role type. |
| RTV Reason Code | Data | With this permission, the user can access specific RTV reason codes. This is selected by each individual reason code. |
| RTV Shipment Reason Code | Data | With this permission, the user can access specific RTV shipment reason codes. This is selected by each individual reason code. |
| Scan List Type | Data | With this permission, the user can access specific scan list types. This is selected by each individual scan list type. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Shelf Adjustment Type | Data | With this permission, the user can access specific shelf adjustment types. This is selected by each individual adjustment type. |
| Shelf Replenishment Type | Data | With this permission, the user can access specific shelf replenishment types. This is selected by each individual type. |
| Store Order Delivery Timeslots | Data | With this permission, the user can access specific delivery timeslots. This is selected by each individual timeslot. |
| Transaction Type | Data | With this permission, the user can access specific transaction types. This is selected by each individual transaction type. |
| Transfer Destination Type | Data | With this permission, the user can access specific transfer destination types. This is selected by each individual transfer destination type. |
| Transfer Shipment Reason Code | Data | With this permission, the user can access specific transfer destination types. This is selected by each individual transfer destination type. |
| Access Adjust Container DSD Receiving | DSD Receiving | With this permission, the user will be able to adjust the container in a delivery. |
| Access Confirm Container DSD Receiving | DSD Receiving | With this permission, the user can confirm the container receipt from the supplier. |
| Access Confirm DSD Receipt | DSD Receiving | With this permission, the user will be able to confirm the Direct store delivery. |
| Access Create Container | DSD Receiving | With this permission, the user can create a new container in the DSD receipt. |
| Access Delete Container | DSD Receiving | With this permission, the user can delete a container. |
| Access Delete Receipt | DSD Receiving | With this permission, the user will be able to delete a direct store delivery. |
| Access Document DSD Receiving | DSD Receiving | With this permission, the user will be able to select the PO to apply items on the receipt.User must also have Edit container and Modify container permissions in order to do this operation. |
| Access DSD Receiving | DSD Receiving | With this permission, the user will have access to the DSD Receiving dialog. |
| Access DSD Receiving Ext. Attribute | DSD Receiving | With this permission, the user will be able to access the DSD Receiving Ext. Attributes. |
| Access Purchase Order | DSD Receiving | With this permission, the user will be able to access the Purchase Orders. |
| Access Reject Delivery | DSD Receiving | With this permission, the user will be able to reject a direct store delivery. |
| Activate Container Edits | DSD Receiving | User needs this permission combined with the other respective permission(s) to update any information on the container. |
| Add Unexpected Item | DSD Receiving | With this permission, the user will be able to add the unexpected items to the container. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Allow ASN Over Receiving | DSD Receiving | For PO's with an ASN:<br><br>With this permission, when entering a received quantity, it is OK, to exceed the ASN quantity.<br><br>Without this permission, the received quantity cannot exceed the ASN quantity. |
| Allow DSD Receiving With PO | DSD Receiving | With this permission, the user will be able to create a receipt against a PO but without ASN. |
| Allow DSD Receiving Without PO | DSD Receiving | With this permission, the user will be able to create a receipt without PO. |
| Allow PO Over Receiving | DSD Receiving | For PO's with and without an ASN:<br><br>With this permission, when entering a received quantity, it is OK, to exceed the PO quantity.<br><br>Without this permission, the received quantity cannot exceed the PO quantity. |
| Allow Receiving Damages | DSD Receiving | With this permission, the user will be able to receive damaged items and make all remaining quantity to be received as damaged.User must also have Edit container and Modify container permissions in order to do this operation. |
| Default Qty in All Containers | DSD Receiving | With this permission, the user will be able to default the received quantity for all the containers in the delivery. |
| Default Qty in Container | DSD Receiving | With this permission, the user will be able to default the received quantity with the remaining quantity in the container.User must also have permission to 'Modify container' in order to do this operation. |
| Display Expected Quantity | DSD Receiving | With this permission, the user will be able to view the expected quantity during the receipt. |
| Edit Container | DSD Receiving | With this permission, the user can edit the container line items, modify the line item quantities or delete the item.User must also have permission to 'Modify container' in order to do this operation. |
| Edit Container CFA | DSD Receiving | With this permission, the user will be able to capture CFAs for a Container in DSD Receiving. |
| Edit Container Info DSD Receiving | DSD Receiving | With this permission, the user will be able to edit the container information.User must also have permission to 'Modify container' in order to do this operation. |
| Edit Delivery CFA | DSD Receiving | With this permission, the user will be able to capture CFAs for a DSD. |
| Edit Delivery Info | DSD Receiving | With this permission, the user will be able to edit the delivery information. |
| Edit DSD Receiving Ext. Attribute | DSD Receiving | With this permission, the user will be able to edit the DSD Receiving Ext. Attributes. |
| Edit Quantity | DSD Receiving | With this permission, the user will be able to edit the quantity of the item in the delivery.<br><br>User must also have Edit container and Modify container permissions in order to do this operation. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Override Not After Date Check | DSD Receiving | With this permission, the user can override the not after date check. This permission will allow the user to receive delivery where the receipt date passed the not after date in PO. |
| Override Supplier Discrepancies | DSD Receiving | With this permission, the user will be able to override the supplier discrepancies when quantity is validated. |
| Receive Direct Delivery on Shop Floor | DSD Receiving | With this permission, the user will be able to receive delivery on shop floor directly. |
| Access Inventory Adjustment | Inventory Adjustments | With this permission a user will have access to the Inventory Adjustment dialog. |
| Access Inventory Adjustment Attribute | Inventory Adjustments | With this permission a user will have access to the Inventory Adjustment Attributes. |
| Complete Inventory Adjustment | Inventory Adjustments | With this permission, the user can Confirm an inventory adjustment. User must also have data permissions for each adjustment reason on the adjustment. |
| Create Inventory Adjustment | Inventory Adjustments | With this permission, the user can create a new inventory adjustment. This permission must exist as well for a user to copy a "completed" inventory adjustment. User must also have data permissions for each adjustment reason on the adjustment. |
| Delete Inventory Adjustment | Inventory Adjustments | With this permission, the user will be able to delete an inventory adjustment. User must also have data permissions for each adjustment reason on the adjustment. |
| Edit Inventory Adjustment | Inventory Adjustments | With this permission, the user will be able to edit existing inventory adjustments. User must also have data permissions for each adjustment reason on the adjustment. |
| Edit Inventory Adjustment Attribute | Inventory Adjustments | With this permission, the user will be able to edit existing inventory adjustment attributes. |
| Edit Inventory Adjustment CFA | Inventory Adjustments | With this permission, the user will be able to capture CFAs for an inventory adjustment. |
| Edit Quantity | Inventory Adjustments | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity. User must also have Edit Inventory Adjustment permission as well as data permissions for each adjustment reason on the adjustment. |
| Access Item Basket | Item Basket | With this permission, the user will have access to the Item Basket dialog. |
| All Stores Item Basket | Item Basket | With this permission, Item Baskets for All Stores can be created as well as edited. Without this permission, Item Baskets cannot be created for all stores and they can only be viewed for all stores. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Confirm Item Basket | Item Basket | With this permission, the user will be able to Confirm an Item Basket. |
| Create Item Basket | Item Basket | With this permission, the user will be able to create Item Baskets. |
| Delete Item Basket | Item Basket | With this permission, the user will be able to delete an Item Basket. |
| Edit Item Basket | Item Basket | With this permission, the user will be able to edit active Item Baskets. |
| Edit Item Basket CFA | Item Basket | With this permission, the user will be able to capture CFAs on an Item Basket. |
| Edit Quantity Item Basket | Item Basket | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity.<br><br>User must also have Edit Item Basket permission.<br><br>Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Import Item Basket | Item Basket | With this permission, the user will have the Import Item Basket menu option within an Item Basket on mobile. |
| Investigate Item Basket | Item Basket | With this permission, the investigate menu option in item lookup will be available and the user will be able to add an item to an existing item basket or create a new item basket if one doesn't exist for investigation in item lookup.<br><br>Without this permission, the investigate menu option in item lookup will not be available. |
| Access Notifications | Notifications | With this permission, the bell notification icon will be displayed in the drawer/menu as well as on the Open Transactions header.<br><br>The view on the notification will also be displayed. |
| Customer Order Pick Reminder | Notifications | With this permission, the user will be notified, if the pick list has been created but not actioned. |
| Customer Order Reauthorization | Notifications | With this permission the user will receive a notification when the payment reauthorization for a customer order is successful. |
| Customer Order Receipt | Notifications | With this permission, the user will be notified when customer orders are received. |
| Customer Order Reminder | Notifications | With this permission, the user will be notified when the customer order has not been fulfilled. |
| Damaged Delivery | Notifications | With this permission, the user will be notified when the delivery includes damaged items. |
| Display External Scanner Notifications | Notifications | With this permission, the user will be notified with a popup that appears when a scanner is connected or disconnected or has a low battery for that event. |
| Display Notification Warning | Notifications | With this permission, the user will receive a notification warning when a new notification is created / inserted into the system. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Finisher Delivery Unable to Auto-Receive | Notifications | With this permission, the user will receive a notification warning when a finisher delivery is not auto received. |
| Finisher UIN Discrepancy | Notifications | With this permission, the user will be notified when a finisher return received quantity does not match the number of serial numbers on the return.<br>Without this permission, the user will not be notified. |
| Misdirected Container | Notifications | With this permission, the user will be notified when a container has been received in another location. |
| New Customer Order | Notifications | With this permission, the user will be notified when customer orders are created. |
| New Customer Order Reverse Pick | Notifications | With this permission, the user will be notified when a new cross channel customer order reverse picks arrives. |
| Over Received Quantity | Notifications | With this permission, the user will be notified when the number of pre-populated serial numbers exceeds the received quantity.<br>Without this permission, the user will not be notified. |
| Receiving UIN Discrepancy | Notifications | With this permission, the user will be notified when the number of pre-populated serial numbers does not match the received quantity.<br>Without this permission, the user will not be notified. |
| RTV Request Expiration Approaching | Notifications | With this permission, the user will be notified if the supplier return request expiration date is approaching. |
| RTV Unavailable request quantity | Notifications | With this permission, the user will be notified if there is not enough inventory in the unavailable bucket to send back to supplier from a return request. |
| Shipped Delivery Overdue | Notifications | With this permission, the user will be notified when the shipped delivery has not been received and has passed the expected date. |
| Store Delivery Unable to Auto-Receive | Notifications | With this permission, the user will be notified when a store delivery has discrepancies and cannot be auto received. |
| Store Receiving Over/Under | Notifications | With this permission, the user will be notified when a store transfer has over/under received quantities. |
| Transfer Request | Notifications | With this permission, the user will be notified when a transfer request is created. |
| Transfer Request Approved | Notifications | With this permission, the user will be notified when a transfer request is approved. |
| Transfer Request Expiration Approaching | Notifications | With this permission, the user will be notified when a transfer request has not been approved and the request is about to expire. This is based on the not after date set. |
| Transfer Request Rejected | Notifications | With this permission, the user will be notified when a transfer request is rejected. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Transfer Unavailable Request Quantity | Notifications | With this permission, the user will be notified when the requested quantity is no longer available at the requested source location. |
| UIN Items on Incoming ASN Failed | Notifications | With this permission, the user will be notified if an Auto Generated SN item is on the ASN with pre-generated numbers when processing thru the RIB.<br>Without this permission, the user will not be notified. |
| Unexpected UIN (Store Changed) | Notifications | With this permission, the user will be notified when UINs are discovered at a store where they should not be.<br>Without this permission, the user will not be notified. |
| Warehouse Delivery Unable to Auto-Receive | Notifications | With this permission, the user will be notified when the delivery includes pre-populated serial numbers and cannot be automatically received. |
| Send Transaction Notification | Open Transactions | With this permission the user will be able to send a notification. This is done from within the Open Transactions dialog; the user will be able to swipe an open transaction and send a notification. |
| View Transactions | Open Transactions | With this permission the user will have the potential to view all the open transactions for the user's store (depending on data permissions).<br>Without this permission, the Open Transactions dialog will still display (just without the transactions listed); however, the list of open transactions will be empty. |
| Access Out of Stock Lookup | Operational Views | On EICS, with this permission, the Out of Stock operational view menu is displayed and user is allowed to view the out of stock operational view. |
| Access New Received Items | Operational Views | On EICS, with this permission the user will be able to access the Access New Items view in Operational Views. |
| Access Expiring Items Lookup | Operational Views | On EICS, with this permission the user will be able to access the Expiring Items view in Operational Views. |
| Access Stock Counts - Ready to Authorize | Operational Views | On EICS, with this permission the user will be able to access the Stock Counts - Ready to Authorize view in Operational Views. |
| Access Shopfloor Out of Stock | Operational Views | On EICS, with this permission the user will be able to access the Access Shopfloor Out of Stock view in Operational Views. |
| Accept RTV | RTV | With this permission, the user will be able to approve a return request.<br>User must also have data permissions for each return reason on the return. |
| Access RTV | RTV | With this permission, a user will have access to the RTV dialog and the RTV menu option will appear in the Drawer. |

ORACLE®

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Activate RTV Edits | RTV | User is required to have Activate RTV Edits permission for Edit RTV or Edit RTV Info to activate. That is, the changes made on the RTV, because the user had Edit RTV and Edit RTV Info permissions, cannot be committed unless the user has Activate RTV Edits permission. |
| Add Items To RTV | RTV | With this permission, the user will be able to add items to a return. |
| | | User must also have Modify RTV and Edit RTV permissions as well as data permissions for each return reason on the RTV |
| Close RTV | RTV | With this permission, the user will be able to close an RTV. |
| | | User must also have data permissions for each return reason on the return. |
| Create RTV | RTV | With this permission, the user can create a new return. |
| | | User must also have data permissions for each return reason on the return. |
| Delete RTV | RTV | With this permission, the user can delete a return. |
| | | User must also have data permissions for each return reason on the return. |
| Edit Quantity | RTV | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity. |
| | | User must also have Modify RTV and Edit RTV permissions as well as data permissions for each return reason on the RTV. |
| | | Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit RTV | RTV | With this permission, the user will be able to edit existing RTV like line item details, qty, and so on. |
| | | User must also have data permissions for each Return reason on the RTV. |
| Edit RTV CFA | RTV | With this permission, the user will be able to capture CFAs for an RTV document. |
| Edit RTV Info | RTV | With this permission, the user will be able to edit the header information of an RTV. |
| Reject RTV | RTV | With this permission, the user will be able to reject a return request. |
| | | User must also have data permissions for each return reason on the return. |
| Access RTV Shipment | RTV Shipment | With this permission, the user can access shipments from RTV requests. |
| | | Without this permission, the user will not be able to access the shipments from RTV requests. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access RTV Shipment Attribute | RTV Shipment | With this permission, the Attributes tab on the RTV Shipment Container - Item Detail screen is displayed and enabled. |
| Activate RTV Shipment Container Edits | RTV Shipment | With this permission, the user will be allowed to commit any changes made to the shipment. |
| Add Unexpected Items to RTV Shipment | RTV Shipment | With this permission, the user will be allowed to add items that are not present in the RTV Document, into the shipment.<br><br>User must also have Modify Container and Edit Container permission as well as data permissions for each return reason on the container. |
| Adjust Carrier | RTV Shipment | With this permission, the user will be able to update the BOL details of a shipment even after at least one container has been confirmed. |
| Adjust Container RTV Shipment | RTV Shipment | With this permission, the user will be allowed to bring the container back to editable status. User must also have data permissions for each return reason on the container. |
| Allow over shipping RTV Shipment | RTV Shipment | With this permission, the user will be allowed to go over Approved quantity in the RTV document<br><br>User must also have Modify Container and Edit Container permission as well as data permissions for each return reason on the container.<br><br>Without this permission, the user will not be allowed to enter qty more than Approved qty. |
| Cancel Submit RTV Shipment | RTV Shipment | With this permission, the user can cancel submit RTV shipments. |
| Close RTV Shipment | RTV Shipment | With this permission, the user can close RTV shipments.<br><br>Without this permission, the user will not be able to close RTV shipments. |
| Confirm RTV Shipment Container | RTV Shipment | With this permission, the user can confirm containers in the shipments. User must also have data permissions for each return reason on the container. |
| Create RTV Shipment | RTV Shipment | With this permission, the user can create shipments for RTV requests. |
| Create RTV Shipment Container | RTV Shipment | With this permission, the user will be allowed to create a container for the shipment. |
| Default Items to RTV Shipment | RTV Shipment | With this permission, the user will be allowed to add items from the RTV document into shipment. User must also have data permissions for each return reason on the shipment.<br><br>Without this permission, the user will not be allowed to add items from the RTV document into shipment |
| Delete RTV Shipment | RTV Shipment | With this permission, the user can delete RTV shipments. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Delete RTV Shipment Container | RTV Shipment | With this permission, the user can delete containers in the shipments. |
| | | User must also have data permissions for each return reason on the container. |
| Dispatch Shipment | RTV Shipment | With this permission, the user can dispatch RTV shipments. |
| Edit Container CFA | RTV Shipment | With this permission, the user will be able to capture CFAs for a container in RTV shipment. |
| Edit Container RTV Shipment | RTV Shipment | With this permission, the user will be allowed to edit the line item details, update qty, remove item, restore item, cancel the current edits, and so on, |
| | | User must also have Modify Container permission as well as data permissions for each return reason on the container. |
| | | Without this permission, the user will not be allowed to edit any line item details, update qty, remove item, restore item, cancel the current edits, and so on. |
| Edit Quantity | RTV Shipment | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity. |
| | | User must also have Modify Container and Edit Container permission as well as data permissions for each return reason on the container. |
| | | Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit RTV Container Info | RTV Shipment | With this permission, the user will be allowed to edit the container header details. |
| | | User must also have Activate Container Edits permission. |
| Edit RTV Shipment Attribute | RTV Shipment | With this permission, the user will be allowed to edit the RTV Shipment Attributes. |
| Edit RTV Shipment BOL | RTV Shipment | With this permission, the user will be allowed to edit the shipment BOL details. |
| Edit RTV Shipment Info | RTV Shipment | With this permission, the user will be allowed to edit the shipment header details. |
| Edit Shipment CFA | RTV Shipment | With this permission, the user will be able to capture CFAs on an RTV shipment. |
| Submit RTV Shipment | RTV Shipment | With this permission, the user can submit RTV shipments. |
| View RTV Shipment BOL | RTV Shipment | With this permission, the user will be allowed to view the shipment BOL details. |
| Access Role Maintenance | Security | User must have this permission for the Role Maintenance menu option to be available under Security in EICS. |
| Access Security | Security | With this permission the user will have access to the Security dialog in EICS. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access User Maintenance | Security | User must have this permission for the User Assignment menu option to be available under Security in EICS. |
| Delete Role | Security | User must have this permission in order to delete roles. |
| Delete User | Security | User must have this permission in order to delete user profiles. |
| Edit User | Security | User must have this permission in order to assign roles and stores to a user. |
| Access Scan List | Shelf Replenishment | With this permission, the user will have access to Item Scan List dialog. |
| Access Shelf Adjustment | Shelf Replenishment | With this permission, the user will have access to Shelf Adjustment dialog. |
| Access Shelf Replenishment | Shelf Replenishment | With this permission, the user will have access to the in-store replenishment dialog. |
| Confirm Shelf Adjustment | Shelf Replenishment | With this permission, the user can confirm the shelf adjustment. |
| Confirm Shelf Replenishment | Shelf Replenishment | With this permission, the user can confirm the replenishment pick. |
| Create Scan List | Shelf Replenishment | With this permission, the user can create a new item scan list. User must also have data permissions for each scan list type to create a new scan list of that type. |
| Create Shelf Adjustment | Shelf Replenishment | With this permission, the user can create a new shelf adjustment. |
| Create Shelf Replenishment | Shelf Replenishment | With this permission, the user can create a new shelf replenishment pick. User must have the data permission for each shelf replenishment pick type to do this operation. |
| Default Shelf Replenishment Quantity | Shelf Replenishment | With this permission, the user can default the quantity on replenishment pick. The user must also have the Edit replenishment permission to do this operation. |
| Delete Scan List | Shelf Replenishment | With this permission, the user will be able to delete a scan list. |
| Delete Shelf Adjustment | Shelf Replenishment | With this permission, the user can delete the shelf adjustment. |
| Delete Shelf Replenishment | Shelf Replenishment | With this permission, the user can delete the shelf replenishment pick. |
| Edit Scan List | Shelf Replenishment | With this permission, the user will be able to edit and save the scan list. |
| Edit Scan List CFA | Shelf Replenishment | With this permission, the user will have the ability to capture CFAs for a Scan List. |

ORACLE®

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Edit Scan List Quantity | Shelf Replenishment | With this permission, the user will be able to edit the quantity on the UI using the widget. |
| | | User must also have the Edit Scan List permission to do this. |
| Edit Shelf Adjustment | Shelf Replenishment | With this permission, the user can edit and save the shelf adjustment. |
| Edit Shelf Adjustment CFA | Shelf Replenishment | With this permission, the user will have the ability to capture CFAs for a Shelf Adjustment. |
| Edit Shelf Adjustment Quantity | Shelf Replenishment | With this permission, the user can edit the quantity using the widget on the UI. |
| Edit Shelf Replenishment | Shelf Replenishment | With this permission, the user will be able to edit the existing shelf replenishment pick. |
| Edit Shelf Replenishment CFA | Shelf Replenishment | With this permission, the user will have the ability to capture CFAs for Replenishment Pick List. |
| Edit Shelf Replenishment Quantity | Shelf Replenishment | With this permission, the user can edit the quantity using the quantity widget on the UI. |
| | | Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Replenishment Item Substitution | Shelf Replenishment | With this permission, the user can substitute the pick items. |
| | | The user must also have the Edit replenishment permission to do this operation. |
| Access Stock Count | Stock Counts | With this permission, the user can access the Stock Counts dialog. |
| Access Stock Count Attribute | Stock Counts | With this permission, the Attributes tab on the Stock Counts - Item Detail screen is displayed and enabled. |
| Apply Late Sales | Stock Counts | With this permission, Apply Late Sales button will be available to the user in the Stock Count Authorization Detail screen. |
| Complete Child Stock Count | Stock Counts | With this permission, the user can complete the child stock count. |
| Confirm Authorization Stock Count | Stock Counts | With this permission, the user can confirm the authorization. |
| Create Ad Hoc Stock Count | Stock Counts | With this permission, the user can create a new adhoc stock count. |
| Delete Stock Count | Stock Counts | With this permission, the user can delete a stock count. |
| Edit Adhoc Stock Count | Stock Counts | With this permission, the user can edit the ad hoc stock count. |
| Edit Adhoc Stock Count Lock | Stock Counts | With this permission, the user will have the ability to enable and disable the Adhoc Stock count Lock for an adhoc stock count. |
| Edit Authorizaton Stock Count | Stock Counts | With this permission, the user can access the Stock count authorization dialog on the desktop. |
| | | Ability to apply late sales. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Edit Stock Count Attribute | Stock Counts | With this permission, extended attributes can be added/removed: Add Attributes and Remove Attributes (Trash can) buttons will be available on the Attributes tab of the Stock Counts - Item Detail screen. |
| Edit Stock Count CFA | Stock Counts | With this permission, the user will have the ability to capture CFAs in Stock Count and Recount. |
| Edit Stock Count Quantity | Stock Counts | With this permission, the user can edit the quantity using the quantity widget. |
| Edit Unit Amount Stock Count | Stock Counts | With this permission, the user is allowed to do update for the stock counts that are of type unit and amount |
| Edit Unit Stock Count | Stock Counts | With this permission, the user can update the stock count of unit or problem line stock count types. |
| Rejected Item Stock Count | Stock Counts | With this permission, the user will have access to the Rejected Items dialog. |
| Snapshot Stock Count | Stock Counts | With this permission, the user is allowed to take a snapshot at the master stock count level. |
| Snapshot Stock Count Child | Stock Counts | With this permission, the user is allowed to take the snapshot at the child stock count level. |
| Stock Count Import Basket | Stock Counts | With this permission, the user will have the Import Item Basket footer menu option in the Bulk Scan screen within Stock Counts. |
| Update Authorization Quantity | Stock Counts | With this permission, the user can update the authorization quantity and default the last count quantity to authorized quantity while in the process of authorization. |
| View Variance | Stock Counts | With this permission, the user will be able to view the variance (difference) between the snapshot quantity and the total quantity counted during counting. |
| Access Store Orders | Store Order | With this permission, the user will have access to the Store Orders dialog. |
| Approve Store Orders | Store Order | With this permission, the user will be able to Approve a Store Order. |
| Create Store Orders | Store Order | With this permission, the user will be able to create Store Orders. |
| Delete Store Orders | Store Order | With this permission, the user will be able to delete a Store Order. |
| Display Sales Forecast | Store Order | With this permission, the Sales Forecast on the Sales Data screen will be displayed. |
| Display Sales History | Store Order | With this permission, the Sales History on the Sales Data screen will be displayed. |
| Display Store Order Cost | Store Order | With this permission the Total Estimated Cost will be displayed on the Info screen in store orders. The unit cost will be displayed on the Store Orders Item Detail. The Refresh Cost button will be displayed in the Store Order Items footer menu. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
| --- | --- | --- |
| Edit Quantity | Store Order | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity.<br><br>User must also have Edit Store Orders permission.<br><br>Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit Store Order CFA | Store Order | With this permission, the user will be able to edit CFAs on the store order. The Edit CFA menu option will be available in the footer menu. |
| Edit Store Orders | Store Order | With this permission, the user will be able to edit active Store Orders. |
| Access Format Assignment | Ticketing | With this permission, the user is allowed to access the format assignment dialog in the desktop application. |
| Access Print Format | Ticketing | With this permission, the user is allowed to access the ticket print format dialog. |
| Access Print Item | Ticketing | With this permission, the user is allowed to access the print item dialog. |
| Access Ticket List | Ticketing | With this permission, the user can access the Ticket List. |
| Access Ticket Template Upload | Ticketing | With this permission, the user can access the Upload Ticket Templates screen to upload the ticketing layout. |
| Allow Override Ticket Price | Ticketing | With this permission, the user can override the ticket price on the ticket detail screen. |
| Create Format Assignment | Ticketing | With this permission, the user is allowed to create a new item basket based format assignment. |
| Create Ticket | Ticketing | With this permission, the user is allowed to create a new ticket in the ticketing dialog. |
| Delete Format Assignment | Ticketing | With this permission, the user is allowed to delete a format assignment. |
| Delete Ticket | Ticketing | With this permission, the user is allowed to delete a ticket in the ticketing dialog. |
| Delete Ticket Template | Ticketing | With this permission, the user is allowed to delete a ticket template screen. |
| Edit Format Assignment | Ticketing | With this permission, the user is allowed to edit an existing format assignment. |
| Edit Ticket | Ticketing | With this permission, the user can edit an existing ticket. |
| Print Ticket | Ticketing | With this permission, the user can print the tickets in the ticketing dialog. |
| Print Tickets from Container Items | Ticketing | With this permission, the user is allowed to generate and print tickets from the container items screen both in transaction and lookup |
| Accept Transfer Request | Transfer | With this permission, the user will be able to accept a transfer request. |

**Table 9-3　(Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Access Transfer | Transfer | With this permission, a user will have access the transfers. |
| Access Transfer context | Transfer | With this permission, a user will be able to view the Context type details in a transfer. |
| Access Transfer Request | Transfer | With this permission, a user will have access to the Transfer dialog in the application |
| Approve Transfer | Transfer | With this permission, the user will be able to approve a transfer document. |
| Close Transfer | Transfer | With this permission, the user will be able to close a transfer. |
| Create Request | Transfer | With this permission, the user will be able to create a transfer request. |
| Create Transfer | Transfer | With this permission, the user will be able to create a transfer document. |
| Delete Request | Transfer | With this permission, the user will be able to delete a transfer request. |
| Delete Transfer | Transfer | With this permission, the user will be able to delete a transfer document. |
| Edit Quantity | Transfer | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity. Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit Transfer | Transfer | With this permission, the user will be able to edit a transfer document. |
| Edit Transfer Request | Transfer | With this permission, the user will be able to edit a transfer request. |
| Edit Transfer Request CFA | Transfer | With this permission, the user will be able to capture CFAs in transfer request documents. |
| Reject Transfer Request | Transfer | With this permission, the user will be able to reject a transfer request. |
| Request Transfer | Transfer | With this permission, the user will be able to submit a transfer request. |
| Access Quick Receiving | Transfer Receiving | With this permission, a user will have access to Transfer Quick Receiving. |
| Access Transfer Receiving | Transfer Receiving | With this permission, a user will have access to Transfer Receiving. |
| Access Transfer Receiving Attribute | Transfer Receiving | With this permission, the Attributes tab on the Transfer Receiving Container - Item Detail screen is displayed and enabled. |
| Activate Container Edits | Transfer Receiving | With this permission, the user will be allowed to commit any changes made to the container. The user will be able to edit the details with the Edit permission however for the changes to be saved to the DB, the Activate Container Edits permission is needed. |

ORACLE®

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Add Unexpected Item to Transfer Receiving | Transfer Receiving | With this permission, the user will be allowed to receive items that are not present in the original delivery. |
| | | User must also have Modify Container and Edit Container permission. |
| | | Without this permission, the user will not be allowed to receive items that are not present in the original delivery. |
| Adjust Container | Transfer Receiving | With this permission, the user will be able to bring back a confirmed container to editable status. |
| Allow Default Zero at Confirmation | Transfer Receiving | With this permission, the user will be able to confirm a transfer receipt with the option to set all non received items to zero. |
| | | Without this permission, if there are any non received items on the container, the user will get a hard stop and not be able to confirm the delivery. |
| Confirm Container | Transfer Receiving | With this permission, the user will be able to confirm a container. |
| Confirm Receipt | Transfer Receiving | With this permission, the user will be able to confirm a transfer delivery. |
| Create Container | Transfer Receiving | With this permission, the user will be able to create a container. |
| Default Qty in All Containers | Transfer Receiving | With this permission, the user will be able to default the expected qty in received qty field in all the containers. |
| Default Quantity in Container | Transfer Receiving | With this permission, the user will be able to default the expected qty in received qty field for the items in the container. |
| | | User must also have Modify Container and Edit Container permission. |
| Delete Container | Transfer Receiving | With this permission, the user will be able to delete a container. |
| Display Expected Qty | Transfer Receiving | With this permission, the user will be able to view Expected Qty of an item in the containers. |
| Edit Container | Transfer Receiving | With this permission, the user will be allowed to edit the line item details, update qty, remove item, restore item, cancel the current edits, and so on. |
| | | User must also have Modify Container permission for the changes to be committed to the DB. |
| | | Without this permission, the user will not be allowed to edit any line item details, update qty, remove item, restore item, cancel the current edits, and so on. |
| Edit Container CFA | Transfer Receiving | With this permission, the user will be allowed to capture CFAs in the containers of transfer deliveries. |
| Edit Container Info | Transfer Receiving | With this permission, the user will be allowed to edit the container header details. |
| | | User must also have Modify Container permission for the changes to be committed to the DB. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Edit Delivery CFA | Transfer Receiving | With this permission, the user will be allowed to capture CFAs in the transfer deliveries. |
| Edit Quantity | Transfer Receiving | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity. |
| | | User must also have Modify Container and Edit Container permission. |
| | | Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit Receiving Info | Transfer Receiving | With this permission, the user will be able to edit the header details of a transfer delivery |
| Edit Transfer Receiving Attribute | Transfer Receiving | With this permission, extended attributes can be added/removed: Add Attributes and Remove Attributes (Trash can) buttons will be available on the Attributes tab of the Transfer Receiving Container - Item Detail screen. |
| Misdirected Container | Transfer Receiving | With this permission, the user will be allowed to copy the items from a misdirected container. |
| Receive On Shop Floor | Transfer Receiving | With this permission, the user will be able to receive the container on the shopfloor. |
| | | User must also have Modify Container and Edit Container Info permission. |
| | | Without this permission, user will not be able to receive the container on the shopfloor. |
| Record Receipt Damages | Transfer Receiving | With this permission, the user will be able to receive damaged items in a transfer delivery. |
| Access Container Attribute | Transfer Shipment | With this permission, the Attributes tab on the Transfer Shipment Container - Item Detail screen is displayed and enabled. |
| Access Shipment | Transfer Shipment | With this permission, a user will have access to the Transfer Shipment dialog for the user in the application. |
| Activate Container Edits | Transfer Shipment | With this permission, the user will be allowed to commit any changes made to the shipment. The user will be able to edit the details with the Edit Container permission however for the changes to be saved to the DB, the Activate Container Edits permission is needed. |
| | | Without this permission, the user will not be allowed to commit any changes made to the shipment. |
| Add Items with No Document | Transfer Shipment | With this permission, the user will be allowed to create an adhoc document through Shipments and add items to it. This controls the visibility of 'No document' button on the Select Document screen. |
| | | Without this permission, the user will not be allowed to create an adhoc document through shipments and add items to it. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Add Unexpected Item to Transfer Shipment | Transfer Shipment | With this permission, the user will be allowed to add items that are not present in the Transfer Document, into the shipment. |
| | | User must also have Modify Container and Edit Container permissions. |
| | | Without this permission, the user will not be allowed to add items that are not present in the Transfer document, into the shipment. |
| Adjust Carrier | Transfer Shipment | With this permission, the user will be able to update the BOL details of a shipment even after at least one container has been confirmed. |
| | | Without this permission, the user will not be able to update the BOL details of a shipment after at least one container has been confirmed. |
| Adjust Container | Transfer Shipment | With this permission, the user will be allowed to bring the container back to editable status. |
| Cancel Submit Shipment | Transfer Shipment | With this permission, the user can cancel submit Transfer shipments. |
| Confirm Container | Transfer Shipment | With this permission, the user can confirm containers in the shipments. |
| Create Shipment | Transfer Shipment | With this permission, the user can create shipments for Transfer documents. |
| | | Without this permission, the user will not be able to create shipments for Transfer documents. |
| Delete Container | Transfer Shipment | With this permission, the user can delete containers in the shipments. |
| Delete Shipment | Transfer Shipment | With this permission, the user can delete transfer shipments. |
| Dispatch Shipment | Transfer Shipment | With this permission, the user can dispatch shipments. |
| Edit Container | Transfer Shipment | With this permission, the user will be allowed to edit the line item details, update qty, remove item, restore item, cancel the current edits, and so on., |
| | | User must also have Modify Container permission. |
| | | Without this permission, the user will not be allowed to edit any line item details, update qty, remove item, restore item, cancel the current edits, and so on. |
| Edit Container Attribute | Transfer Shipment | With this permission, extended attributes can be added/removed: Add Attributes and Remove Attributes (Trash can) buttons will be available on the Attributes tab of the Transfer Shipment Container - Item Detail screen. |
| Edit Container CFA | Transfer Shipment | With this permission, the user will be able to capture CFAs in the containers of transfer shipments. |
| Edit Container Info | Transfer Shipment | With this permission, the user will be allowed to edit the container header details. |
| | | User must also have Modify Container permission. |

**Table 9-3    (Cont.) Security Permissions**

| Permission | Topic | Usage |
|---|---|---|
| Edit Quantity | Transfer Shipment | With this permission, the user will be able to tap on the item quantity and the quantity widget will open to edit the quantity. |
| | | User must also have Modify Container and Edit Container permission. |
| | | Without this permission, the quantity is not editable via the manual dialog and only scanning is allowed. |
| Edit Shipment BOL | Transfer Shipment | With this permission, the user will be allowed to edit the shipment BOL details. |
| Edit Shipment CFA | Transfer Shipment | With this permission, the user will be able to capture CFAs in transfer shipments. |
| Edit Shipment Info | Transfer Shipment | With this permission, the user will be allowed to edit the shipment header details. |
| Override Exclude Shipping Network | Transfer Shipment | With this permission, user will be able to override Shipping Network exclusion and will be able to dispatch to the stores that are not in the network. |
| Select Container Document | Transfer Shipment | With this permission, the user will be allowed to select transfer documents to be added to the shipment. |
| | | User must also have Modify Container and Edit Container permission. |
| Submit Shipment | Transfer Shipment | With this permission, the user can submit shipments. |
| Create UIN on the Fly | UIN | With this permission, the user is allowed to create a UIN on the fly when creating an inventory adjustment using a reason code of Disposition Movement from Out (Dist) to Available to Sell (ATS) = UIN Status in Stock. |
| | | User must also have Edit Inventory Adjustment permission as well as data permissions for each adjustment reason on the adjustment. |

# 10
# Operational Issues Screens

This chapter describes administration screens which users with System-Operator role, can view for operational issues.The operational issues are divided into four categories.

> **✎ Note:**
>
> **Date search** range has been defaulted to last 14 days. Users can change the date range.

- **Bulk Data Import**

  Scheduled background work tasks that handle importing mass data from external systems (through file or BDI). The issue counts the number of work tasks that failed.

  Primary Tables: BATCH_EXECUTION, BATCH_ACTIVITY

- **Data Purge**

  Scheduled background work tasks that archive and remove data from the database. The issue counts the number of work tasks that failed.

  Primary Tables: BATCH_EXECUTION, BATCH_ACTIVITY

- **Message Processing**

  Scheduled background work tasks that process asynchronous messages in a queue. The issue counts the number of messages that failed.

  Primary Tables: MPS_STAGED_MESSAGE

- **Transactional Execution**

  Scheduled background work tasks that execution business processes on transactional data. The issue counts the number of work tasks that failed.

  Primary Tables: BATCH_EXECUTION, BATCH_ACTIVITY

## Operational Issues List Screen

This screen displays the summary of operational issues for each category.

**Figure 10-1    Operational Issues List Screen**
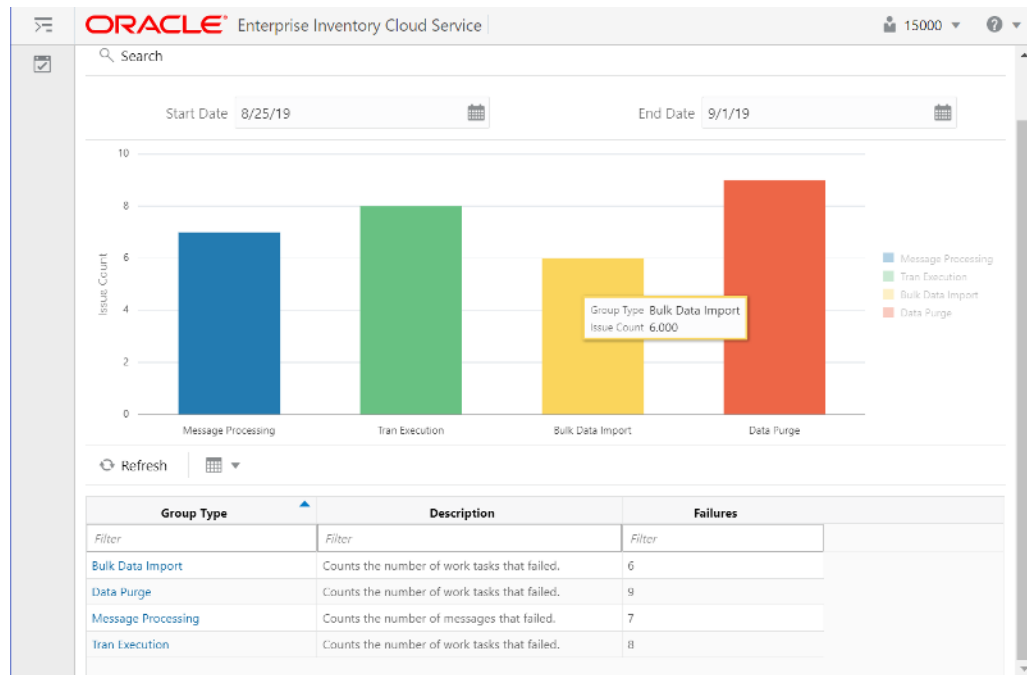


> **Note:**
>
> On the bar graph, the **Issue Count** will be displayed when user moves the cursor to the bar graph

# Operational Issues Review

This screen displays the operational issues for selected categories and date ranges.

It provides following common actions in the operational issue review screens.

## Operational Review Screen Actions

- **Search Filter**

  Each Operational Review Screen has the Search filters.

  The search filter fields would vary based on the operational groups.

  The search filter fields are: search limit, date range.

**Figure 10-2    Search Filter**
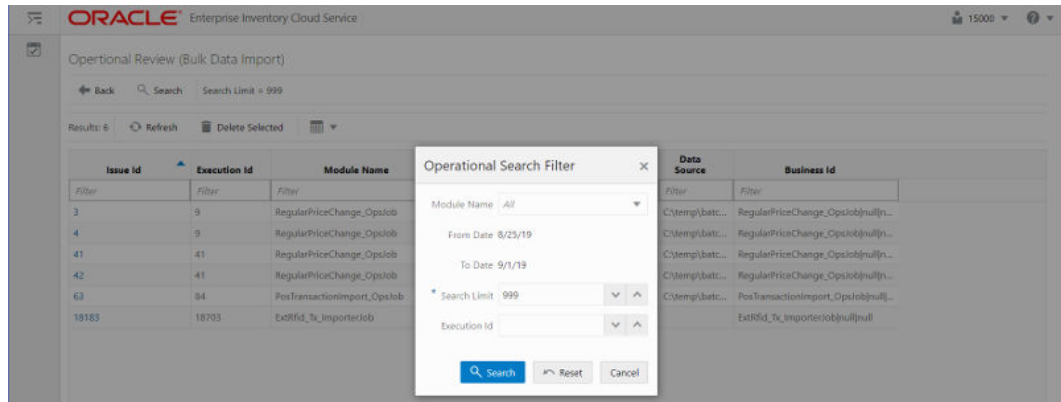


- **Issue Detail**

    Click the Issue Link, the issue detail dialog will be displayed. For example, the following screen shot displays the Bulk Data Import Issue Detail.

**Figure 10-3    Issue Detail**
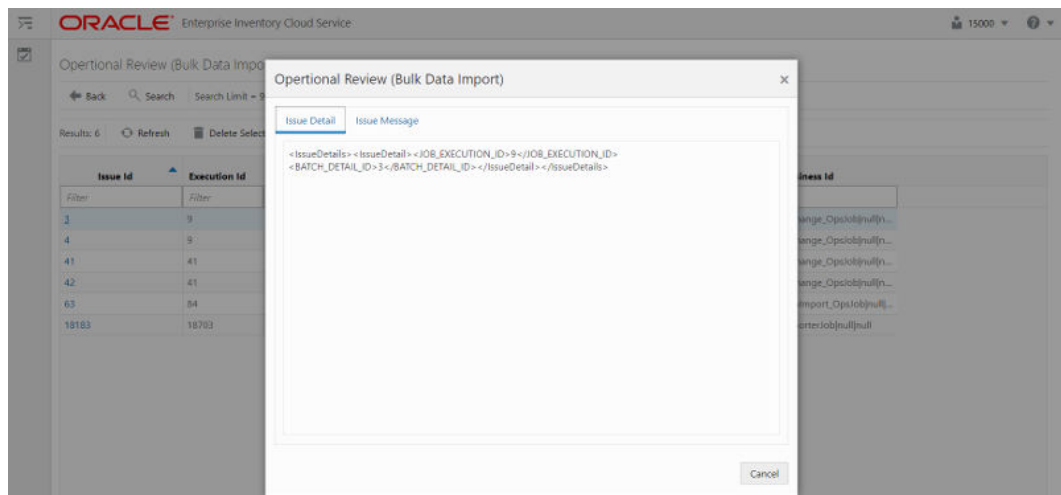


- **Delete Button**

    Mark the record as deleted, user needs to have **Batch Execution Delete** security permission to delete a record.

    The following security permissions are required to perform delete actions.

**Table 10-1    Security Permissions for Delete Actions**

| Operational Issue Category | Security Permissions |
| --- | --- |
| Bulk Data Import | Batch Execution Delete |
| Data Purge | Batch Execution Delete |
| Tran Execution | Batch Execution Delete |
| Message Processing | Delete MPS Staged Message |

## Bulk Data Import

This screen displays list of failed scheduled background work tasks that handle importing mass data from external system (through file or BDI).

**Figure 10-4    Bulk Data Import Screen**



## Data Purge

This screen displays a list of failed scheduled background work tasks that archive and remove data from the database.

**Figure 10-5    Data Purge Screen**



## Message Processing

This screen displays a list of failed background work tasks that process asynchronous messages in a queue.

**Figure 10-6    Message Processing Screen**



# Transactional Execution

This screen displays a list scheduled background work tasks that execute business processes on transactional data.

**Figure 10-7    Transactional Execution**

# A

# Appendix: Run Importer Separately from SIOCS Job Admin

## Run Data Seeding Importer Jobs

If you prefer import and load in separate processes, follow these steps:

1. **EICS Disable Data Seed**

   Login Oracle Enterprise Inventory Cloud Service page as EICS System Ops user.

   On Main Menu, Navigate to **Admin/Configuration/System Administration**.

   Search for data seed on option field, change value and save the change.

   **Enabled Seed Foundation**: set value to false

   **Enable Seed Store Data**: set value to false

2. **Start BDI Data Seeding Process Flow**

   Review process status.

3. **Start EICS Data Seeding Importer Job**

   Once the BDI process flow is completed, you can load data from EICS inbound stage tables into the application tables.

   To load stage data to application tables:

   Login EICS as system ops user.

   From Main Menu, navigate to **Admin/Technical Maintenance/Job Admin**.

**Figure A-1  Job Admin**



On Job Admin Screen, click **Data Seed**. This will bring you to the Data Seed screen.

**Figure A-2  Data Seed Screen**



Select **Job Name** (for example Store). From Seed Option drop-down list, select **Seed**. Next, click **Start Job**.

> **Note:**
>
> The store id input field is applicable to modules which contain store data.
>
> To load all stores data, leave the store id as blank. To load data for a specific store, please enter the store id into the store Id field.
>
> The Data Set Id is optional parameter for importer job to process the specified data set from bulk data interface table.

Review importer status in EICS Job Admin console.

**Figure A-3    Importer Status**



# BDI Transaction (TX) Importer Jobs

The following BDI Transaction Importer Jobs can be run as Ad hoc job if BDI process flowed failed on importer stage. The data set id is required. By default, the job schedules are setup in BDI job schedule admin console. The SIOCS/EICS job schedule for following importer jobs are disabled.

**Table A-1    BDI Transaction (TX) Importer Jobs**

| Importer Job | Notes |
| --- | --- |
| Clearance Transaction Import | Clearance pricing data from producer system to receiver system. |
| | Run after initial data seed, recurring or as needed. |
| Price Change Transaction Import | Regular price change data from producer system to receiver system. |
| | Run after initial data seed, recurring or as needed. |
| Warehouse Avail Inventory Import | Current available inventory for items at warehouses from producer system to receiver system. |
| | Run after initial data seed, recurring or as needed. |
| Third Party Price Import | Item prices coming from an external pricing application to receiver system. |
| | Run after initial data seed, recurring or as needed. |
| Third Party RFID Import | RFID data coming from an external RFID application to receiver system. |
| | Run after initial data seed, recurring or as needed. |

To Start an Adhoc Job, see Batches chapter Start an Ad hoc Job section for details.

# B

# Appendix: BDI Transmitter CLI Procedure - Load External Data File

**Prerequisites**

Following Server side installations must be installed by Oracle Retail Cloud Service Providers: EICS and BDI-SIM Edge App Job Admin.

Before you can load data files use BDI Transmitter CLI tool, you need install BDI Transmitter on Client machine. See following steps for installing BDI Transmitter CLI tools on Client side.

## BDI Transmitter CLI Tool Setup

To prepare the tool for use, follow these steps.

- The bdi-cli-transmitter home directory (where the tool software package is extracted) contains 'conf' directory where the tool related configuration files will be present, and 'bin' directory where the executable to run the tool will be present.
- Configure conf/bdi-file-transmitter.properties. The following describes the properties to be configured. The properties file provides some sample values for reference to start with. The values specified in the properties file can be overridden using the command-line input options if required, when running the tool for file transmission.

**Table B-1    bdi-file-transmitter.properties**

| Property | Description |
|---|---|
| source.system.name | The name of the source system or application that provides the source data to be transmitted.<br><br>For example, `source.system.name=EXTERNAL` |
| <receiverAppName>.receiver.url | The Receiver REST service URL of the BDI Receiver application indicated by <receiverAppName> (should be in lower-case). When use BDI Transmitter CLI tool to integrate with EICS, need to set the sim.receiver.url.<br><br>For example, if the BDI receiver application is SIM, then specify the property and value as:<br><br>sim.receiver.url=http://<bdi-sim-app-hostname>:<port>/sim-batch-job-admin/resources/receiver |
| <receiverAppName>.receiver.url.useralias | Alias name for the credentials to be used to connect to the corresponding receiver service. The alias name with the credentials are stored in a wallet. <receiverAppName> should be in lowercase.<br><br>Example:<br>`sim.receiver.url.useralias=simReceiverUrlUserAlias` |

**Table B-1  (Cont.) bdi-file-transmitter.properties**

| Property | Description |
|---|---|
| <InterfaceModuleName>.receiver.appname | Name of the BDI receiver application for the interface module <InterfaceModuleName>. |
| | Specify the name in lowercase. |
| | Example: |
| | `ExtPrice_Tx.receiver.appname=sim` |
| <InterfaceModuleName>.dataset.type | The data set type of the data to be transmitted for the interface module identified by <InterfaceModuleName>. |
| | Valid value is FULL or PARTIAL. |
| | Example: |
| | `ExtPrice_Tx.dataset.type=PARTIAL` |
| <InterfaceModuleName>.interfaceShortNames | The interface name(s) for the corresponding interface module <InterfaceModuleName>. Multiple interface names can be specified (each separated by a comma) as multiple interfaces can be part of an interface module. The interface module name and interface names should be the same as expected by the BDI receiver application where the files are transmitted. |
| | Example: |
| | `ExtPrice_Tx.interfaceShortNames=Ext_Price` |
| <InterfaceModuleName>.<InterfaceShortName>.input.filepath | Specify the file location where the corresponding interface data files to be transmitted are present. Each interface in a interface module should have separate file locations. |
| | Example: |
| | `ExtPrice_Tx.Ext_Price.input.filepath=/home/bdi/ext_price_tx/ext_price/files` |

- Run: bdi-file-transmitter.sh -setup-credentials.

> **Note:**
>
> bdi-file-transmitter.sh will be in the 'bin' directory.

Run -setup-credentials to configure the BDI Receiver service user credentials. Running this command will prompt for the username and password for each of the <receiverAppName>.receiver.url.useralias specified in bdi-file-transmitter.properties file.

The credentials entered for each alias will be stored in a secure wallet and used to connect to the corresponding BDI Receiver service for transmission of files.

This is a prerequisite step to use the tool but usually a one-time setup before running bdi-file-transmitter.sh for transmission of files.

- Optionally, configure conf/bdi-file-transmitter-runtime.properties that contains parameters (described below) for performance tuning of the tool.

Start with default values as present in the properties file, analyze the performance and choose optimal values for the parameters for better performance if required. The tool will use default values for the parameters (mentioned below) when no values are specified in the properties file.

| Property | Description |
|---|---|
| multiple_files_process_limit | The maximum number of files to process in parallel at any given time. Default value is 5. |
| file_transmission_thread_limit | The number of parallel threads to run to process a single file. Default value is 3. |
| transmission_record_size | The maximum number of records per block or chunk to transmit to the receiver service per service call. Default value is 20000. |
| transmission_timeout | The timeout in minutes for file transmission. The process will timeout and end when the file transmission is still not complete after the specified time. Default value is 300 minutes. |

**Example B-1    bdi-file-transmitter.properties**

```
source.system.name=ext-source

sim.receiver.url=https://<bdi-sim-app-hostname>:<port>/sim-batch-job-admin/
resources/receiver

sim.receiver.url.useralias=simReceiverUrlUserAlias

interfaceModules=ExtPrice_Tx

ExtPrice_Tx.receiver.appname=sim

ExtPrice_Tx.dataset.type=PARTIAL

ExtPrice_Tx.interfaceShortNames=Ext_Price

ExtPrice_Tx.Ext_Price.input.filepath=/home/bdi/ext_price_tx/ext_price/files
```

# Tool Usage

The BDI CLI Transmitter tool is run using the shell script: bdi-file-transmitter.sh from the 'bin' directory.

The tool can be run in interactive and non-interactive modes.

- Interactive Mode: Run bdi-file-transmitter.sh

  For user interactive mode where the program prompts for input, just run bdi-file-transmitter.sh with no options.

  This will prompt for each input with descriptions which will be self-explanatory. The user can enter value as required or skip optional parameters. When no value is specified for optional parameters, the tool will try to use the default values as specified in the bdi-file-transmitter.properties file or stop executing when no default value is present.

- Non-Interactive Mode: Run bdi-file-transmitter.sh [input]

  The tool can be run with the following inputs as described below.

  > **Note:**
  >
  > The only required input is interface module name, when the other input values are specified in bdi-file-transmitter.properties file.

| Input | Description |
|---|---|
| -m or --interfacemodule <interfaceModuleName > | (Required) The interface module name. Should be the same as the interface module name expected by the BDI receiver application. |

Some examples of running the transmitter tool command-line:

bdi-file-transmitter.sh -m ExtPrice_Tx

# File Processing

The BDI Transmitter tool supports transmission of flat files, for example, .csv files, in UTF-8 format. The BDI Receiver application supports only csv files. Hence the interface data files to be transmitted need to contain records with comma-separated field values.

The order of the fields in the file should be as expected by the BDI Receiver application, so that each value is inserted in the right columns of the destination interface tables. No header line should be present in the file (each line is treated as data record). Each record should be in a new line.

The interface module name and interface names for the files to be transmitted should be same as expected by the BDI Receiver application.

The transmitter tool can process a single file or a directory containing multiple files. But the tool does not process files recursively in subdirectories.

Files are processed and transmitted per interface module. Each run of processing of files of the interface module will be considered a transaction and a Transaction Id will be generated and associated to the transmission of files (at the interface module level). Files of multiple interfaces in an interface module will be part of the same transaction.

Each file transmission within a transaction will have a Transmission Id associated to it. The same transaction Id and transmission Id are sent to the BDI Receiver application, so the corresponding transmission details can be seen in the Job Admin console of the BDI Receiver application.

After successful transmission, the file will be moved to the archive directory:

<inputFileDirectory>/archive/<interfaceModuleName>/<transactionId>

For example, if the input file location is '/home/bdi/interface/files' and the interface module of the files is 'ExtPrice_Tx', and the transaction Id of the file transmission is 'Tx#1475858081837', then after successful transmission the file will be moved to the directory:

/home/bdi/interface/files/archive/ExtPrice_Tx/Tx#1475858081837.

# Output Logs

The transmitter tool outputs messages and logs to the terminal console where the command is run.

The tool also creates a log file that contains detailed logs about the processing of files. The log will show the Transaction Id and Transmission Id of each file transmission among other details.

The log file is created in the logs directory under the tool home directory (bdi-cli-transmitter/ logs).

The name of the log file will be in the format: bdi-file-transmitter_yyyy-mm-dd_hh:mm:ss, for example bdi-file-transmitter_2016-07-04_10:38:59.

## Error Reprocessing

In case of any error in file processing, error in transmission of file to the receiver service, timeout of file transmission, or any other failure, the file will be moved to the 'failed' directory: <inputFileDirectory>/failed/<interfaceModuleName>/<transactionId>

For example, if the input file location is '/home/bdi/interface/files' and the interface module of the files is 'ExtPrice_Tx', and the transaction Id of the file transmission is 'Tx#1475858081837', then if the transmission of file fails, the file will be moved to the directory: /home/bdi/interface/files/failed/ExtPrice_Tx/Tx#1475858081837.

A properties file containing the input details corresponding to the failed file will be created. For example, if the file named 'ExtPrice_1.csv' has failed, then a file named 'ExtPrice_Tx_1.csv.properties' will be created in the 'failed' directory. This acts as the input context that will be used when the file is reprocessed. The user should not delete or modify this properties file, if the data file has to be re-processed with the original input context.

Due to parallel processing of files by the transmitter, there may be a scenario where some records in the file may have been transmitted successfully, but part of the file transmission may have failed. Even in this case, the entire file will be treated as failed and moved to the 'failed' directory.

Reprocessing will be at the file level and not at the block level where the transmission may have failed. In the case of partial transmission of file, the BDI Receiver application also marks the whole transmission as failed and hence the entire file can be retransmitted to be processed again by the receiver application.

To retry failed files (that did not get transmitted successfully in previous transmission) use the below command:

bdi-file-transmitter.sh -retry-failed <inputFileDir or inputFilePath>

For example, bdi-file-transmitter.sh -retry-failed /home/bdi/interface/files/failed/ExtPrice_Tx/ Tx#1475858081837

bdi-file-transmitter.sh -retry-failed /home/bdi/interface/files/failed/ExtPrice_Tx/ Tx#1475858081837/ExtPrice_1.csv

Once a file is successfully reprocessed, it will be renamed as <filename>-retransmitted. For example, ExtPrice_1.csv-retransmitted. And the corresponding properties file will be deleted.

## Schedule EICS Importer Jobs for Transmitter Loaded Data

1. Login in EICS and navigate to the Job Scheduler screen.

2. Select the desired job (for example, External Price TX Import or External RFID TX Import) from the list.

3. Edit the Job details on right panel.

4. Set required interval for execution and **Enabled** to **Yes**.

5. Click **Apply** to exit the edit mode.

6. For configuring multiple jobs on a go, repeat Step 2 for each job.

7. Click **Save** to save the current changes made on the screen.

# View EICS Importer Jobs Execution Details

1. Launch EICS application, login EICS as System Ops user.

2. On Main Menu, select Admin/ Technical Maintenance/Job Admin.

3. To view detail record, select the record, then click hyperlink on the execution id. This will bring you to the Job Execution Detail screen.

# C

# Appendix: BDI Security Roles and Groups

BDI Batch Admin Security

BDI Process Flow Security

BDI Scheduler Security

## BDI Batch Admin Security

The following list shows the available BDI Edge Job Security groups in IDCS or OCI IAM. The customer can create an IDCS or OCI IAM user and assign to the proper group based on job duties:

- BdiEdgeSimJobAdminGroup
- BdiEdgeSimJobMonitorGroup
- BdiEdgeSimJobOperatorGroup

> ✎ **Note:**
>
> In stage environment, the preprod suffix is added to the groups.

**Table C-1    BDI Batch Admin Role/Groups**

| Role Name | Admin Role | Operator Role | Monitor Role |
|---|---|---|---|
| **Group Name** | **BdiJobAdminGroup** | **BdiJobOperatorGroup** | **BdiJobMonitorGroup** |
| Edit configuration from UI | Yes | No | No |
| Create/update/delete system options | Yes | No | No |
| Create/update/delete system credentials | Yes | No | No |
| View credentials | Yes | No | No |
| Run Jobs | Yes | Yes | No |
| Monitor Jobs | Yes | Yes | Yes |
| HTTP GET method access to rest services | Yes | Yes | Yes |
| HTTP POST/PUT/DELETE method access to rest services | Yes | Restricted to few services. | No |

# BDI Process Flow Security

The following list shows the available BDI Process Flow Enterprise Security groups in IDCS or OCI IAM. The customer can create an IDCS or OCI IAM user and assign to the proper group based on job duties:

- BdiProcessAdminGroup
- BdiProcessOperatorGroup
- BdiProcessMonitorGroup

> **Note:**
>
> In stage environment, the *_PREPROD suffix is added to the groups.

**Table C-2    BDI Process Flow Security Roles/Groups**

| Role Name | Admin Role | Operator Role | Monitor Role |
|---|---|---|---|
| Group Name | BdiProcessAdminGroup | BdiProcessOperatorGroup | BdiProcessMonitorGroup |
| Update Process DSL | Yes | No | No |
| Start/Restart Process | Yes | Yes | No |
| All other services | Yes | Yes | No |
| Read only Access to Process Flow Live, Manage Process Flow, Historical Process Flow Executions, System Logs UI tabs. | Yes | Yes | Yes |
| HTTP GET method access to rest services | Yes | Yes | Yes |
| HTTP POST/PUT/DELETE method access to rest services | Yes | Restricted to few services. | No |

# BDI Scheduler Security

The following list shows the available BDI Scheduler Enterprise Security groups in IDCS or OCI IAM. The customer can create an IDCS or OCI IAM user and assign to the proper group based on job duties:

- BdiSchedulerAdminGroup
- BdiSchedulerOperatorGroup
- BdiSchedulerMonitorGroup

> **✎ Note:**
>
> In stage environment, the *_PREPROD suffix is added to the groups.

**Table C-3    BDI Process Flow Security Roles/Groups**

| Role Name | Admin Role | Operator Role | Monitor Role |
|---|---|---|---|
| **Group Name** | **BdiSchedulerAdminGroup** | **BdiSchedulerOperatorGroup** | **BdiSchedulerMonitorGroup** |
| View and search | Yes | Yes | Yes |
| Create schedule | Yes | No | No |
| Edit schedule | Yes | No | No |
| Delete schedule | Yes | No | Yes |
| Manual run schedule | Yes | Yes | Yes |
| Disable schedule | Yes | Yes | No |
| Enable schedule | Yes | Yes | No |
| HTTP GET method access to rest services | Yes | Yes | Yes |
| HTTP POST/PUT/ DELETE method access to rest services | Yes | Yes | No |

# D
# Appendix: Report Formats

## Reports

**Table D-1    Reports**

| Report Name | Report Parameters |
| --- | --- |
| Figure D-1 | bin_id, sim_customer_order_id, copies |
| Figure D-2 | delivery_id, store_timezone, copies |
| Figure D-3 | delivery_id, store_timezone, copies |
| Figure D-4 | store_timezone, pick_id, copies |
| Figure D-5 | store_timezone, pick_id, copies |
| Figure D-6 | order_id, store_timezone, copies |
| Figure D-7 | reverse_pick_id, store_timezone, copies |
| Figure D-8 | receipt_id, store_timezone, copies |
| Figure D-9 | receipt_id, store_timezone, copies |
| Figure D-10 | Inv_Adjust_ID, copies |
| Figure D-11 | inv_adj_id, store_timezone, copies |
| Figure D-12 | Item_basket_id, store_timezone, copies |
| Figure D-13 | itemid, storeid, store_timezone, copies |
| Figure D-14 | purchase_order_id, store_timezone, copies |
| Figure D-15 | replenish_gap_id, copies, store_timezone, locale_id |
| Figure D-16 | item_id, from_date, to_date, locale_id, copies |
| Figure D-17 | return_id, store_timezone, copies |
| Figure D-18 | shelf_adjust_id, copies, store_timezone |
| Figure D-19 | shelf_replenish_id, store_timezone, copies |
| Figure D-20 | store_id, stock_count_id, copies |
| Figure D-21 | stock_count_id, copies |
| Figure D-22 | store_id, copies |
| Figure D-23 | stock_count_id, location_id, store_timezone, phase, copies |
| Figure D-24 | store_order_id, store_timezone, locale_id |
| Figure D-25 | carton_ID, copies |
| Figure D-26 | delivery_id |
| Figure D-27 | carton_id |
| Figure D-28 | delivery_id |
| Figure D-29 | transfer_id, store_timezone, copies |

**Table D-1    (Cont.) Reports**

| Report Name | Report Parameters |
|---|---|
| Figure D-30 | shipment_id, store_timezone, copies |
| Figure D-31 | store_timezone, carton_id, copies |
| Figure D-32 | shipment_id, copies, store_timezone |
| Figure D-33 | carton_id |
| Figure D-34 | carton_ID, copies |
| Figure D-35 | carton_id, locale_id |
| Figure D-36 | store_timezone, ship_number, copies |
| Figure D-37 | store_timezone, carton_id, copies |
| Figure D-38 | store_timezone, ship_number, copies |
| Figure D-39 | carton_id, copies |

# Report Formats

The following section describes the report formats.

**Figure D-1    Customer Order Bin Label Report**
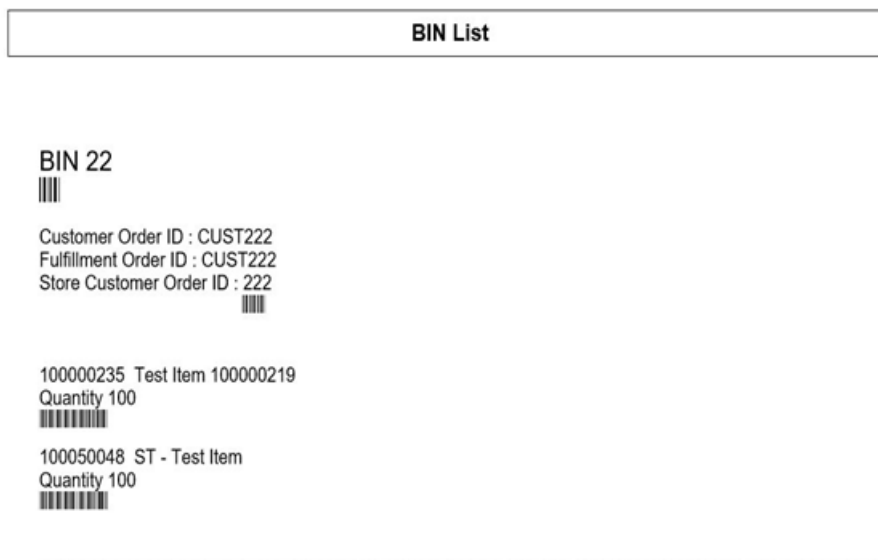
**Figure D-2    Customer Order Delivery BOL Report**



**Figure D-3    Customer Order Delivery Report**

**Figure D-4    Customer Order Pick Discrepancy Report**

**Customer Order Pick Discrepancy Report**

| | | | |
|---|---|---|---|
| **Store:** | 1511 - Phoenix | **Pick Create Date:** | 12/19/2017 |
| **Pick ID:** | 1086 | **Pick Create User:** | QAADMIN |
| **Pick Status:** | In Progress | | |

| Item | Description | Store Customer Order ID | Bin ID | Fulfillment ID | UOM | Pack Size | Old Pick Qty | Adjusted Pick Qty |
|---|---|---|---|---|---|---|---|---|
| 100005016 | Signal booster | 301 | 142 | LG1 | EA | 1 | 2 | 0 |

**Figure D-5    Customer Order Pick Report**

| Customer Order Pick Report |
|---|

| | | | | |
|---|---|---|---|---|
| **Store:** | 1111 - Charlotte * | **Pick Create Date:** | 12/19/2017 |
| **Pick ID:** | 1087 | **Pick Create User:** | qaadmin |
| **Pick Status:** | New | **Pick Complete Date:** | |
| | | **Pick Complete User:** | |

| Item | Description | Store Customer Order ID | Bin ID | Fulfillment ID | UOM | Pack Size | Suggested Pick Qty | Actual Pick Qty | Substitute |
|---|---|---|---|---|---|---|---|---|---|
| 100005016 | Signal booster | 1486 | | PERF_CUS_E XT1486 | EA | 1 | 10 | | |

| **Printed:** 1/4/2018 | **Page Number:** 1 |
|---|---|

**Figure D-6   Customer Order Report**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Customer Order Report** | | | | | | | | |

| | | | | | | |
|---|---|---|---|---|---|---|
| **Store:** | 1111 | | | | | |
| **Store Customer Order Id:** | 404 | **Create Date:** | 2017-07-19T13:29: 22.000-07:00 | **Delivery Type:** | Ship To Customer |
| **Customer Order Id:** | test1 | **Release Date:** | 2017-08-13T13:13: | **Carrier:** | Other |
| **Fulfillment Order Id:** | test1 | **Delivery Date:** | 12.000-07:00 2017-08-13T13:13: 12.000-07:00 | **Service:** | Other |
| **Status:** | Completed | | | **Allow Partial Delivery:** | Yes |
| **Reservation Type:** | Web Order | | | | |
| **Comments:** | Testing the External Comments. Do they work? | | | | |

| Item | Description | UOM | Order Qty | Picked Qty | Delivered Qty | Canceled Qty | Last Update Date | Comments | Substitute |
|---|---|---|---|---|---|---|---|---|---|
| 100005016 | Signal booster | EA | 20 | 20 | 20 | 0 | 2017-08-11T10:14: 00.000-07:00 | I'm testing the item comments. | |

**Figure D-7    Customer Order Reverse Pick Report**

| | |
|---|---|
| **Customer Order Reverse Pick Report** | |

| | | | | | | |
|---|---|---|---|---|---|---|
| **Store:** | 1511 | | | | | |
| **Reverse Pick ID:** | 501 | **Order Status:** | In Progress | | | |
| **Store Customer Order ID:** | 63 | **Reverse Pick Status:** | Canceled | | | |
| **Customer Order ID:** | CC2 | **Reservation Type:** | Web Order | **Comments:** | Testing the External Comments. Do they work? |
| **Fulfillment Order ID:** | CC2 | **Reverse Pick Create Date:** | 2017-06-29T11:16:48.000-07:00 | | | |

| Item | Description | UOM | Order Qty | Picked Qty | Delivered Qty | Canceled Qty | Sugg. Reverse Qty | Qty | Substitute |
|---|---|---|---|---|---|---|---|---|---|
| 100005013 | Induction Test Item | EA | 1 | 1 | 0 | 0 | 0 | 0 | |

**Figure D-8    Direct Delivery Discrepant Items Report**

| Direct Delivery Discrepant Items Report |
|---|

**Supplier:** 6100 - Local Grocery Supplier #2
**Store:** 1141 - Nashville
**Delivery/ASN:** DQ3
**PO Number:** 23456 ;

**Container ID:** 220
**Status:** Received

| Item | Description | UOM | Pack Size | Expected | Quantity | Disposition |
|---|---|---|---|---|---|---|
| 100350059 | ncg item | LB | 1 | 7 | 2 | Damaged |
| 100350059 | ncg item | LB | 1 | 7 | 2 | Damaged |
| | | | Totals: | 14.00 | 4.00 | |

**Driver Signature:**

**Employee Signature:**

**Figure D-9    Direct Delivery Report**



**Figure D-10    Inventory Adjustment AGSN Report**

**Figure D-11    Inventory Adjustment Report**

**Inventory Adjustment Report**

| | |
|---|---|
| Store: | 1141 |
| Adjustment Id: | 1081 |
| Create Date: | 08/03/2017 |
| Create User: | QA_005 |
| Approval Date: | |
| Approval User: | |
| Status: | In Progress |
| Comment: | |

| Item | Description | UOM | Pack Size | Qty | Reason | Sub-Bucket |
|---|---|---|---|---|---|---|
| 100000008 | Test Item 100000008 | Cases | 10 | 10 | Stock Count In | |

**Figure D-12    Item Basket Report**

**Figure D-13    Item Detail Report**



**Item Report**

| | | | | | |
|---|---|---|---|---|---|
| **Item** | SIM_800 | **Item Description** | SIM_800 | **Ranged** | Yes |
| **Primary UPC** | | **Primary Supplier Name** | Fine Jewelry Supplier | **Merchandise Hierarchy** | |
| **VPN** | | **Primary Supplier Number** | 1300 | **Dept** | dept5600 |
| | | | | **Class** | class5601 |
| **Item Status** | Active | **Ticket Type** | | **Subclass** | subclass5602 |
| | | | | **Differentiators:** | |

| | | | | | |
|---|---|---|---|---|---|
| **Stock on Hand Units:** | | **Ordering Attributes:** | | **Pricing:** | |
| **Total Stock on Hand** | 0 | **Repl Method** | | **Current Retail** | USD100 |
| **Pack Size** | | **Reject Store Order** | No | **Pricing Status** | Permanent |
| **Available SOH** | 0 | **Next Delivery Date** | null | **Promotional Type** | |
| **Shop Floor** | | | | | |
| **Back Room** | 0 | | | | |
| **Unavailable** | 0 | | | | |
| **Transfer Reserved** | 0 | | | | |
| **RTV Reserved** | 0 | | | | |
| **Ordered Quantity** | 0 | | | | |
| **Delivery Bay** | 0 | | | | |
| **In Transit** | 0 | | | | |
| **Received Today** | 0 | | | | |

**Allocations:**
**Sequencing:**

Printed: 1/4/2018                                                                    Page Number: 1

**Figure D-14    Purchase Order Report**

### Purchase Order Report

| | |
|---|---|
| **PO Id:** | 5681 |
| **Not Before Date:** | 03/12/2021 |
| **Not After Date:** | 03/15/2021 |
| **Supplier:** | 6200 - Local Grocery Supplier #3 |
| **PO Number:** | PORe1 |
| **To Location:** | 1141 - Nashville |
| **Status:** | Completed |
| **Complete Date:** | 03/12/2021 |

| Receipt Id | Delivery/ASN Id | Receipt Date | Receipt Status |
|---|---|---|---|
| 5703 | ASNre69 | 03/12/2021 | Received |
| 5704 | ASNre70 | 03/12/2021 | Received |
| 5705 | ASNre71 | 03/12/2021 | Received |
| 5706 | ASNre72 | 03/12/2021 | Received |
| 5702 | ASNre68 | 03/12/2021 | Received |

| Item | Description | UOM | Pack Size | Expected | Received | Unit Cost |
|---|---|---|---|---|---|---|
| 100000024 | Mouse | Cases | 1 | 2 | 2 | 10 USD |
| 100000059 | Washing Machine | Cases | 1 | 2 | 2 | 10 USD |
| 100000083 | Test Item 100000067 | Cases | 1 | 2 | 2 | 10 USD |
| 100000112 | Test Item 100000091 | Cases | 1 | 2 | 2 | 10 USD |
| **Totals:** | | | | 8.00 | 8.00 | |

**Figure D-15    Replenishment Gap Report**

| | |
|---|---|
| **Scan List Report** | |

| | |
|---|---|
| **Store:** | 1511 - Phoenix |
| **ID:** | |
| **Type:** | Gap |
| **Create Date/Time:** | 05/10/2017 |
| **Update Date/Time:** | 05/16/2017 |
| **User:** | |
| **Status:** | Completed |

| SKU | Description | UOM | Pack Size | Qty |
|---|---|---|---|---|
| 100000032 | Test Item 100000032 | Cases | 10 | 1 |
| 100005016 | Signal booster | Cases | 1 | 1 |
| 100005017 | Induction Test Item | Cases | 1 | 1 |
| SIM_125 | SIM_125 | Cases | 1 | 1 |
| SIM_400 | SIM_400 | Cases | 1 | 3 |

| | |
|---|---|
| Printed: 1/2/2018 | Page Number: 1 |

**Figure D-16    RFID History Report**

| RFID History Report |
| --- |

Item    100050056 - ST - Test Item

Date:  08/14/2019
EPC:  EPC95278

| Zone | Location | Transaction Type | Transaction ID | Observed |
| --- | --- | --- | --- | --- |
|  | Store   1311 | POS Sale | 1462 | No |

Date:  08/14/2019
EPC:  854126

| Zone | Location | Transaction Type | Transaction ID | Observed |
| --- | --- | --- | --- | --- |
| 61 | Store   1311 | RFID | 761 | Yes |

Date:  08/14/2019
EPC:  980403

| Zone | Location | Transaction Type | Transaction ID | Observed |
| --- | --- | --- | --- | --- |
| 404 | 2 | RFID | 762 | Yes |

| Printed: 11/19/2019 | Page Number: 1 |
| --- | --- |

**Figure D-17 RTV Report**

**RTV Report**

| | |
|---|---|
| RTV Number: | 1584 |
| External ID: | 1 |
| Authorization Number: | RTVREG1 |
| Status: | Canceled Request |
| User: | EXTERNAL |
| Not After Date: | 11/25/2017 |
| Approved Date: | 11/23/2017 |
| Supplier: | 1200 |
| Total SKUs: | 2 |
| Return Type: | |

| Item | Description | UOM | Pack Size | Reason Code | Req Qty | App Qty | Rem Qty | In-Ship Qty | Shipped Qty |
|---|---|---|---|---|---|---|---|---|---|
| SIM_125 | SIM_125 | Cases | 1 | Externally Initiated | 2 | 2 | 2 | 0 | |
| SIM_126 | SIM_126 Short Desc | EA | 1 | Overstock | 2 | | | | |

Printed: 1/2/2018     Page Number: 1

**Figure D-18 Shelf Adjustment Report**

**Shelf Adjustment List Report**

| | |
|---|---|
| Store: | 1511 - Phoenix |
| ID: | 623 |
| Type: | Move To Backroom |
| Create Date/Time: | 2017-12-14T09:03:37.000-08:00 |
| Update Date/Time: | 2017-12-14T09:05:00.000-08:00 |
| User: | qa_004 |
| Status: | In Progress |

| SKU | Description | UOM | Pack Size | Qty |
|---|---|---|---|---|
| SIM_125 | SIM_125 | Cases | 1 | 1 |
| SIM_126 | SIM_126 Short Desc | Cases | 1 | 1 |

Printed: 1/2/2018     Page Number: 1

**Figure D-19    Shelf Replenishment Report**

**Shelf Replenishment Report**

| | |
|---|---|
| **Store:** | 1511 - Phoenix |
| **ID:** | 1001 |
| **Shelf Replenishment Type:** | Capacity |
| **Replenishment mode:** | End of Day |
| **Product Group:** | SIM 125 126 140 |
| **Hierarchy: Scan List:** | |
| **Create Date/Time:** | 09/11/2017 |
| **User:** | qa_004 |
| **Status:** | Canceled |
| **Quantity:** | 181 |

| Item Id | Item Description | Pick From Area | Type | Selling Unit Of Measure | Pack Size | Qty | Actual Qty |
|---|---|---|---|---|---|---|---|
| SIM_125 | Mobile Signal booster | Backroom | Capacity | | 1 | 100 | |
| SIM_126 | Wifi booster | Backroom | Capacity | | 1 | 81 | |
| SIM_143 | SIM_143 | Backroom | Capacity | | 1 | 81 | 1 |
| SIM_144 | SIM_144 | Backroom | Capacity | | 1 | 81 | 0 |

| | |
|---|---|
| Printed: 1/2/2018 | Page Number: 1 |

**Figure D-20    Stock Count All Location Report**

### All Location Stock Count Report

**Description:**    125126 140
**Date:**    10/22/2017
**Total Items:**    1
**Stock Count User:**
**Re-Count User:**
**Authorization User:**

| Item | Item Description | Location | UOM | Count |
|------|------------------|----------|-----|-------|
| SIM_140 | SIM_140 | No Location | EA | |

**Description:**    125126 140
**Date:**    10/22/2017
**Total Items:**    2
**Stock Count User:**
**Re-Count User:**
**Authorization User:**

| Item | Item Description | Location | UOM | Count |
|------|------------------|----------|-----|-------|
| SIM_125 | SIM_125 | Back Room1 | LB | |
| SIM_126 | SIM_126 Short Desc | Back Room1 | EA | |

**Description:**    125126 140
**Date:**    10/22/2017
**Total Items:**    2
**Stock Count User:**
**Re-Count User:**
**Authorization User:**

| Item | Item Description | Location | UOM | Count |
|------|------------------|----------|-----|-------|
| SIM_125 | SIM_125 | Shop Floor1 | LB | |
| SIM_126 | SIM_126 Short Desc | Shop Floor1 | EA | |

Private and Confidential

**Figure D-21    Stock Count Export Report [XML Format]**

```
<STOCK_COUNT_EXPORT>

<STOCK_COUNT>
            <COUNT_ID>662</COUNT_ID>
            <STORE_ID>1511</STORE_ID>
            <DESCRIPTION>125126 140</DESCRIPTION>
            <LIST_STOCK_COUNT_LINE_ITEM>

            <STOCK_COUNT_LINE_ITEM>
                        <ITEM_ID>SIM_126</ITEM_ID>
                        <ITEM_DESC>SIM_126 Short Desc</ITEM_DESC>
                        <ITEM_SNAPSHOT></ITEM_SNAPSHOT>
                        <LIST_UINS>
                                <UINS>

                        <UIN/>
                                </UINS>
                        </LIST_UINS>
            </STOCK_COUNT_LINE_ITEM>
            <STOCK_COUNT_LINE_ITEM>
                        <ITEM_ID>SIM_125</ITEM_ID>
                        <ITEM_DESC>SIM_125</ITEM_DESC>
                        <ITEM_SNAPSHOT></ITEM_SNAPSHOT>
                        <LIST_UINS>
                                <UINS>

                        <UIN/>
                                </UINS>
                        </LIST_UINS>
            </STOCK_COUNT_LINE_ITEM>
            <STOCK_COUNT_LINE_ITEM>
                        <ITEM_ID>SIM_140</ITEM_ID>
                        <ITEM_DESC>SIM_140</ITEM_DESC>
                        <ITEM_SNAPSHOT></ITEM_SNAPSHOT>
                        <LIST_UINS>
                                <UINS>

                        <UIN/>
                                </UINS>
                        </LIST_UINS>
            </STOCK_COUNT_LINE_ITEM>

            </LIST_STOCK_COUNT_LINE_ITEM>
        </STOCK_COUNT>
</STOCK_COUNT_EXPORT>
```

**Figure D-22    Stock Count Rejected Item Report**

Rejected Items Report

**Stock Count Description:**    Nithin Stk Cnt
**Stock Count Group:**    41
**Schedule Date:**    1/8/14
**Total Rejected Items:**    1

| SIM Item Id | Item Description | Rejected Item ID | Rejected UIN | Count Quantity | Count Location | Status | Comments |
|---|---|---|---|---|---|---|---|
| | | 100177107 | | 1 | | Item Rejected | |

**Stock Count Description:**    TEST Schedule
**Stock Count Group:**    141
**Schedule Date:**    11/1/13
**Total Rejected Items:**    5

| SIM Item Id | Item Description | Rejected Item ID | Rejected UIN | Count Quantity | Count Location | Status | Comments |
|---|---|---|---|---|---|---|---|
| | | 100000657 | | 2 | | Item Not On Count | |
| | | 100006021 | | 2 | | Item Not On Count | |
| | | 1000006021 | | 1 | | Item Rejected | |

Private and Confidential

**Figure D-23    Stock Count Report**

**Stock Count Report**

| | |
|---|---|
| **Stock Count ID:** | 221 |
| **Stock Count Child ID:** | 2768 |
| **Description:** | Ish_Unit_ORSIM-22925_s6_sch1 : Shelfish#1 |
| **Date:** | 9/17/2021 |
| **Total Items:** | 1 |
| **Stock Count User:** | sim_qa7 |
| **Re-Count User:** | |
| **Notes:** | |

| Item | Description | UOM | Counted |
|---|---|---|---|
| 890790890 | Ish_RegItem_1 | EA | 10 |

**Figure D-24    Store Order Report**

**Store Order Report**

| | | | | | |
|---|---|---|---|---|---|
| **Store ID:** | 1411 - Seattle* | **Requested Date:** | 11/22/2020 | **Restrictions** | |
| **ID:** | 1 | **Create Date:** | 06/23/2020 | **Supplier:** | - |
| **Reference ID:** | 4 | **Approved Date:** | | **Warehouse:** | - |
| **External ID:** | EXT1 | **Auto Approval Date:** | | **Department:** | - |
| | | | | **Class:** | - |
| **Description:** | Report Testing | **Created User:** | sim_qa1 | **Sub-Class:** | - |
| **Status:** | In Progress | **Approved User:** | | **Area:** | - |
| **Context:** | Customer Orders | | | **Store Order Items:** | Yes |
| **Origin:** | External | **Total Quantity:** | 168 | | |

**Custom Flexible Attributes**
CFA Alpha : 2
CFA Sue 10 : 2020-10-02T00:00:00.000-04:00
Attr1 : 23
Attr2 : 2020-09-10T12:41:32.000-04:00

| Item | Description | UOM | External Quantity | Quantity |
|---|---|---|---|---|
| 100000294 | Test Item 100000278 | Units | 300 | 9 |
| 100000147 | Toned milk | Units | 3 | 150 |
| 100000083 | Test Item 100000067 | Units | 3 | 9 |

**Figure D-25    Transfer Delivery AGSN Report**

**Figure D-26   Transfer Delivery Exception Report**

**Transfer Receiving Exception Report**

**Source:**              1311 - Chicago*
**Destination:**         1321 - Indianapolis
**Source Type:**         Store
**Delivery/ASN:**        546
**Status:**              New
**Expected Date:**       06/28/2017

**Container ID :** 000132132011
**Status:** New

| Item | Description | UOM | Pack Size | Expected | Received | Damaged | Difference |
|------|-------------|-----|-----------|----------|----------|---------|------------|
| SIM_3 | SIM_3 | Cases | 1 | 1 | 0 | 0 | 1 |

**Printed:** 1/3/2018                                                    **Page Number:** 1

**Figure D-27    Transfer Delivery Label**

From
## Chicago*
123 Street
Anytown
Anycity MN  50250
US

To
## Indianapolis
123 Street
Anytown
Anycity MN 50250
US

(420) 50250

Label Type

# TRNSFR

**Label Reason** : Reprint
**Reference Container Id** :
**Number Of Items** : 1

Dept #S

5555

Store
(01) 1321

Store

# 1321

SSCC -18

000132132011

**Figure D-28    Transfer Delivery Report**

**Transfer Receiving Report**

| | |
|---|---|
| **Transfer Receipt ID:** | 3766 |
| **Source:** | 1151 - Dallas |
| **Destination:** | 1141 - Nashville |
| **Source Type:** | Store |
| **Delivery/ASN:** | 7744 |
| **Status:** | Received |
| **Expected Date:** | 11/09/2020 |
| **Received Date:** | 03/12/2021 |
| **Notes:** | |

**Container ID : 000000011410334061**
**Status: Received**

| Item | Description | UOM | Pack Size | Expected | Received | Damaged | Out of Stock |
|---|---|---|---|---|---|---|---|
| 100000024 | Mouse | EA | 10 | 1 | 0 | 0 | |
| 456789123456 01 | Sue GS1 Item | Cases | 1 | 1 | 2 | 0 | Yes |

**Figure D-29    Transfer Report**

**Transfer Report**

| | | | | |
|---|---|---|---|---|
| **Transfer ID:** | 8827 | | | |
| **External ID:** | 76 | | | |
| **No of Items:** | 4 | | | |
| **Not After Date:** | 26-DEC-21 | **Approved Date:** | 15-MAR-21 |
| **Unavailable:** | Yes | **Partial Delivery:** | Yes |
| **Customer Order Id:** | AO20 | **Fulfillment Order Id:** | AO20 |
| **Context Type:** | Promotion | **Context Value:** | 124 |

| | | | |
|---|---|---|---|
| **Source Type:** | Store | **Destination Type:** | Store |
| **Source:** | 1151 - Dallas | **Destination:** | 1141 - Nashville |

| Transfer Receipt ID | ASN | Delivery Status | Expected Date | Received Date | Shipment ID | Ship Date | Shipment Status |
|---|---|---|---|---|---|---|---|
| 4006 | 8265 | Received | 03/15/2021 | 03/15/2021 | 8265 | 03/15/2021 | Shipped |
| 4004 | 8263 | Received | 03/15/2021 | 03/18/2021 | 8263 | 03/15/2021 | Shipped |
| 4005 | 8264 | New | 03/15/2021 | | 8264 | 03/15/2021 | Shipped |

| Item | Description | UOM | Requested | Approved | In-Shipping | Shipped | Received | Damaged |
|---|---|---|---|---|---|---|---|---|
| 100000083 | Test Item 100000067 | Cases | 1 | 1 | 0 | 1 | 1 | 0 |
| 100000059 | Washing Machine | Cases | 1 | 1 | 0 | 1 | | |
| 100000024 | Mouse | Cases | 1 | 1 | 0 | 1 | 1 | 0 |
| 100000112 | Test Item 100000091 | Cases | 1 | 1 | 0 | 1 | | |

**Figure D-30    Transfer Shipment BOL Report**

---

### Transfer Shipment BOL Report

| ASN: 561 | Barcode: ||||| |
|----------|---------|

| BOL ID: 723 | Shipment ID: 561 | Motive: Bill of Lading Transfer New |
|-------------|------------------|------|
| Create Date: 2017-06-29 | Create User: qa_007 | |

| **Sender**<br>3111 - Montreal*<br>123 Street<br>Anytown<br>Anycity MN<br>50250<br>US | **Receiver**<br>3112 - Quebec<br>123 Street<br>Anytown<br>Anycity MN<br>50250<br>US |
|---|---|
| **Ship From**<br>123 Street<br>Anytown<br>Anycity , MN<br>50250<br>US | **Ship To**<br>Quebec<br>123 Street<br>Anytown<br>Anycity, MN<br>50250<br>US<br>3122222473 |

**Carrier**

Requested Pick-Up Date:

◇ Sender    ◇ Receiver    ◇ Third Party

Carrier Name:          Carrier Signature:
  Parcel Test

          Dispatch Date:

Carrier Address:

| Service: Parcel Test | | Tax ID : | |
|---|---|---|---|
| **Container ID** | **Weight (LBS)** | **Package Type** | **Tracking ID** |
| 11 | 11.00 | | |

**Notes**

---

Ship Container No: 11                    Barcode: ||||

| Item ID | EAN | Description | UOM | Quantity |
|---------|-----|-------------|-----|----------|
| 100300166 | | 100300166 | Cases | 1.00 |

| Legalese fine print |
|---|
| |

| Driver signature | Date | Receiver signature | Date |
|---|---|---|---|

**Figure D-31    Transfer Shipment Carton Report**

**Transfer Shipment Container Report**

| | |
|---|---|
| Source: | 1511 - Phoenix |
| Destination: | 1411 - Seattle* |
| Destination Type: | Store |
| Ship Date: | |
| Shipment ID: | 909 |
| Authorization Number: | |
| Status: | In Progress |
| User: | |
| Container: | 950 |
| Container Status: | In Progress |

Document: 1506

| Item | Description | UOM | Pack Size | Ship Qty | Reason Code |
|---|---|---|---|---|---|
| SIM_125 | SIM_125 | Cases | 1 | 1 | |
| SIM_1 | SIM_1 | Cases | 1 | 1 | |

Printed:  1/3/2018                                                    Page Number: 1

**Figure D-32    Transfer Shipment Report**

**Transfer Shipment Report**

| | |
|---|---|
| Source: | 1151 - Dallas |
| Destination: | 1141 - Nashville |
| Destination Type: | Store |
| Ship Date: | 03/16/2021 |
| Shipment ID: | 8283 |
| Authorization Number: | |
| Status: | Shipped |
| User: | sim_qa5 |

Container ID :    7643
Container Status: Shipped

| Item | Description | UOM | Document | Pack Size | Ship Quantity | Reason Code |
|---|---|---|---|---|---|---|
| 100000024 | Mouse | EA | 8843 | 10 | 1 | |
| 100000059 | Washing Machine | EA | 8843 | 10 | 1 | |

ORACLE®

**Figure D-33    Transfer Shipping Label**

| From | To |
|------|-----|
| **Phoenix**<br>123 Street<br>Anytown<br>Anycity MN  50250<br>US | **Seattle\*** |

**(420) Ship To Postal Code**

(420)50250

**Label Type**

# Bill of Lading Transfer

Customer Order: No
Number Of Items: 2
Context Value:

**Dept #S**

5555

**Store**

(01)1411

**Store**

1411

**SSCC -18**
000141155106

**Figure D-34    Vendor Delivery AGSN Report**

**Figure D-35    Vendor Delivery Label**

| From | To |
|---|---|
| **Local Grocery Supplier #2**<br>123 Main St<br>Portland OR  83273<br>US | **Nashville**<br>123 Street<br>123 Street<br>Anycity MN 50250<br>US |

| (420) 50250 | **Label Type** |
|---|---|
| [barcode] | **DSD** |

| **Label Reason:** Reprint<br>**Reference Container Id:** REF123<br>**Number Of Items:** 3 | **Dept #S**<br><br>1117 |
|---|---|

| **Store**<br>**(01) 1141**<br>[barcode] | **Store**<br>**1141** |
|---|---|

**SSCC -18**

[barcode]

**DQ10**

**Figure D-36    Vendor Shipment BOL Report**



| RTV Shipment BOL Report | | | |
|---|---|---|---|
| **RTV:** 1142 | | **Barcode:** ‖‖‖‖ | |

| | | |
|---|---|---|
| **BOL ID:** 1355 | **Shipment:** 1130 | **Motive:** RTV |
| **Create Date:** 08/24/2017 | **Create User:** qa_004 | |

| **Sender** | **Receiver** |
|---|---|
| 1511 - Phoenix<br>123 Street<br>Anytown<br>Anycity    MN    50250<br>US | 1200 - Fashion Importer (Euro)<br>gggg<br>gggg<br>ggg<br>MN    89<br>US |
| **Ship From** | **Ship To** |
| Phoenix<br>123 Street<br>Anytown<br>Anycity    MN    50250<br>US | Fashion Importer (Euro)<br>gggg<br>gggg<br>ggg<br>MN    89<br>US    Phone: |

| **Carrier**<br> ?Sender    ?Receiver    ?Third Party<br>Carrier Name:<br>Carrier Address: | Requested Pick-Up Date<br><br>Carrier Signature:<br><br><br>Dispatch Date: |
|---|---|
| Service: | Tax ID: |

| **Container ID** | **Weight (UOM)** | **Package Type** | **Tracking ID** |
|---|---|---|---|
| 000120054129 | | | |

<div align="center">Legalese fine print</div>

**Notes**

| **Ship Container No:** 000120054129 | **Barcode :** ‖‖‖‖‖‖‖‖‖ | | | |
|---|---|---|---|---|
| **EAN** | **Item** | **Description** | **UOM** | **Quantity** |
| | SIM_13 | SIM_13 | Cases | 2 |

| Driver signature | Date | Receiver Signature | Date |
|---|---|---|---|
| | | | |

**Figure D-37    Vendor Shipment Carton Report**

RTV Shipment Container Report

**Source:** 1511 - Phoenix
**Supplier:** 1200 - Fashion Importer (Euro)
**Ship Date:**
**Shipment Number:** 1130
**Authorization Number:** 987878
**Status:** In Progress
**User:**
**Not After Date:** 08/23/2017
**Container:** 000120054129
**Container Status:** In Progress

| Item | Description | UOM | Pack Size | Ship Qty | Reason Code |
|------|-------------|------|-----------|----------|-------------|
| SIM_13 | SIM_13 | Cases | 1 | 2 | Overstock |

**Printed:** 1/3/2018 **Page Number:** 1

**Figure D-38    Vendor Shipment Report**

**RTV Shipment Report**

| | |
|---|---|
| **Source:** | 1511 - Phoenix |
| **Supplier:** | 1200 - Fashion Importer (Euro) |
| **Ship Date:** | |
| **Shipment Number:** | 1130 |
| **RTV Type:** | |
| **Authorization Number:** | 987878 |
| **Status:** | In Progress |
| **User:** | |

**Container ID:** 000120054129                    **Status:** In Progress

| Item | Description | UOM | Pack Size | Ship Qty | Reason Code |
|------|-------------|-----|-----------|----------|-------------|
| SIM_13 | SIM_13 | Cases | 1 | 2 | Overstock |

**Printed:** 1/3/2018                                                              **Page Number:** 1

**Figure D-39    Vendor Shipment Label**

| From<br>Phoenix<br>123 Street<br>Anytown<br>Anycity MN  50250<br>US | To<br>Fashion Importer (Euro)<br>gggg<br>gggg<br>MN<br><br>89<br>US |
|---|---|
| **(420) Ship To Postal Code**<br><br>(420)89<br>▐▌▐ | **Label Type**<br><br>**Return To Vendor** |
| **Return ID  8789**<br><br>▐▌▌▐▌▐ | |
| **(01)1200**<br><br>▐█▌▐█▌█▐ | **1200** |
| **SSCC -18**<br>000120058020<br>▐█▌▐█▌▐█▌█▐█▌▐█▌ | |

# E

# Appendix: Batch File Layout Specifications

This chapter describes the batch file layout specification.

## Third Party Stock Count Parser Import File Layout Specification

### Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat

Where file prefix value is STK and loc id is the store identifier. This allows file to be unique for every upload.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

STK_20180129133250_1111.dat - Data File.

STK_20180129133250_1111.dat.complete - Complete File.

### Zip Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>.zip

Where file prefix value is STK. The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

STK_20180129133250.zip - Zip File.

STK_20180129133250.zip.complete - Complete File.

### Third Party Stock Count Import File Format

Pipe-delimited (|) file contains store count data for a store and stock count ID as shown in the File Layout table.

# File Layout

**Table E-1    Third Party Stock Count Import File**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FHEAD | Record Descriptor | Char(5) | FHEAD | File head marker |
| | Store Number | Char(10) | | Store number file was uploaded for. It is assumed only one store is passed in per file. (Required) |
| | Stock Count ID | Number(12) | | Unique identifier for item. Assumption is application will always take first stock count ID listed. (Required) |
| FDETL | Record Descriptor | Char(5) | FDETL | Detail record marker. |
| | Stock Count Date | Date(14) | | Indicates date/time item was physically counted by the third party. (*YYYYMMDDHH24MISS*) For example, 20180129134600 (Required) **Note:** If not using timestamp, use **00** for time. |
| | Area Number | Char(10) | | 10-digit code indicating where in the store the item is located. (Optional) |
| | UPC or Item Number | Char(25) | | 25-digit universal product code. (Required) |
| | Count Quantity | Number (12,4) | | Quantity counted for item, required. This field must allow for decimals when counting in UOM other than each. (Required) |
| | UIN(Item Serial Number) | Char(128) | | Unique identification serial number for item, required if current item requires serial number. |
| FTAIL | Record Descriptor | Char(5) | FTAIL | File tail marker. |

The following is a sample Third Party Stock Count Import File:

```
FHEAD|5000|1074|
FDETL|20180129235959|1|100665085|1|ItemSerialNum1234|
FDETL|201180129140000|1|100665085|1|ItemSerialNum9999|
FDETL|20180129000000|1|100665085|1||
FTAIL|
```

# Clearance Price Change Import File Layout Specification

## Filename Format

<event type>_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>.dat

Where event type value is "CLRPC" (clearances), the loc type field would be either 'S' or 'W'.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

CLRPC_20180129133250_1111_S.dat - Data File.

CLRPC_20180129133250_1111_S.dat.complete - Complete File.

## Zip Filename Format

<event type>_<date in YYYYMMDDHH24MISS format>_<loc type>.zip

Where event type value is "CLRPC". The loc type field would be either 'S' or 'W'.

The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

CLRPC_20180129133250_S.zip - Zip File.

CLRPC_20180129133250_S.zip.complete - Complete File.

## File Record Types

- FHEAD (required): File identification, one line per file.
- FDETL (optional): Price Change Event (Create or Modify).
- FDELE (optional): Price Change Event (Delete).
- FTAIL (required): End of file marker, one line per file.

## File Layout

**Table E-2    Clearance Price Change File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FHEAD | Record Descriptor | Char(5) | FHEAD | File head marker |
| FHEAD | Line ID | Number(10) | 1 | Unique line ID |
| FHEAD | File Type | Char(5) | CLRPC | Clearance Price Changes |
| FHEAD | Export timestamp | Timestamp | | System clock timestamp (YYYYMMDDHHMISS) |

**Table E-2    (Cont.) Clearance Price Change File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FHEAD | Location | Number(10) | | Location identifier |
| FHEAD | Location Type | Char(1) | | S = Store, W = Warehouse |
| FDETL | Record Descriptor | Char(5) | FDETL | File Detail Marker (one per clearance create or modify) |
| FDETL | Line ID | Number(10) | | Unique line ID |
| FDETL | Event Type | Char(3) | | **CRE** = Create **MOD** = Modify |
| FDETL | ID | Number(15) | | Clearance identifier |
| FDETL | Item | Char(25) | | Item identifier |
| FDETL | Effective Date | Date | | Clearance Effective Date (YYYMMDDHHMISS) |
| FDETL | Selling Retail | Number(20,4) | | Selling retail with price change applied |
| FDETL | Selling Retail UOM | Char(4) | | Selling retail unit of measure |
| FDETL | Selling Retail Currency | Char(3) | | Selling retail currency |
| FDETL | Reset Clearance ID | Number(15) | | ID of clearance reset |
| FDELE | Record Descriptor | Char(5) | FDELE | File Detail Delete Marker (one per clearance delete) |
| FDELE | Line ID | Number(10) | | Unique line ID |
| FDELE | ID | Number(15) | | Clearance identifier |
| FDELE | Item | Char(25) | | Item identifier |
| FTAIL | Record Descriptor | Char(5) | FTAIL | File tail marker |
| FTAIL | Line ID | Number(10) | | Unique line ID |
| FTAIL | Number of lines | Number(10) | | Number of lines in file not counting FHEAD and FTAIL |

# Regular Price Change Import File Layout Specification

## Filename Format

<event type>_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>.dat

Where event type is "REGPC" (regular price changes). The loc type field would be either 'S' or 'W'.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

REGPC_20180129133250_1111_S.dat - Data File.

REGPC_20180129133250_1111_S.dat.complete - Complete File.

## Zip Filename Format

<event type>_<date in YYYYMMDDHH24MISS format>_<loc type>.zip

Where event type value is "REGPC". The loc type field would be either 'S' or 'W'.

The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

REGPC_20180129133250_S.zip - Zip File.

REGPC_20180129133250_S.zip.complete - Complete File.

## File Record Types

- FHEAD (required): File identification, one line per file.
- FDETL (optional): Price Change Event (Create or Modify).
- FDELE (optional): Price Change Event (Delete).
- FTAIL (required): End of file marker, one line per file.

## File Layout

**Table E-3    Regular Price Change File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FHEAD | Record Descriptor | Char(5) | FHEAD | File head marker |
| FHEAD | Line ID | Number(10) | 1 | Unique line ID |
| FHEAD | File Type | Char(5) | REGPC | Regular Price Changes |
| FHEAD | Export timestamp | Timestamp | | System clock timestamp (YYYYMMDDHHMISS) |
| FHEAD | Location | Number(10) | | Location identifier |
| FHEAD | Location Type | Char(1) | | S = Store W= Warehouse |
| FDETL | Record Descriptor | Char(5) | FDETL | File Detail Marker (one per price change create or modify) |
| FDETL | Line ID | Number(10) | | Unique line ID |
| FDETL | Event Type | Char(3) | | **CRE** = Create **MOD** = Modify |
| FDETL | ID | Number(15) | | Price Change identifier |
| FDETL | Item | Char(25) | | Item identifier |

**Table E-3    (Cont.) Regular Price Change File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
| --- | --- | --- | --- | --- |
| FDETL | Effective Date | Date | | Effective Date of price change (YYYYMMDDHHMISS) |
| FDETL | Selling Unit Change Ind | Number(1) | | Did selling unit retail change with this price event (**0** = no change, **1** = changed) |
| FDETL | Selling Retail | Number(20,4) | | Selling retail with price change applied |
| FDETL | Selling Retail UOM | Char(4) | | Selling retail unit of measure |
| FDETL | Selling Retail Currency | Char(3) | | Selling retail currency |
| FDETL | Multi-Unit Change Ind | Number(1) | | Did multi unit retail change with this price event (**0** = no change, **1** = changed) |
| FDETL | Multi-Units | Number(12,4) | | Number Multi Units |
| FDETL | Multi-Unit Retail | Number(20,4) | | Multi Unit Retail |
| FDETL | Multi-Unit UOM | Char(4) | | Multi Unit Retail Unit Of Measure |
| FDETL | Multi-Unit Currency | Char(3) | | Multi Unit Retail Currency |
| FDELE | Record Descriptor | Char(5) | FDELE | File Detail Delete Marker (one per price change delete) |
| FDELE | Line ID | Number(10) | | Unique line ID |
| FDELE | ID | Number(15) | | Price Change identifier |
| FDELE | Item | Char(25) | | Item identifier |
| FTAIL | Record Descriptor | Char(5) | FTAIL | File tail marker |
| FTAIL | Line ID | Number(10) | | Unique line ID |
| FTAIL | Number of lines | Number(10) | | Number of lines in file not counting FHEAD and FTAIL |

# Promotion Price Change Import File Layout Specification

## Filename Format

<event type>_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>.dat

Where event type is "PRMPC" (promotional price change). The loc type field would be either 'S' or 'W'.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

PRMPC_20180129133250_1111_S.dat - Data File.

PRMPC_20180129133250_1111_S.dat.complete - Complete File.

## Zip Filename Format

<event type>_<date in YYYYMMDDHH24MISS format>_<loc type>.zip

Where event type value is "PRMPC". The loc type field would be either 'S' or 'W'.

The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

PRMPC_20180129133250_S.zip - Zip File.

PRMPC_20180129133250_S.zip.complete - Complete File.

## File Record Types

- FHEAD (required): File identification, one line per file.
- TMBPE (optional): Event Type.
- TPDTL (required): Promotion Component Detail.
- TPGRP (required): Promotion Group (one or more per TPDTL).
- TGLST (required): Promotion List (one or more per TPGRP).
- TLITM (required): Promotion Item (one or more per TGLST).
- TPDSC (required): Promotion Discount (one or more per TGLST).
- TPISR (optional): Promotion Item Selling retail (one or more per TPDTL).
- TPCDT (optional): Promotion Credit Detail (one or more per TPDTL).
- TTAIL (required): Transaction tail (one per promotion).
- TPCIL (optional): Promotion Cancel Item Location
- FPDEL (optional): Promotion Delete.
- FTAIL (required): End of file marker, one line per file.

## File Layout

**Table E-4    Promotion Price Change Output File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
| --- | --- | --- | --- | --- |
| FHEAD | Record Descriptor | Char(5) | FHEAD | File head marker |
| FHEAD | Line ID | Number(10) | 1 | Unique line Identifier |
| FHEAD | File Type | Char(5) | PROMO | Promotions |
| FHEAD | Export timestamp | Timestamp | | System clock timestamp (YYYYMMDDHHMISS) |

**Table E-4    (Cont.) Promotion Price Change Output File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FHEAD | Location | Number(10) | | Location identifier |
| FHEAD | Location Type | Char(1) | | S = Store, W = Warehouse |
| TMBPE | Record Descriptor | Char(5) | TMBPE | Promotion (transaction head) |
| TMBPE | Line ID | Number(10) | | Unique line identifier |
| TMBPE | Event Type | Char(3) | | **CRE** = Create<br>**MOD**= Modify |
| TPDTL | Record Descriptor | Char(5) | TPDTL | Promotion Detail Component |
| TPDTL | Line ID | Number(10) | | Unique line identifier |
| TPDTL | Promo ID | Number(10) | | Promotion identifier |
| TPDTL | Promo Comp ID | Number(10) | | Promotion Component ID |
| TPDTL | Promo Name | Char(160) | | Promotion Header Name |
| TPDTL | Promo Desc | Char(640) | | Promotion Header Description |
| TPDTL | Promo Comp Desc | Char(160) | | Promotion Component Name |
| TPDTL | Promo Type | Number(2) | | Valid values will be modified to be as follows: 0 - Multi-Buy Promotion; 1 - Simple Promotion; 2 - Threshold Promotion; 3 - Finance Promotion (formerly tied to a value of 6); 4 - Transaction Promotion |
| TPDTL | Promo Comp Detail ID | Number(10) | | Promotion Component Detail identifier |
| TPDTL | Start Date | Date | | Start Date of Promotion Component Detail (YYYYMMDDHH24MISS) |
| TPDTL | End Date | Date | | End Date of Promotion Component Detail (YYYYMMDDHH24MISS) |
| TPDTL | Apply To Code | Number(1) | | Holds the apply to code for the promotion detail. Determines if the promotion is applied to regular retail only (no clearances in effect), clearance retail only (only when a clearance is in effect) or both regular and clearance retail. Valid values are 0 - Regular Only; 1 - Clearance Only; 2 - Regular and Clearance. |
| TPDTL | Discount Limit | Number(3) | | The number of times that the promotion that can be applied to a transaction. |

**Table E-4    (Cont.) Promotion Price Change Output File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| TPDTL | Apply Order | Number(1) | | Application Order of the Promotion |
| TPDTL | Threshold ID | Number(6) | | Threshold identifier |
| TPDTL | Customer Type ID | Number(10) | | Customer Type identifier |
| TPDTL | Threshold Qualification Type | Number(1) | | The qualification type for the threshold. Will only be populated for threshold promotions. Valid values are 0 for item level and 1 for threshold level. |
| TPDTL | Exception Parent ID | Number(10) | | The promotion component detail ID for the parent event that this event is a child to. Used only for transaction promotion detail exclusions to be able to connect the child exclusion to it's parent promotion detail. |
| TPGRP | Record Descriptor | Char(5) | TPGRP | Promotion Detail Group |
| TPGRP | Line ID | Number(10) | | Unique line identifier |
| TPGRP | Group ID | Number(10) | | Group Number |
| TGLIST | Record Descriptor | Char(5) | TGLIST | Promotion Group List |
| TGLIST | Line ID | Number(10) | | Unique line identifier |
| TGLIST | List ID | Number(10) | | List identifier |
| TGLIST | Reward Application | Number(1) | | How this reward is applied to the promotion detail. |
| TGLIST | Description | Char(120) | | Description |
| TGLIST | Price Range Min | Number (20,4) | | Contains price range promotion minimum valid retail value. |
| TGLIST | Price Range Max | Number (20,4) | | Contains price range promotion maximum valid retail value. |
| TLITM | Record Descriptor | Char(5) | TLITM | Promotion Group List |
| TLITM | Line ID | Number(10) | | Unique line identifier |
| TLITM | Item ID | Char(25) | | Transaction Item Identifier |
| TPDSC | Record Descriptor | Char(5) | TPDSC | Discount Detail for List |
| TPDSC | Line ID | Number(10) | | Unique line identifier |
| TPDSC | Change Type | Number(2) | | Change Type |
| TPDSC | Change Amount | Number(20,4) | | Change Amount |
| TPDSC | Change Currency | Char(3) | | Change Currency |
| TPDSC | Change Percent | Number(20,4) | | Change Percent |
| TPDSC | Change Selling UOM | Char(4) | | Change Selling UOM |

**ORACLE**

**Table E-4 (Cont.) Promotion Price Change Output File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| TPDSC | Qual Type | Number(2) | | Qualification Type |
| TPDSC | Qual Value | Number(2) | | Qualification Value |
| TPDSC | Change Duration | Number(20,4) | | Change Duration |
| TPISR | Record Descriptor | Char(5) | TPISR | Items in Promotion |
| TPISR | Line ID | Number(10) | | Unique line identifier |
| TPISR | Item ID | Char(25) | TTAIL | Transaction Item Identifier |
| TPISR | Selling Retail | Number(20,4) | | Selling retail of the item |
| TPISR | Selling UOM | Char(4) | | Selling UOM of the item |
| TPISR | Effective Date | Date | | Effective Date of the selling retail - YYYYMMDDHH24MISS |
| TPISR | Selling Retail Currency | Char(3) | | Selling retail currency |
| TPISR | Referenced Promo Detail ID | Number(15) | | Referenced promotion detail ID. This is an optional field. |
| TPCDT | Record Descriptor | Char(5) | TPCDT | Credit Detail |
| TPCDT | Credit Detail ID | Number(10) | | Credit Detail ID |
| TPCDT | Line ID | Number(10) | | Unique line ID |
| TPCDT | Credit Type | Char(40) | | Credit Type |
| TPCDT | BinNumberFrom | Number(10) | | BinNumber From |
| TPCDT | BinNumberTo | Number(10) | | Bin Number To |
| TPCDT | Commission Rate | Number(10) | | Commission Rate |
| TPCDT | Comments | Char(160) | | Comments |
| TTAIL | Record Descriptor | Char(5) | TTAIL | Transaction Tail |
| TTAIL | Line ID | Number(10) | | Unique line identifier |
| TPCIL | Record Descriptor | Char(5) | TPCIL | Cancel Item Loc |
| TPCIL | Line ID | Number(10) | | Unique line identifier |
| TPCIL | Promo ID | Number(10) | | The ID of the promotion |
| TPCIL | Promo Comp ID | Number(10) | | Promotion Component ID |
| TPCIL | Promo Comp Detail ID | Number(10) | | Promotion Component Detail identifier |
| TPCIL | Item ID | Char(25) | | Transaction Item Identifier for item |
| TPCIL | Cancellation Date | Date | | Cancellation effective date - YYYYMMDDHH24MISS |
| FPDEL | Record Descriptor | Char(5) | FPDEL | Delete Promotion |
| FPDEL | Line ID | Number(10) | | Unique line identifier |
| FPDEL | Promo ID | Number(10) | | The ID of the promotion |
| FPDEL | Promo Comp ID | Number(10) | | Promotion Component ID |

**Table E-4    (Cont.) Promotion Price Change Output File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FPDEL | Promo Comp Detail ID | Number(10) | | Promotion Component Detail ID |
| FPDEL | Group ID | Number(10) | | Group Number |
| FPDEL | List ID | Number(10) | | List ID |
| FPDEL | Item ID | Char(25) | | Transaction Item Identifier for item |
| FTAIL | Record Descriptor | Char(5) | FTAIL | File tail marker |
| FTAIL | Line ID | Number(10) | | Unique line identifier |
| FTAIL | Number of lines | Number(10) | | Number of lines in file not counting FHEAD and FTAIL |

# POS Sale Transaction Import File Specification

## Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat

Where file prefix value is "SIMTLOG" and loc id is the store identifier. This allows file to be unique for every upload.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

SIMTLOG_20180129133250_1111.dat - Data File.

SIMTLOG_20180129133250_1111.dat.complete - Complete File.

## Zip Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>.zip

Where file prefix value is "SIMTLOG". The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

SIMTLOG_20180129133250.zip - Zip File.

SIMTLOG_20180129133250.zip.complete - Complete File.

## POS Sale Transaction Import File Format

The input file would be in Pipe ('|') delimited format.

# File Layout

**Table E-5    T-LOG File**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FILE HEADER | FILE HEADER File Type Record Descriptor | VARCHAR2(5) | FHEAD | Identifies the File Record Type |
| FILE HEADER | Location Number | NUMBER(10) | | Store Number |
| FILE HEADER | Business Date | VARCHAR2(14) | | Business Date of transactions in YYYYMMDDHHSS format |
| FILE HEADER | File Creation Date | VARCHAR2(14) | SYSDATE | File Create Date in YYYMMDDHHMSS format |
| TRANSACTION HEADER | File Type Record Descriptor | VARCHAR2(5) | THEAD | Identifies the File Record Type |
| TRANSACTION HEADER | Transaction Number | VARCHAR2(128) | | The unique transaction reference number generated by ORXPOS/OMS. |
| TRANSACTION HEADER | Transaction Date and Time | VARCHAR2(14) | | Date transactions were processed in ORXPOS/OMS |
| TRANSACTION HEADER | Customer Order ID | VARCHAR2(128) | | External customer order ID, if transaction is a customer order |
| TRANSACTION HEADER | Customer Order Comments | VARCHAR(512) | | Comments on the customer order |
| TRANSACTION DETAIL | File Type Record Descriptor | VARCHAR2(5) | TDETL | Identifies the File Record Type |
| TRANSACTION DETAIL | Item ID | VARCHAR2(25) | | ID number of the item. |
| TRANSACTION DETAIL | UIN | VARCHAR2(128) | | This is the UNIQUE_ID value from RTLOG |
| TRANSACTION DETAIL | Item Quantity | NUMBER(12,4) | | Quantity of the item on this transaction |
| TRANSACTION DETAIL | Selling UOM | VARCHAR2(4) | | UOM at which this item was sold |
| TRANSACTION DETAIL | Reason Code | NUMBER(4) | | Reason entered by cashier for some transaction types. Required for voids, returns, for example. |
| TRANSACTION DETAIL | Comments | VARCHAR(512) | | Comments for this line item |

**ORACLE**

**Table E-5    (Cont.) T-LOG File**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| TRANSACTION DETAIL | Transaction Code | VARCHAR2(25) | | The type of sale represented by this line item. Valid value are SALE,RETURN,VOID_SALE, VOID_RETURN,ORDER_NEW,ORDER_FULFILL,ORDER _CANCEL,ORDER_CANCEL _FULFILL |
| TRANSACTION DETAIL | Reservation Type | VARCHAR(25) | | Reservation type if POS transaction is a customer order. Valid values are SPECIAL_ORDER, WEB_ORDER, PICKUP_AND DELIVERY,LAYAWAY |
| TRANSACTION DETAIL | Fulfillment Order Number | VARCHAR2(48) | | Fulfillment Order Number from OMS |
| TRANSACTION DETAIL | Drop Ship Indicator | VARCHAR(1) | | 'P' if it is drop ship otherwise 'N' |
| TRANSACTION TAIL | File Record Type Descriptor | VARCHAR2(5) | TTAIL | Identifies the File Record Type |
| TRANSACTION TAIL | Transaction Record Counter | NUMBER(6) | | Number of TDETL records in this transaction set. |
| FILE TAIL | File Record Type Descriptor | VARCHAR2(5) | FTAIL | Identifies the File Record Type |
| FILE TAIL | File Record Counter | NUMBER(10) | | Number of records/ transactions processed in current file (only records between head and tail) |

# Retail Sale Audit Import File Specification

## Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat

Where file prefix value is "SIMT" and loc id is the store identifier. This allows file to be unique for every upload.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

SIMT_20180129133250_1111.dat - Data File.

SIMT_20180129133250_1111.dat.complete - Complete File.

## Zip Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>.zip

Where file prefix value is "SIMT". The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

SIMT_20180129133250.zip - Zip File.

SIMT_20180129133250.zip.complete - Complete File.

## ReSA Transaction Import File Format

The input file would be in Pipe ('|') delimited format.

## File Layout

**Table E-6    ReSA File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FILE HEADER | FILE Type Record Descriptor | VARCHAR2(5) | FHEAD | Identifies the File Record Type |
| FILE HEADER | File Line ID | VARCHAR(10) | | Sequential file line number |
| FILE HEADER | File Type Definition | VARCHAR2(4) | SIMT | Identifies the File Type |
| FILE HEADER | Location Number | NUMBER(10) | | Store Number |
| FILE HEADER | Business Date | VARCHAR2(14) | | Business Date of transactions in YYYYMMDDHHSS format |
| FILE HEADER | File Creation Date | VARCHAR2(14) | SYSDATE | File Create Date in YYYMMDDHHMSS format |
| TRANSACTION HEADER | File Type Record Descriptor | VARCHAR2 (5) | THEAD | Identifies the File Record Type |
| TRANSACTION HEADER | File Line ID | VARCHAR(10) | | Sequential file line number |
| TRANSACTION HEADER | Transaction Number | NUMBER(10) | | The unique transaction reference number generated by ORXPOS/OMS |
| TRANSACTION HEADER | Register Number | NUMBER(10) | | The register reference number generated by ORXPOS |
| TRANSACTION HEADER | Revision Number | NUMBER(3) | | The version of the transaction being sent |
| TRANSACTION HEADER | Transaction Date and Time | VARCHAR2(14) | | Date transactions were processed in ORXPOS/OMS |

**Table E-6    (Cont.) ReSA File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| TRANSACTION HEADER | Transaction Type | VARCHAR2(14) | | Transaction Type Code (for example, SALE, RETURN, SPLORD) |
| TRANSACTION HEADER | Pos created flag | VARCHAR2(1) | | 'Y' identifies that the transaction occurred at ORXPOS, 'N' identifies that the transaction was created in ReSA |
| TRANSACTION DETAIL | File Type Record Descriptor | VARCHAR2(5) | TDETL | Identifies the File Record Type |
| TRANSACTION DETAIL | File Line ID | VARCHAR(10) | | Sequential file line number. |
| TRANSACTION DETAIL | Item Sequence Number | NUMBER(4) | | The order in which items were entered during a transaction |
| TRANSACTION DETAIL | Item | VARCHAR2(25) | | ID number of the item. |
| TRANSACTION DETAIL | Item Type | VARCHAR2(6) | | Type of Item sold. Can be 'ITEM', 'REF', 'GCN', 'NMITEM' |
| TRANSACTION DETAIL | Reference Item | VARCHAR2(25) | | Identified sub-transaction level merchandise item. |
| TRANSACTION DETAIL | Non-Merchandise Item | VARCHAR2(25) | | Identifies non-merchandise item |
| TRANSACTION DETAIL | Item Status | VARCHAR2(6) | | Status of the item within the transaction, V for item void, S for sold item, R for returned item, Layaway Initiate (LIN), Layaway Cancel, Layaway Complete (LCO), Order Initiate (ORI), Order Cancel (ORC) Order Complete (ORD) |
| TRANSACTION DETAIL | Serial Number | VARCHAR2(128) | | This is the UNIQUE_ID value from RTLOG |
| TRANSACTION DETAIL | Pack Indicator | VARCHAR2(1) | | Pack indicator of item sold or returned |
| TRANSACTION DETAIL | Catch Weight Indicator | VARCHAR2(1) | | Indicates if item is a catchweight item |
| TRANSACTION DETAIL | Item Quantity Sign | VARCHAR2(1) | | Determines if the Total Sale Quantity is positive or negative 'P' - Positive 'N' - Negative |
| TRANSACTION DETAIL | Item Quantity Value | NUMBER(20) | | Total sales value of goods sold/returned (4 implied decimal places), for example, Total Quantity * 10000 |

**ORACLE**

**Table E-6    (Cont.) ReSA File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| TRANSACTION DETAIL | Standard UOM | VARCHAR2(4) | | Standard UOM of the Item |
| TRANSACTION DETAIL | Selling UOM | VARCHAR2(4) | | UOM at which this item was sold |
| TRANSACTION DETAIL | Wastage Type | VARCHAR2(6) | | Wastage type of item sold or returned |
| TRANSACTION DETAIL | Wastage Percentage | NUMBER(12) | | Wastage Percent*10000 (4 implied decimal places), wastage percent of item sold or returned |
| TRANSACTION DETAIL | Drop Ship Indicator | VARCHAR2(1) | N | This will always be N for Export |
| TRANSACTION DETAIL | Actual Weight Quantity | NUMBER(12) | | Actual Weight Quantity*10000 (4 implied decimal places), the actual weight of the item, only populated if catchweight_ind = 'Y' |
| TRANSACTION DETAIL | Reason Code | VARCHAR2(6) | | Reason entered by cashier for some transaction types |
| TRANSACTION DETAIL | Sale Value | NUMBER(20) | | Total Sales Value * 10000 (4 implied decimal places), sales value, net sales value of goods sold |
| TRANSACTION DETAIL | Sales Sign | VARCHAR2(1) | | Determines if the Total Sales Value is positive or negative 'P' - Positive 'N' - Negative |
| TRANSACTION DETAIL | Unit Retail | NUMBER(20,4) | | Unit retail with 4 implied decimal places |
| TRANSACTION DETAIL | Sales Type | VARCHAR2(1) | | Indicates if the line item is a Regular Sale, a CO serviced by OMS (External CO), or a CO serviced by Inventory management application (In-Store CO) |
| TRANSACTION DETAIL | Customer Order Number | VARCHAR2(50) | | Customer Order Number |
| TRANSACTION DETAIL | Customer Order Type | | | Customer order type |
| TRANSACTION DETAIL | Fulfillment Order Number | VARCHAR2(50) | | Fulfillment Order Number from OMS |
| TRANSACTION TAIL | File Record Type Descriptor | VARCHAR2(5) | TTAIL | Identifies the File Record Type |
| TRANSACTION TAIL | File Line ID | NUMBER(10) | | Sequential file line number |

**Table E-6    (Cont.) ReSA File Layout**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| TRANSACTION TAIL | Transaction Record Counter | NUMBER(6) | | Number of TDETL records in this transaction set |
| FILE TAIL | File Record Type Descriptor | VARCHAR2(5) | FTAIL | Identifies the File Record Type |
| FILE TAIL | File Line ID | NUMBER(10) | | Sequential file line number |
| FILE TAIL | File Record Counter | NUMBER(10) | | Number of records/ transactions processed in current file (only records between head and tail) |

# Stock Count Results Export File Specification

The stock count result export file is generated when unit amount stock count authorization completes. The stock count authorization process can be a manual authorization or invoked by third party stock count batch for an auto-authorized unit amount stock count. This export file can be uploaded to RMS by RMS file to update their inventory with the actual physical stock count.

**Table E-7    Stock Count Export File**

| Record Name | Field Name | Field Type | Description |
|---|---|---|---|
| File Header | file type record descriptor | Char(5) | hardcode FHEAD |
| File Header | file line identifier | Number(10) | ID of current line being processed, hardcode **000000001** |
| File Header | file type | Char(4) | hardcode STKU |
| File Header | file create date | Date(14)YYYYMMDDHHMISS | date written by convert program |
| File Header | stocktake_date | Date(14)YYYYMMDDHHMISS | take_head.stocktake_date |
| File Header | cycle count | Number(8) | stake_head.cycle_count |
| File Header | loc_type | Char(1) | hardcode **W** or **S** |
| File Header | location | Number(10) | stake_location.wh or stake_location.store |
| Transaction record | file type record descriptor | Char(5) | hardcode **FDETL** |
| Transaction record | file line identifier | Number(10) | ID of current line being processed, internally incremented |
| Transaction record | item type | Char(3) | hardcode **ITM** |
| Transaction record | item value | Char(25) | item ID |

**Table E-7    (Cont.) Stock Count Export File**

| Record Name | Field Name | Field Type | Description |
| --- | --- | --- | --- |
| Transaction record | inventory quantity | Number(12,4) | total units or total weight |
| Transaction record | location description | Char(30) | Where in the location the item exists. For example, Back Stockroom or Front Window Display |
| File trailer | file type record descriptor | Char(5) | hardcode **FTAIL** |
| File trailer | file line identifier | Number(10) | ID of current line being processed, internally incremented |
| File trailer | file record count | Number(10) | Number of detail records |

# Store Sequence Data Parser Import File Layout Specification

Sequencing functionality provides users the ability to know the relative location of an item in a store. Sequencing a store improves store processes and reduces the time that employees spend looking for items. The retailer can sequence all items in the store and create unique locations to hold the items.

Sequencing defines how many items can be stored in a particular location, and allows the definition of a capacity for that item location combination. The capacity is used for in-store replenishment when generating the shelf replenishment pick list. Sequencing is used within Stock Counts, Customer Order Picking, Transfer Request, and Shelf Replenishment to aid the user in proceeding to the next item during the transaction for efficiency. Lastly, the Sequencing Primary Location is displayed to the user on the Item Detail screen.

## Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat

Where file prefix value is "SSEQ" and loc id is the store identifier. This allows file to be unique for every upload.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

SSEQ_20180129133250_1111.dat - Data File.

SSEQ_20180129133250_1111.dat.complete - Complete File.

## Zip Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>.zip

Where file prefix value is "SSEQ". The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.

Example:

SSEQ_20180129133250.zip - Zip File.

SSEQ_20180129133250.zip.complete - Complete File.

## Store Sequence Import File Format

The input file would be in pipe ('|') delimited format.

## File Layout

**Table E-8    Store Sequence Import File**

| Record Name | Field Name | FieldType | Description |
| --- | --- | --- | --- |
| File Header | file type record descriptor | Char(5) | hardcode FHEAD |
| File Header | Store ID | Number(10) | Store identifier |
| Sequence record | file type record descriptor | Char(5) | hardcode SHEAD |
| Sequence record | Area type | Number(9) | The Store Sequence Area. 0 = None, 1 = Shopfloor, 2 = Backroom |
| Sequence record | Child sequenced | Varchar2(1) | 'Y' if child is sequenced, 'N' if not |
| Sequence record | Department ID | Number(12) | Department ID |
| Sequence record | Class ID | Number(12) | Class ID |
| Sequence record | Description | Varchar2(255) | Description of Store Sequence |
| Sequence record | Not sequenced | Varchar2(1) | Y indicates a default sequence containing all items that have not been sequenced elsewhere |
| Sequence record | Sequence Order | Number(20) | The order the store sequence is in compared to other store sequences |
| Sequence detail | file type record descriptor | Char(5) | hardcode SDETL |
| Sequence detail | Item ID | Varchar2(25) | Item ID |
| Sequence detail | Primary location | Varchar2(1) | Indicator if the location specified is the primary location for the item, Y if is primary location for item, N otherwise |
| Sequence detail | Item sequence order | Number(20) | Order of item within store sequence |

**Table E-8　(Cont.) Store Sequence Import File**

| Record Name | Field Name | FieldType | Description |
|---|---|---|---|
| Sequence detail | Capacity | Number(11,2) | The size of the location appropriate to unit of measure |
| Sequence detail | Ticket quantity | Number(11,2) | The quantity of tickets that need to be printed or used for the item inventory location |
| Sequence detail | Ticket format ID | Number(10) | Item ticket format identifier |
| Sequence detail | Uom mode | Number(2) | The Unit Of Measure display mode: 1 = Units, 2 = Cases |
| Sequence detail | Width | Number(12) | Width value to indicate how many items can fit across the width of the shelf |
| Sequence trailer | File type record descriptor | Char(5) | hardcode STAIL |
| File trailer | File type record descriptor | Char(5) | hardcode FTAIL |

# Warehouse Stock Import File Layout Specification

## Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>.dat

Where file prefix value is "WHIMS". The prefix and date combination allows file to be unique for every upload.

For each file, there need to be a .complete file present to job to process the file. The complete file can be any empty. This is to notify the job that uploading of data file is completed.

Example:

WHIMS_20180129133250_1111.dat - Data File.

WHIMS_20180129133250_1111.dat.complete - Complete File.

## Zip Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>.zip

Where file prefix value is "WHIMS". The zip file can contain one or more files. The complete file needs to be added for zip file for job to pick it for processing.

Example:

WHIMS_20180129133250.zip - Zip File.

WHIMS_20180129133250.zip.complete - Complete File.

## Warehouse Stock Import File Format

The input file would be in Pipe ('|') delimited format.

## File Layout

**Table E-9    Warehouse Stock Import File**

| Record Name | Field Name | Field Type | Default Value | Description |
| --- | --- | --- | --- | --- |
| 1 | Warehouse ID | Number(10) | NA | ID of the Warehouse, for example, 7000 |
| 2 | Item ID | Varchar2(25) | NA | Item Identifier. |
| 3 | Stock On Hand | Number(12) | NA | Stock on hand of the item in the warehouse. |
| 4 | Status | Varchar2(2) | NA | Status code of the item. Example A - Active, D - Deleted, I - Inactive, C - Discontinued. |

# Initial Inventory Import File Layout Specification

## Filename Format

<file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat

Where file prefix value is EXTSTK_ and loc id is the store identifier. This allows file to be unique for every upload.

Example: EXTSTK_20180129133250_1111.dat - Data File.

## Initial Inventory Import File Format

Pipe-delimited (|) file contains store count data for a store as shown.

## File Layout

**Table E-10    Initial Inventory Import File**

| Record Name | Field Name | Field Type | Default Value | Description |
| --- | --- | --- | --- | --- |
| FHEAD | Record Descriptor | Char(5) | FHEAD | File head marker |
| | Store Number | Char(10) | | Store number file was uploaded for. It is assumed only one store is passed in per file. (Required) |

**Table E-10    (Cont.) Initial Inventory Import File**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| FDETL | Record Descriptor | Char(5) | FDETL | Detail record marker |
| | Upload Date | Date(14) | | Indicates date/time item was physically counted. (YYYYMMDDHH24MISS) For example, 20180129134600 (Required for UIN Records) |
| | Area Number | Char(10) | | 10-digit code indicating where in the store the item is located. (Optional) |
| | UPC or Item Number | Char(25) | | 25-digit universal product code. (Required) |
| | Count Quantity | Number(12, 4) | | Quantity counted for item, required. This field must allow for decimals when counting in UOM other than each. (Required) |
| | UIN(Item Serial Number) | Char(128) | | Unique identification serial number for item, required if current item requires serial number. |
| FTAIL | Record Descriptor | Char(5) | FTAIL | File tail marker |

The following is a sample Initial Inventory Import File:

```
FHEAD|5000|

FDETL|20180129235959|1|100665085|1|ItemSerialNum1234|

FDETL|201180129140000|1|100665085|1|ItemSerialNum9999|

FDETL|20180129000000|1|100665085|1||

FTAIL|
```

# Inventory Extract Export File Specification

The file generated Inventory extract contains current available stock on hand for all the item in a store. Single file will contain inventory for single store, when executed for all stores job will create single file per store and zip all the file to single zip archive.

## Filename Format

<file prefix>_<store id>_<date in YYYYMMDDHH24MISS format>.txt

Where file prefix value is "PRODUCT_LOCATION_INV" and the store id is unique store identifier.

<file prefix>_<date in YYYYMMDDHH24MISS format>.zip

Where file prefix value is "PRODUCT_LOCATION_INV".

# Inventory Extract File Format

The first row will contain the placeholder information of the data on the file separated by pipe ('|').

Example:

system_cd|location_cd|product_cd|available_qty|next_po_qty|next_po_date|
daily_sell_through_qty|sell_qty_multiple|minimum_sell_qty|shrink_rate|sales_velocity

The input file would be in Pipe ('|') delimited format.

# File Layout

**Table E-11    Inventory Extract File**

| Record Name | Field Name | Field Type | Default Value | Description |
|---|---|---|---|---|
| 1 | System Code | Varchar2 | INV | This field will always hold default value that is, INV |
| 2 | Location Code | Number | NA | This field holds the store identifier. |
| 3 | Product Code | Varchar2 | NA | This field hold the item identifier. |
| 4 | Available Quantity | Number | NA | This field holds the current available stock on hand for the item in store. |
| 5 | Next PO Quantity | Number | 0 | This field will hold next purchase order quantity. |
| 6 | Next PO Date | Varchar2 | NA | This field hold the next purchase order date. This field is currently unused and will be empty. |
| 7 | Daily Sell Through Quantity | Number | NA | This field is currently unused and will be empty. |
| 8 | Sell Qty Multiple | Number | NA | This field is currently unused and will be empty. |
| 9 | Minimum Sell Qty | Number | NA | This field is currently unused and will be empty. |
| 10 | Shrink Rate | Number | NA | This field is currently unused and will be empty. |
| 11 | Sales Velocity | Number | NA | This field is currently unused and will be empty. |

The following is a sample Inventory Extract File:

system_cd|location_cd|product_cd|available_qty|next_po_qty|next_po_date|
daily_sell_through_qty|sell_qty_multiple|minimum_sell_qty|shrink_rate|sales_velocity

INV|5000|100665050|10|0||||||

INV|5000|100665051|10|0||||||

INV|5000|100665052|10|0||||||

# F

# Appendix: Auto-Authorized Third-Party Stock Count Process Overview

This section describe overview steps to setup and auto authorize a third party stock count:

1. In the **Operations/Product Group** dialog, create and save a new product group with the following attributes:

   **Type**: Select Unit or Unit and Amount

   **Counting Method**: Select Third Party

   A**uto Authorize**: Select this check box

   > ✎ **Note:**
   >
   > If auto authorize is selected, the processing of the stock count will attempt to do many automated steps when loading the third party stock count information. If auto authorize is not selected, after loading the file information the authorization process is manual.

2. In the **Operations/Product Group Component** dialog, update the created product group with the desired items to count and save. To count all items in all departments, set **All Department** attribute to **Yes**.

3. In the **Operations/Product Group Schedule** dialog, create a product group schedule for the previous created product group.

   > ✎ **Note:**
   >
   > If creating a schedule for a unit count that is active on the current date, you will have the option of generating the stock count immediately.

4. In the **Admin/Technical Maintenance/Job Admin** dialog, create and start a new job.

   Choose **Generate Unit Stock Count** to generate unit counts.

   Choose **Generate Unit and Amount Stock Count** to generate unit and amount counts.

> **Note:**
>
> After the generate stock count batch has completed, you can log onto the mobile application, and from the Main Menu, you can navigate to **Inventory Management / Stock Counts / Stock Count List** dialog. Select the generated stock count and you will notice stock count child records have been created for each department. The batch creates stock count groups for all items for all departments for the store, including items with SOH values of zero grouped by department. The stock count will be in new status, as will each of the child department records.

5. The next step of the process is to take a snapshot of the stock count. This is most often done manually but can also be done with an automated job. The snapshot must be taken before uploading the third-party flat file.

   **Manual**. On the mobile application, you will need to use the application to take the appropriate snapshot.

   **Automated**. For a unit and amount stock count, you can run the **Admin/Technical Maintenance/Job Admin** dialog previously used to generate the stock count, you can execute the **Stock Count Unit and Amount Snapshot** batch job.

> **Note:**
>
> Selecting **Take Snapshot** in the mobile application or running the batch job takes a snapshot of the current SOH figure and assigns this to every item in the stock count records. The snapshot button is displayed only if there is an extracted **Third Party Stock Count** or **Unit and Amount stock count** on the **Stock Count List** screen. You must first select at least one record from the **Third Party Stock Count** for the snapshot to be taken. Status of the stock count will change to In Progress. This will indicate that the snapshot has occurred. The user will not be able to access the stock count records until the file has been uploaded. If the user double-clicks one of the department stock counts on the list screen, the application will prompt with the message "The stock count will not be accessible until the import process has completed". The user will not be able to drill into the detail screen if the third-party file has not yet been imported into the application.

6. Once the snapshot is taken and the workforce is done counting the items, the appropriate third-party stock count file should be loaded into the system.

7. Once the third-party count file is in place, you can access the **Admin/Technical Maintenance/Job Admin** dialog and execute the **Third Party Stock Count Import** batch job.

> **Note:**
>
> When the batch is complete, each item within the count will be updated with the appropriate counted quantity and timestamps assigned. In addition, any item errors will be tracked and written to the database as rejected or unprocessed items. If auto authorize was not chosen, no further processing will take place. Authorization and rejected items management can then be dealt with.

# Third Party Processing

1. When the third-party file import process starts, it will attempt to snapshot the stock count if the snapshot has not already taken place. A failure to snapshot will stop the job from processing.

2. Next, it updates all the counted quantity and dates on all the items from the file information. A failure in this step stops the job from processing.

3. It then attempts to perform the completion of each child count without the stock count. Completing the count does business processing on the counted information and moves the status of each completed child to the authorize phase. Any failures that occur are logged and the processing is halted.

4. If auto-authorize was not selected, the processing halts as the files are loaded and count phase completed.

5. If auto-authorize was selected, the processing releases all current user activity locks on the stock count, so it is not being used during further processing.

6. If the stock count was for all items, the automated processing will attempt to find and correct any errors within the rejected items, such as items found but not ranged at the store. This part of the processing will then attempt to range the items.

7. The stock count is then marked ready to approve and so that it can begin final authorization.

8. The process approves each stock count child individually. The batch error log keeps track of each authorization failure. If any authorization failed among the children record the process halts.

9. If the stock count is unit and amount and authorization succeeded, the process attempts to create an export file.

# Third Party Recovery

1. Import Failure - If this occurs before or during loading the import file fails, you can begin the entire import process again.

2. Authorization Failure - If the import succeeds, but the authorization fails, you can run authorization recovery. Access the **Admin/Technical Maintenance/Job Admin** dialog and execute the **Stock Count Authorize Recovery** batch job.

# G

# Appendix: Unit and Amount Stock Counts Export

Unit and Amount Stock count authorization generates export file which can be uploaded to external inventory system. The stock count authorization process can be started by user through stock count authorization screen or be invoked by third party stock count batch for an auto-authorized unit amount stock count. The export files can be uploaded to merchandising system (for example, RMS) to update merchandising inventory with the actual physical stock count.

## Export File Layout

See the Stock Count Results Export File Specification for file layout details. The generated file will be zipped into an archive with same file naming standard followed for the file generation. A complete file is added once the generated file has been zipped.

## Export File Location

Export file directory is created by application installer. Integration admin will need to move the export data files from the application server export directory to a shared upload network location.

## Export File Name

STK_<store id>_<schedule id>_<date in YYYYMMDDHH24MISS format>.dat

STK_<store id>_<schedule id>_<date in YYYYMMDDHH24MISS format>.zip

STK_<store id>_<schedule id>_<date in YYYYMMDDHH24MISS format>.zip.complete

# H

# Appendix: UPC Barcode

UPC-E items compress a normal 12-digit UPC-A item into six digits. The application can decompress UPC-E barcodes to UPC-A. A seventh digit acts as a check digit for the UPC-E number. When the user scans the UPC-E barcode, the application finds the UPC-A barcode and displays the item ID associated with it.

## Differences between UPC-A and UPC-E

UPC-E is also called zero suppressed UPC because UPC-E compresses a normal twelve-digit UPC-A number into a six-digit code by suppressing the number system digit, trailing zeros in the manufacturers code and leading zeros in the product identification part of the bar code message. A seventh check digit is encoded into a parity pattern for the six main digits. UPC-E can thus be uncompressed back into a standard UPC-A twelve-digit number.

> **✎ Note:**
>
> Most bar code readers can be configured to automatically convert six-digit UPC-E numbers to twelve-digit UPC-A numbers before they are transmitted to a host computer.

The main difference between a UPC-A symbol and a UPC-E symbol is the size. The following image presents a UPC-A bar code (left) and the same data encoded as a UPC-E bar code (right):

**Figure H-1    UPC-A and UPC-E Differences**



To convert between UPC-A and UPC-E bar code numbers, you can use the following table or try online UPC-E converter program. In the following, the number 0 and each of the letters (a,

b, c, d and e) represent individual digits in the bar code message. The letter X represents the UPC check digit.

**Table H-1    UPC Conversion Table**

| UPC-A Number | Equivalent UPC-E | Notes |
| --- | --- | --- |
| 0ab00000cdeX | abcde0X | Manufacturer code must have two leading digits with three trailing zeros and the item number is limited to three digits (000 to 999). |
| 0ab10000cdeX | abcde1X | Manufacturer code must have three leading digits ending with 1 and two trailing zeros. The item number is limited to three digits. |
| 0ab20000cdeX | abcde2X | Manufacturer code must have three leading digits ending with 2 and two trailing zeros. The item number is limited to three digits. |
| 0abc00000deX | abcde3X | Manufacturer code must have three leading digits and two trailing zeros. The item number is limited to two digits (00 to 99). |
| 0abcd00000eX | abcde4X | Manufacturer code must have four leading digits with one trailing zero and the item number is limited to one digit (0 to 9). |
| 0abcde00005X | abcde5X | Manufacturer code has all five digits. The item number is limited to a single digit consisting of either 5, 6, 7, 8 or 9. |
| 0abcde00006X | abcde6X | |
| 0abcde00007X | abcde7X | |
| 0abcde00008X | abcde8X | |
| 0abcde00009X | abcde9X | |

# Conversion between UPC-A and UPC-E

Not all UPC-A numbers can be compressed to UPC-E. These codes with a corresponding UPC-E code must have at least four zeros. The requirements are:

1. 1. If the manufacturer code ends with 000, 100, or 200, the UPC-E code consists of the first two characters of the manufacturer code, the last three characters of the product code, followed by the third character of the manufacturer code. In this case, the product code must be 00000 and 00999.

2. If the manufacturer code ends with 00 but does not meet the first requirement, the UPC-E code consists of the first three characters of the manufacturer code, the last two characters of the product code, followed by digit 3. The product code can only contain two digits (00000 to 00099).

3. If the manufacturer code ends in 0 but none of the previous qualifies, the UPC-E consists of the first four digits of the manufacturer code and the last digit of the product code, followed by the digit 4. The product code in this case can only contain one digit (00000 to 00009).

4. If the manufacturer code ends with non-zero digit, the UPC-E code consists of the manufacturer code and the last digit of the product code. In this case the product case can only be one from 00005 to 00009 because 0 through 4 has been used for the previous four cases.

# Appendix: EICS Provided URLs

## EICS web-client URL

**Table I-1    EICS Application URL**

| | URL |
|---|---|
| EICS web-client | https://<eics_external_load_balanced_address>/sim-web-client |

## SOCS (connections config) URL

**Table I-2    SOCS Connections URL**

| | URL |
|---|---|
| SOCS (Connections Config) | https://<eics_external_load_balancer_address>/sim-mobile/ oracle.retail.sim.mobile.client.SimMobile/connections.xml |

## EICS Web Service URLs

**Table I-3    EICS Web Service URLs**

| SIM-WS | URL |
|---|---|
| | https://<eics_external_load_balancer_address>/ActivityLockBean/ActivityLockService?wsdl |
| | https://<eics_external_load_balancer_address>/FulfillmentOrderDeliveryBean/FulfillmentOrderDeliveryService?wsdl |
| | https://<eics_external_load_balancer_address>/FulfillmentOrderPickBean/FulfillmentOrderPickService?wsdl |
| | https://<eics_external_load_balancer_address>/FulfillmentOrderReversePickBean/FulfillmentOrderReversePickService?wsdl |
| | https://<eics_external_load_balancer_address>/InventoryAdjustmentBean/InventoryAdjustmentService?wsdl |
| | https://<eics_external_load_balancer_address/ItemBasketBean/ItemBasketService?wsdl |
| | https://<eics_external_load_balancer_address>/RfidInventoryBean/RfidInventoryService?wsdl |
| | https://<eics_external_load_balancer_address>/POSTransactionBean/POSTransactionService?wsdl |
| | https://<eics_external_load_balancer_address>/ProductGroupScheduleBean/ProductGroupScheduleService?wsdl |

**Table I-3    (Cont.) EICS Web Service URLs**

| SIM-WS | URL |
| --- | --- |
| | https://<eics_external_load_balancer_address>/ProductGroupBean/ProductGroupService?wsdl |
| | https://<eics_external_load_balancer_address>/ShelfAdjustmentBean/ShelfAdjustmentService?wsdl |
| | https://<eics_external_load_balancer_address>/ShelfReplenishmentBean/ShelfReplenishmentService?wsdl |
| | https://<eics_external_load_balancer_address>/StockCountBean/StockCountService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreFulfillmentOrderBean/StoreFulfillmentOrderService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreInventoryBean/StoreInventoryService?wdl |
| | https://<eics_external_load_balancer_address>/StoreInventoryUinBean/StoreInventoryUinService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreItemPriceBean/StoreItemPriceService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreBean/StoreService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreShipmentManifestBean/StoreShipmentManifestService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreShipmentReasonBean/StoreShipmentReasonService?wsdl |
| | https://<eics_external_load_balancer_address>/StoreTransferBean/StoreTransferService?wsdl |
| | https://<eics_external_load_balancer_address>/TransferDeliveryBean/TransferDeliveryService?wsdl |
| | https://<eics_external_load_balancer_address>/TransferShipmentBean/TransferShipmentService?wsdl |
| | https://<eics_external_load_balancer_address>/VendorDeliveryBean/VendorDeliveryService?wsdl |
| | https://<eics_external_load_balancer_address>/VendorReturnBean/VendorReturnService?wsdl |
| | https://<eics_external_load_balancer_address>/VendorShipmentBean/VendorShipmentService?wsdl |

# BDI Related URLs

**Table I-4    BDI Related URLs**

| | URL |
| --- | --- |
| BDI SIM Job Admin | https://<eics_external_load_balanced_address>/bdi-sim-batch-job-admin |
| BDI Work Flow | https://<rics_external_load_balanced_address>/bdi-process-flow |
| BDI Scheduler | https://<rics_external_load_balanced_address>/bdi-scheduler |

# EICS-RICS Integration URLs

## EICS-RICS Message Publishing

**Publisher Web Service URL**

The message publisher service (SIM -> RIB) is hosted by RIB-SIM. EICS installation takes in the message publisher service WSDL URLs and updates the DB system configuration table.

The WSDL URL can also be updated via EICS System Configuration UI post install.

**Table I-5    RIB Message Publishing**

| Database CONFIG_SYSTEM Name | System Configuration UI Display Name |
| --- | --- |
| integration.rib.publisher.wsdl.url | Integration Publisher Web Service URL |

**Web Service User Management**

Oracle RICS RIB manages the RIB publisher user.

The publisher user (for example ribadmin) is created as part of the RIB/RTG install.

EICS chef scripts takes the input to add to EICS credential stores.

**EICS Install Properties**

- input.sim.integration.rib.user.alias=rib-user

- This property is for SIM calling the ApplicationMessagePublishingService. The username and password for the alias should match the user which is created as part of RIB-SIM and belongs to ribAdminGroup.

- The alias name should be unique within the domain, and should not conflict with database data source user alias (for example, in some environment, the SIMRIBAlias is used for db user sim01_RIB, if that is case, choose a different alias for sim -> rib publishing user)

## Setting Up RIB Message Injector

**Injector Web Service URL**

RIB Message Injector Service is hosted in EICS application server.

The WSDL URL is:

https://<sim_external_load_balanced_addresss>/ ApplicationMessageInjector-Bean/ InjectorService?WSDL

**Web Service User Management**

The injector user (for example, sim_int) must belong to integration_users IDCS or OCI IAM Application Role, the injector user needs to be created as part of EICS provisioning process.

**EICS Installation Properties**

- input.sim.integration.rib.inject.user.alias

- This property is used for RIB-SIM -> SIM (RIB-SIM calling SIM's injector service), the user is hosted in SIM app and assign integration role (integration_users IDCS or OCI IAM Application Role), the user is created as part of SIM install (for example, sim_int user)

  See *Oracle® Retail Integration Bus Implementation Guide* Provisioning InjectorSrvice URL

# BI Related URL

**Table I-6    BI Related URL**

|  | URL |
|---|---|
| xmlpserver | https://<eics_external_load_balanced_address>/xmlpserver |

# ORDS (Apex Data Viewer) URL

**Table I-7    ORDS (Apex Data Viewer) URL**

|  | URL |
|---|---|
| ORDS (Apex Data Viewer) | https://<eics_external_load_balanced_address>/ords |

# J
# Appendix: Data Seeding Process Flows

- **Data Seeding Process Flows from Merchandising to EICS**
- **Data Seeding Mapping Tables between SIOCS and BDI**

## Data Seeding Process Flows from Merchandising to EICS

The following list shows the Data Seeding Process Flows from Merchandising to EICS with Groups.

**Table J-1    Data Seeding Process Flow Dependencies List**

| Group# | Process Flow | Notes |
|---|---|---|
| 1 | MerchHier_Fnd_ProcessFlow_From_RMS | Merchandise Hierarchy foundation data seed from Merchandising Foundation Data System. Run once for initial full data seed. |
| 1 | Supplier_Fnd_ProcessFlow_From_RMS | Supplier foundation data seed from Merchandising Foundation Data System. Run once for initial full data seed. |
| 1 | SupplierAddr_Fnd_ProcessFlow_From_RMS | Supplier address foundation data seed from Merchandising Foundation Data System. |
| 1 | Partner_Fnd_ProcessFlow_From_RMS | Partner foundation data seed from Merchandising Foundation Data System. |
| 1 | PartnerAddr_Fnd_ProcessFlow_From_RMS | Partner address foundation data seed from Merchandising Foundation Data System. |
| 1 | Wh_Fnd_ProcessFlow_From_RMS | Run once for initial full warehouse foundation data seed from Merchandising Foundation Data System. |
| 1 | WhAddr_ProcessFlow_From_RMS | Run once for initial full warehouse address foundation data seed from Merchandising Foundation Data System. |
| 2 | PartOrgUnit_Fnd_ProcessFlow_From_RMS | Run once for initial full partner Org Unit foundation data seed from Merchandising Foundation Data System. |
| 2 | ItemHdr_Fnd_ProcessFlow_From_RMS | Run once for initial full item foundation data seed from Merchandising Foundation Data System. |
| 2 | Uda_Fnd_ProcessFlow_From_RMS | Run once for initial full User Defined Attribute (UDA) foundation data seed from Merchandising Foundation Data System. |
| 2 | DiffGrp_Fnd_ProcessFlow_From_RMS | Run once for initial full foundation data seed from Merchandising Foundation Data System. |

**Table J-1    (Cont.) Data Seeding Process Flow Dependencies List**

| Group# | Process Flow | Notes |
|---|---|---|
| 2 | CodeHead_Fnd_ProcessFlow_From_RMS | Run once for initial full Code Head foundation data seed from Merchandising Foundation Data System. |
| 3 | PckitemBrkout_Fnd_ProcessFlow_From_RMS | Run once for initial full Pack Item Breakout foundation data seed from Merchandising Foundation Data System. |
| 3 | RelatedItem_Fnd_ProcessFlow_From_RMS | Run once for initial full Related Item foundation data seed from Merchandising Foundation Data System. |
| 4 | ItemSupplier_Fnd_ProcessFlow_From_RMS | Run once for initial full Item Supplier foundation data seed from Merchandising Foundation Data System. |
| 4 | ItSupCtryDim_Fnd_ProcessFlow_From_RMS | Run once for initial full Item Supplier Country Dimension foundation data seed from Merchandising Foundation Data System. |
| 4 | ItSupCtry_Fnd_ProcessFlow_From_RMS | Run once for initial full Item Supplier Country foundation data seed from Merchandising Foundation Data System. |
| 4 | ItSupCtryManCtry_Fnd_ProcessFlow_From_RMS | Run once for initial full Item Supplier Manufacturing Country foundation data seed from Merchandising Foundation Data System. |
| 4 | ItemSuppUom_Fnd_ProcessFlow_From_RMS | Run once for initial full Item Supplier Unit of Measure foundation data seed from Merchandising Foundation Data System. |
| 5 | Store_Fnd_ProcessFlow_From_RMS | Run once for initial full store foundation data seed from Merchandising Foundation Data System. |
| 5 | StoreAddr_Fnd_ProcessFlow_From_RMS | Run once for initial full store address foundation data seed from Merchandising Foundation Data System. |
| 6 | ItemLoc_Fnd_ProcessFlow_From_RMS | Run once for initial full item location foundation data seed from Merchandising Foundation Data System. |
| 6 | InvAvailStore_Tx_ProcessFlow_From_RMS | Run once for initial full store available inventory foundation data seed from Merchandising Foundation Data System. |
| 7 | ReplItemLoc_Fnd_ProcessFlow_From_RMS | Run once for initial full replenishment item location inventory foundation data seed from Merchandising Foundation Data System. |
| 7 | PriceHist_Fnd_ProcessFlow_From_RMS | Run once for initial full item pricing history foundation data seed from Merchandising Foundation Data System. |
| 8 | ItemImage_Fnd_ProcessFlow_From_RMS | Run once for initial full item image foundation data seed from Merchandising Foundation Data System. |

**Table J-1    (Cont.) Data Seeding Process Flow Dependencies List**

| Group# | Process Flow | Notes |
|---|---|---|
| 8 | UomClass_Fnd_ProcessFlow_From_RMS | Run once for initial full Unit Of Measure foundation data seed from Merchandising Foundation Data System. |
| 8 | UdaItemDate_Fnd_ProcessFlow_From_RMS | Run once for initial full UDA Item Date foundation data seed from Merchandising Foundation Data System. |
| 8 | UdaItemFF_Fnd_ProcessFlow_From_RMS | Run once for initial full UDA Item Free Form foundation data seed from Merchandising Foundation Data System. |
| 8 | UdaItemLov_Fnd_ProcessFlow_From_RMS | Run once for initial full UDA Item LOV foundation data seed from Merchandising Foundation Data System. |
| 8 | UdaValues_Fnd_ProcessFlow_From_RMS | Run once for initial full UDA Values foundation data seed from Merchandising Foundation Data System. |
| 9 | CodeDetail_Fnd_ProcessFlow_From_RMS | Run once for initial full Code Detail foundation data seed from Merchandising Foundation Data System. |
| 9 | DeliverySlot_Fnd_ProcessFlow_From_RMS | Run once for initial full Delivery Slot foundation data seed from Merchandising Foundation Data System. |
| 9 | Diff_Fnd_ProcessFlow_From_RMS | Run once for initial full Differentiators foundation data seed from Merchandising Foundation Data System. |
| 9 | UomConversion_Fnd_ProcessFlow_From_RMS | Run once for initial full UOM Conversion foundation data seed from Merchandising Foundation Data System. |

# Data Seeding Mapping Tables between SIOCS and BDI

The table below specifies the Data Seeding Mapping tables Between SIOCS and BDI.

**Table J-2    Data Seeding Mapping Tables between SIOCS and BDI**

| SIOCS Table | BDI Interface Table |
|---|---|
| ADDRESS | STORE_ADDR_IN |
|  | PARTNER_ADDR_IN |
|  | WH_ADDR_IN |
|  | SUPPLIER_ADDR_IN |
| DIFFERENTIATOR | DIFF_IN |
| DIFFERENTIATOR_TYPE | DIFF_IN |
| ITEM | ITEM_HDR_IN |
| ITEM_COMPONENT | PACKITEM_BREAKOUT_IN |
| ITEM_HIERARCHY | MERCH_HIER_IN |
| ITEM_IMAGE | ITEM_IMAGE_IN |

**Table J-2    (Cont.) Data Seeding Mapping Tables between SIOCS and BDI**

| SIOCS Table | BDI Interface Table |
| --- | --- |
| ITEM_PRICE | ITEM_LOC_IN<br>(LOC_TYPE = 'S') |
| ITEM_PRICE_HISTORY | PRICE_HIST_IN |
| ITEM_UDA | UDA_ITEM_LOV_IN<br>UDA_ITEM_FF_IN<br>UDA_ITEM_DATE_IN |
| PARTNER | PARTNER_IN |
| PARTNER_ITEM | ITEM_LOC_IN<br>(LOC_TYPE = 'E') |
| RELATED_ITEM | RELATED_ITEM_IN |
| RELATED_ITEM_TYPE | RELATED_ITEM_TYPE |
| STORE | STORE_IN |
| STORE_ITEM | ITEM_LOC_IN<br>(LOC_TYPE = 'S')<br>REPL_ITEM_LOC_IN |
| STORE_ITEM_STOCK | INV_AVAIL_STORE_IN |
| STORE_TRANSFER_ZONE | STORE_IN |
| SUPPLIER | SUPS_IN |
| SUPPLIER_ITEM | ITEM_SUPPLIER_IN |
| SUPPLIER_ITEM_COUNTRY | ITEM_SUPP_COUNTRY_IN |
| SUPPLIER_ITEM_COUNTRY_DIM | ITEM_SUP_CTY_DIM_IN |
| SUPPLIER_ITEM_MANUFACTURE | ITEM_SUP_MAN_CTY_IN |
| SUPPLIER_ITEM_UOM | ITEM_SUPP_UOM_IN |
| SUPPLIER_ORGANIZATION | PARTNER_ORG_UNIT_IN |
| UDA | UDA_IN |
| UDA_LOV | UDA_VALUES_IN |
| UOM_CLASS | UOM_CLASS_IN |
| UOM_CONVERSION | UOM_CONVERSION_IN |
| WAREHOUSE | WH_IN |
| WAREHOUSE_ITEM | ITEM_LOC_IN<br>(LOC_TYPE = 'W') |

# K

# Appendix: Pricing Import with On-premise RPM

This section is only applicable for EICS to integration with customer have on-premise RPM, RPM pricing files are placed on EICS SFTP incoming locations, and the file import job processes each file from incoming directory when a corresponding completer file presents. The completer is a zero-byte file which indicates file transfer is complete and ready for processing. The reject files will be pushed to outgoing for client review.

> **Note:**
>
> Pricing file import from On-premise RPM will be phased out in future releases.

- Clearance Price Change Import with On-premise RPM
- Promotion Price Change Import with On-premise RPM
- Regular Price Change Import with On-premise RPM

## Clearance Price Change Import with On-premise RPM

This batch imports the clearance price changes from flat file into item price table. Each file contains price change record for a single location. Files with location type of store (file name with suffix of "S") are relevant to EICS. The import process skips file with location type which are not store.

There are two phases involve in the batch process. The file load phase loads the file into price change worksheet table; the extract phase kicks off multiple threads to extract the approved worksheet records into item price table.

To optimize price change batch import process, admin user can change the system configuration parameter 'DAYS_TO_HOLD_PRICE_CHANGE_WORKSHEET' value

to 0, the completed staged worksheet records will be deleted after price change records are extracted into item_price table.

**File Specification**

**File Name Format**: CLRPC _<date in YYYYMMDDHH24MISS format>_<loc type>

Where <loc type> can be S (store) or W (Warehouse), EICS only process files which have loc_type of S.

**File Layout**: See Clearance Price Change Import File Layout Specification.

**Batch Job Definition Name**

ClearancePriceChange_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if the file staging fails. The staging process is all or none transaction so if an error occurs during the batch process, none of the transactions in the file will be staged. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table K-1    Key Tables for Clearance Price Change with On-premise RPM**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| batch_execution | Yes | Yes | Yes | |
| price_change_worksheet | Yes | Yes | Yes | Yes |
| Item_price | Yes | Yes | Yes | |

# Promotion Price Change Import with On-premise RPM

This batch imports the promotion price changes from flat file into item price table.

There are two phases involve in the batch process. The file load phase loads the file into price change worksheet table; the extract phase kicks off multiple threads to extract the approved worksheet records into item price table.

To optimize price change batch import process, admin user can change the system configuration parameter 'DAYS_TO_HOLD_PRICE_CHANGE_WORKSHEET' value

to 0, the completed staged worksheet records will be deleted after price change records are extracted into item_price table.

**File Specification**

**File Name format:** PRMPC_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>

Where <loc type> can be S (store) or W (Warehouse), EICS only process files which have loc_type of S.

**File Layout**: See Promotion Price Change Import File Layout Specification.

**Batch Job Definition Name**

PromotionPriceChange_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if the file staging fails. The staging process is all or none transaction so if an error occurs during the batch process, none of the transactions in the file will be staged. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table K-2    Key Tables for Promotion Price Change with On-premise RPM**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| price_change_worksheet | Yes | | Yes | Yes |
| Item_price | Yes | Yes | Yes | |
| config_store | Yes | | | |
| item_uda | Yes | | | |
| print_format_type | Yes | | | |

# Regular Price Change Import with On-premise RPM

This batch imports the regular price changes from flat file into item price table.

There are two phases involve in the batch process. The file load phase loads the file into price change worksheet table; the extract phase kicks off multiple threads to extract the approved worksheet records into item price table.

To optimize price change batch import process, admin user can change the system configuration parameter 'DAYS_TO_HOLD_PRICE_CHANGE_WORKSHEET' value

to 0, the completed staged worksheet records will be deleted after price change records are extracted into item_price table.

**File Specification**

**File Name format**: REGPC_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>

Where <loc type> can be S (store) or W (Warehouse), EICS only process files which have loc_type of S.

**File Layout**: See Regular Price Change Import File Layout Specification.

**Batch Job Definition Name**

RegularPriceChange_OpsJob

**Batch Job Parameters**

<File name>

If not specified, then the data file in incoming directory are processed.

**File Error Handling**

The batch job will be marked as failed if the file staging fails. The staging process is all or none transaction so if an error occurs during the batch process, none of the transactions in

the file will be staged. The user will need to rerun the same file again after resolving any errors.

**Key Tables**

**Table K-3    Key Tables for Regular Price Change with On-premise RPM**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| price_change_worksheet | Yes | Yes | Yes | Yes |
| Item_price | Yes | Yes | Yes | |
| config_store | Yes | | | |

**Table K-4    Key Tables for Regular Price Change Batch**

| Tables | Select | Insert | Update | Delete |
|---|---|---|---|---|
| item_uda | Yes | | | |
| print_format_type | Yes | | | |
| batch_execution | Yes | Yes | Yes | |

# L
# Appendix: Open Transaction Seeding via BDI

## Overview

Merchandising Solution usually goes live prior to the SIOCS system. The integration layer supports the flow of foundation data via BDI (bulk data import) process or via RIB Messages. As the customer is currently in the transition phase that is, cutover from old system to Oracle Cloud Solution, the complete transaction freeze for the stores may not be achievable. There would be a few transactions which would be currently in In-Progress states which means they are either not fully received, closed or cancelled. This data needs to be imported into SIOCS as these are open transactions.

Third Party Open TX data imports open transaction data for three main areas.

1. Purchase Order

2. Direct Store Deliveries from Supplier

3. Transfer / Allocations

Third party transaction import is considered to be a one-time activity before go live and not a practice for transactional data flow. The system would allow creating new transactions and updating existing transactions only if the sanity of the transaction that is, the current state in SIOCS database is the same.

As the transaction import is from an external system, the identifier generated for the transactions on the external system will also be mapped to the external identifier of SIOCS transactional data.

The complete integration of data imports direct from Merchandising similar to foundation data import is currently not supported.

The implementation team would be responsible for sourcing this data via BDI External Interfaces.

The following Transaction Import batches are independent of each other and would be responsible for processing the transactions to SIOCS database.

## BDI Implementation Considerations

To load third party data into EICS via BDI, external system needs to provide data sourcing implementations.

This section lists two approaches for data sourcing implementation.

## Option A: BDI External Process Flow (via External Edge App)

With this approach, the BDI Ext Process Flow will move the data from sourcing system to destination system's application table. BDI process flow provides BDI job scheduling, end to

end data transportation from sourcing system's database table into destination application's tables.

For additional details, see RICS documentations:

*Oracle® Retail Bulk Data Integration Cloud Service Implementation Guide*

*Oracle® Retail Enterprise Integration Overview Guide*

## Option B: BDI CLI Transmitter

BDI CLI File Transmitter is a client command line tool, is applicable for smaller data file size which does not require concurrent processing.

BDI CLI file transmitter only transports data files from sourcing systems to destination system's interface tables. It does not provide end to end data transportation. If you choose BDI CLI File transmitter for loading third party data into EICS application tables, it will require three steps:

1.  Download BDI CLI File Transmitter from RICS' and install on customer's system.

2.  Run BDI CLI Transmitter batch to load data file from sourcing system into receiver system's interface table (*_IN tables).

3.  Once the data is loaded into the receiver app's inbound interface tables, the user would need to provide the data set id and start EICS's importer job to move the data from the inbound interface table into the EICS app tables.

# Open Transaction Data Processing

The following section details a few key points for the technical design of transaction data import:

1.  **Flattening the database tables**

    Flattened database architecture makes the query execution much faster than the normalized form. Each row on the table would denote a line item with complete information of the transaction it belongs to and what line item attribute values are.

    This reduces the overhead cost of join and querying multiple tables to get the complete transaction detail identified by a particular identifier.

2.  **Data Validation**

    The first and foremost step of the transaction import is validating the data against the values available. The transaction import batch processes would validate the complete data as the data would then be processed to the SIOCS transaction tables. The basic validations which would be performed are as follows:

    a.  Is the transaction for a valid store?

    b.  Is the transaction from a valid source? (store, warehouse, finisher or supplier)

    c.  Is the item identifier existing on the item master table of SIOCS database?

    d.  Is the status / type specified on data denoting valid information as per the SIOCS technical design?

    Please refer to Data Import Exception Handling section for more details.

3.  **Parallel Processing**

The batch process will split the data into partitions by the transaction external identifiers to support the parallel processing of the transaction import. Each partition will contain the list of transaction identifiers which are to be processed. The number of partitions in which the data would be divided is defined by an existing system admin configuration parameter (Batch Maximum Threads). The default value for this parameter is set to 5. Each thread would be assigned a partition to be processed.

So, if there are 10000 transactions to be imported, 5 separate thread processes would be instantiated to process a workload of 2000 transactions per thread.

The partition feature used here is as per JSR 352 specification defined for Java EE 7 and Java SE 6 platforms.

4. **Data Import**

   The partitioned transaction external identifier will be validated against the existing transactions on the SIOCS database (if any exists). If the no matching transactions are found, a new transaction would be added to the respective transaction table. If the transaction identifier matches to an existing record, then that transaction would be updated with the latest data from the flattened staging table. The data import process does not update the state of transaction.

   The data import processes the data into chunks and a failover mechanism is added to process the correct data and mark the invalid data as failed. This also helps improving the server performance as not all transactions on a partition are imported at one go. The chunk process will further break the partition in sub-lists and process the data in group of records reducing the server overhead and memory consumption.

5. **Inventory Correction**

   The inventory correction logic is not applicable for purchase order processing as there is no inventory bucket to be updated.

   For the transfer transactions that are currently approved, the transfer reserved bucket of the item's inventory at source store needs to be corrected once the transaction is successfully imported into the system.

   For the Direct Store Delivery transactions (if associated with customer order), the picking quantity would need to be updated when the received delivery is created depending on a system parameter.

   Below is the configuration parameter to be considered while processing the direct store delivery which has a container in Received status.

# Direct Delivery - Import Details

This batch imports the direct delivery transaction imported from Merchandising / Third Party System posted on SIOCS BDI table for example, EXT_DSD_IN. The job supports only creation of the deliveries of ASN types. Data validation is also part of processing and proper error mechanism will log the error against each batch execution. This job does not support the importing of UINs and Custom Attributes.

For the interface definition, please refer to the DSD Interface Specification section.

**Table L-1    EXT_DSD_IN and SIOCS Transaction Table Mappings**

| BDI Interface Table | SIOCS Transactional Tables |
| --- | --- |
| EXT_DSD_IN | DSD |

**Table L-1    (Cont.) EXT_DSD_IN and SIOCS Transaction Table Mappings**

| BDI Interface Table | SIOCS Transactional Tables |
| --- | --- |
| | DSD_CARTON |
| | DSD_LINE_ITEM |
| | DSD_ADJUSTMENT |

For the detail column level mapping please refer to Data Mapping section.

**Table L-2    Configuration Parameters for DSD Received Status**

| Parameter Name | Description |
| --- | --- |
| Reserve Customer Order Inventory Upon Receiving | Update the customer order reserved bucket while creating a new transaction. |
| Auto Pick On Receive - Direct Delivery | Will update the picked quantity on customer order (if exists). |
| Auto Pick Mixed Containers | Will update the picked quantity on customer order (if exist) for a container which contains both customer order related items and normal items. |

# Transfer - Import Details

This batch imports the transfer and allocation records from Merchandising / Third Party System posted on SIOCS BDI table that is, EXT_TSF_IN. The job support creates the transfer / allocation records in SIOCS database and makes them accessible on UI for users. The inventory in-transit calculation posts the importing of the transaction and will be taken care of by this batch job. Data validation is also part of processing and proper error mechanism will log the error against each batch execution.

For the interface definition, please refer to the Transfer Interface Specification section.

**Table L-3    EXT_TSF_IN and SIOCS Transaction Table Mappings**

| BDI Interface Table | SIOCS Transactional Tables |
| --- | --- |
| EXT_TSF_IN | TSF |
| | TSF_LINE_ITEM |
| | TSF_ALLOCATION |

# Purchase Order - Import Details

This batch imports the purchase order records from Merchandising / Third Party System posted on SIOCS BDI table that is, EXT_PURCH_ORDER. This job creates the purchase order records in SIOCS database and makes them accessible on UI for users. Data validation is also part of processing and proper error mechanism will log the error against each batch execution.

For the interface definition, please refer to the Purchase Order Interface Specification section.

**Table L-4    EXT_PURCH_ORDER_IN BDI And App Table Mappings**

| BDI Interface Table | SIOCS Transactional Tables |
| --- | --- |
| EXT_PURCH_ORDER_IN | PURCHASE_ORDER |
| | PURCHASE_ORDER_LINE_ITEM |

For the detail column level mapping please refer to the Data Mapping section.

## Data Mapping

The table below specifies the database column mapping for each functional area.

1. **Direct Store Delivery from Supplier (DSD Receiving)**

   BDI Transaction Import Table: EXT_DSD_IN

   | BDI Column | Transaction Mapped Column | Transaction Mapped Table |
   | --- | --- | --- |
   | DSD_ASN | ASN_ID | DSD |
   | DSD_STORE | STORE_ID | DSD |
   | DSD_SUPPLIER | SUPPLIER_ID | DSD |
   | DSD_STATUS | STATUS | DSD |
   | DSD_EXPECTED_DATE | EXPECTED_DATE | DSD |
   | DSD_CREATE_DATE | CREATE_DATE | DSD |
   | DSD_UPDATE_DATE | UPDATE_DATE | DSD |
   | DSD_RECEIVED_DATE | RECEIVED_DATE | DSD |
   | DSD_INVOICE | INVOICE_ID | DSD |
   | DSD_INVOICE_DATE | INVOICE_DATE | DSD |
   | DSD_CREATE_USER | CREATE_USER | DSD |
   | DSD_CUST_ORD_ID | CUST_ORDER_ID | DSD |
   | DSD_FUL_ORD_EXT_ID | FUL_ORD_EXTERNAL_ID | DSD |
   | DSD_CARRIER_ENTITY | CARRIER_ENTITY | DSD |
   | DSD_CARRIER_TYPE | CARRIER_TYPE | DSD |
   | DSD_CARRIER_CODE | CARRIER_CODE | DSD |
   | DSD_COUNTRY_CODE | COUNTRY_CODE | DSD |
   | DSD_SOURCE_ADDRESS | SOURCE_ADDRESS | DSD |
   | DSD_LICENSE_PLATE | LICENSE_PLATE | DSD |
   | DSD_FREIGHT_ID | FREIGHT_ID | DSD |
   | DSD_CURRENCY_CODE | CURRENCY_CODE | DSD |
   | DSD_INVOICE_COST | INVOICE_COST_VALUE | DSD |
   | DSD_RECEIVED_USER | RECEIVED_USER | DSD |
   | DSD_UPDATE_USER | UPDATE_USER | DSD |
   | DSD_BOL_EXT_ID | BOL_EXTERNAL_ID | DSD |
   | DSD_ADJ_PO_EXT_ID | PURCHASE_ORDER_EXT_ID | DSD_ADJUSTMENT |
   | DSD_ADJ_ITEM | ITEM_ID | DSD_ADJUSTMENT |

| BDI Column | Transaction Mapped Column | Transaction Mapped Table |
|---|---|---|
| DSD_ADJ_DATE | ADJUSTMENT_DATE | DSD_ADJUSTMENT |
| DSD_ADJ_QUANTITY | ADJUSTMENT_QUANTITY | DSD_ADJUSTMENT |
| DSD_CARTON_EXT_ID | EXTERNAL_ID | DSD_CARTON |
| DSD_CARTON_STATUS | STATUS | DSD_CARTON |
| DSD_CARTON_STORE | STORE_ID | DSD_CARTON |
| DSD_CARTON_REFERENCE_ID | REFERENCE_ID | DSD_CARTON |
| DSD_CARTON_DAMAGED_REASON | DAMAGED_REASON | DSD_CARTON |
| DSD_CARTON_SERIAL_CODE | SERIAL_CODE | DSD_CARTON |
| DSD_CARTON_TRACKING_NUMBER | TRACKING_NUMBER | DSD_CARTON |
| DSD_CARTON_DAMAGE_REMAINING | DAMAGE_REMAINING | DSD_CARTON |
| DSD_CARTON_CREATE_USER | CREATE_USER | DSD_CARTON |
| DSD_CARTON_UPDATE_USER | UPDATE_USER | DSD_CARTON |
| DSD_CARTON_RECEIVE_USER | RECEIVE_USER | DSD_CARTON |
| DSD_CARTON_CREATE_DATE | CREATE_DATE | DSD_CARTON |
| DSD_CARTON_UPDATE_DATE | UPDATE_DATE | DSD_CARTON |
| DSD_CARTON_RECEIVE_DATE | RECEIVE_DATE | DSD_CARTON |
| DSD_CARTON_UIN_REQUIRED | UIN_REQUIRED | DSD_CARTON |
| DSD_CARTON_RCV_AT_SHOP_FLOOR | RECEIVE_AT_SHOP_FLOOR | DSD_CARTON |
| DSD_CARTON_QUALITY_CONTROL | QUALITY_CONTROL | DSD_CARTON |
| DSD_CARTON_EXTERNAL_CREATE | EXTERNAL_CREATE | DSD_CARTON |
| DSD_CARTON_ADJUSTED | ADJUSTED | DSD_CARTON |
| DSD_CARTON_CUST_ORD_RELATED | CUSTOMER_ORDER_RELATED | DSD_CARTON |
| DSD_LINE_ITEM | ITEM_ID | DSD_LINE_ITEM |
| DSD_LINE_CASE_SIZE | CASE_SIZE | DSD_LINE_ITEM |
| DSD_LINE_EXPECTED | QUANTITY_EXPECTED | DSD_LINE_ITEM |
| DSD_LINE_RECEIVED | QUANTITY_RECEIVED | DSD_LINE_ITEM |
| DSD_LINE_DAMAGED | QUANTITY_DAMAGED | DSD_LINE_ITEM |
| DSD_LINE_UNIT_COST_CURR | UNIT_COST_CURRENCY | DSD_LINE_ITEM |
| DSD_LINE_UNIT_COST_VAL | UNIT_COST_VALUE | DSD_LINE_ITEM |
| DSD_LINE_OVRD_UNIT_COST_CURR | OVERRIDE_UNIT_COST_CURRENCY | DSD_LINE_ITEM |
| DSD_LINE_OVRD_UNIT_COST_VAL | OVERRIDE_UNIT_COST_VALUE | DSD_LINE_ITEM |
| DSD_LINE_STORE | STORE_ID | DSD_LINE_ITEM |
| DSD_LINE_PO_EXT_ID | PURCHASE_ORDER_EXT_ID | DSD_LINE_ITEM |
| DSD_LINE_CUST_ORD_ID | CUST_ORDER_ID | DSD_LINE_ITEM |

| BDI Column | Transaction Mapped Column | Transaction Mapped Table |
|---|---|---|
| DSD_LINE_FUL_ORD_EXT_ID | FUL_ORD_EXTERNAL_ID | DSD_LINE_ITEM |
| DSD_LINE_VPN | VENDOR_PRODUCT_NUMBER | DSD_LINE_ITEM |
| DSD_LINE_RECEIVED_OVER | QUANTITY_RECEIVED_OVER | DSD_LINE_ITEM |
| DSD_LINE_DAMAGED_OVER | QUANTITY_DAMAGED_OVER | DSD_LINE_ITEM |
| DSD_LINE_PREV_RECEIVED | PREVIOUS_RECEIVED | DSD_LINE_ITEM |
| DSD_LINE_PREV_DAMAGED | PREVIOUS_DAMAGED | DSD_LINE_ITEM |

2. **Transfers**

BDI Transaction Import Table: EXT_TSF_IN

Column TSF_DOC_TYPE = T denotes the transaction is transfer.

| BDI Column | Mapped Column | Mapped Column Table |
|---|---|---|
| TSF_EXT_ID | EXTERNAL_ID | TSF |
| TSF_SRC_TYPE | SOURCE_TYPE | TSF |
| TSF_SRC | SOURCE_ID | TSF |
| TSF_DEST_TYPE | DESTINATION_TYPE | TSF |
| TSF_DEST | DESTINATION_ID | TSF |
| TSF_STATUS | STATUS | TSF |
| TSF_CREATE_USER | CREATE_USER | TSF |
| TSF_UPDATE_USER | UPDATE_USER | TSF |
| TSF_APPROVAL_USER | APPROVAL_USER | TSF |
| TSF_CREATE_DATE | CREATE_DATE | TSF |
| TSF_UPDATE_DATE | UPDATE_DATE | TSF |
| TSF_APPROVAL_DATE | APPROVAL_DATE | TSF |
| TSF_CONTEXT_ID | CONTEXT_ID | TSF |
| TSF_CONTEXT_VALUE | CONTEXT_VALUE | TSF |
| TSF_CUST_ORD_ID | CUST_ORD_EXTERNAL_ID | TSF |
| TSF_FUL_ORD_EXT_ID | FUL_ORD_EXTERNAL_ID | TSF |
| TSF_PARTIAL_DELIVERY_IND | ALLOW_PARTIAL_DELIVERY | TSF |
| TSF_AUTH_CODE | AUTHORIZATION_CODE | TSF |
| TSF_NOT_AFTER_DATE | NOT_AFTER_DATE | TSF |
| TSF_REQUEST_DATE | REQUEST_DATE | TSF |
| TSF_REQUEST_USER | REQUEST_USER | TSF |
| TSF_USE_AVAILABLE | USE_AVAILABLE | TSF |
| TSF_LINE_ITEM | ITEM_ID | TSF_LINE_ITEM |
| TSF_LINE_CASE_SIZE | CASE_SIZE | TSF_LINE_ITEM |
| TSF_LINE_REQUESTED | QUANTITY_REQUESTED | TSF_LINE_ITEM |
| TSF_LINE_APPROVED | QUANTITY_APPROVED | TSF_LINE_ITEM |
| TSF_LINE_SHIPPING | QUANTITY_SHIPPING | TSF_LINE_ITEM |

| BDI Column | Mapped Column | Mapped Column Table |
|---|---|---|
| TSF_LINE_SHIPPED | QUANTITY_SHIPPED | TSF_LINE_ITEM |
| TSF_LINE_RECEIVED | QUANTITY_RECEIVED | TSF_LINE_ITEM |
| TSF_LINE_DAMAGED | QUANTITY_DAMAGED | TSF_LINE_ITEM |
| TSF_LINE_PREFERRED_UOM | PREFERRED_UOM | TSF_LINE_ITEM |

3. **Transfer Allocations**

BDI Transaction Import Table: EXT_TSF_IN

Column TSF_DOC_TYPE = A denotes the transaction is transfer allocation.

| BDI Column | Mapped Column | Mapped Column Table |
|---|---|---|
| TSF_EXT_ID | EXTERNAL_ID | TSF_ALLOCATION |
| TSF_DELIVERY_DATE | DELIVERY_DATE | TSF_ALLOCATION |
| TSF_LINE_ITEM | ITEM_ID | TSF_ALLOCATION |
| TSF_DEST | STORE_ID | TSF_ALLOCATION |
| TSF_DELV_SLOT_ID | DELIVERY_SLOT_ID | TSF_ALLOCATION |
| TSF_SRC | SOURCE_ID | TSF_ALLOCATION |
| TSF_SRC_TYPE | SOURCE_TYPE | TSF_ALLOCATION |
| TSF_CONTEXT_VALUE | CONTEXT_VALUE | TSF_ALLOCATION |
| TSF_LINE_DIST_PARENT_ID | DISTRIBUTION_PARENT_ID | TSF_ALLOCATION |
| TSF_STATUS | STATUS | TSF_ALLOCATION |
| TSF_LINE_EXPECTED | QUANTITY_EXPECTED | TSF_ALLOCATION |
| TSF_LINE_RECEIVED | QUANTITY_RECEIVED | TSF_ALLOCATION |
| TSF_LINE_DAMAGED | QUANTITY_DAMAGED | TSF_ALLOCATION |
| TSF_CONTEXT_ID | CONTEXT_ID | TSF_ALLOCATION |

4. **Purchase Order**

BDI Transaction Import Table: EXT_PURCH_ORDER_IN

| BDI Column | Mapped Column | Mapped Column Table |
|---|---|---|
| PO_EXT_ID | EXTERNAL_ID | PURCHASE_ORDER |
| PO_STORE | STORE_ID | PURCHASE_ORDER |
| PO_SUPPLIER | SUPPLIER_ID | PURCHASE_ORDER |
| PO_STATUS | STATUS | PURCHASE_ORDER |
| PO_NOT_BEFORE_DATE | NOT_BEFORE_DATE | PURCHASE_ORDER |
| PO_NOT_AFTER_DATE | NOT_AFTER_DATE | PURCHASE_ORDER |
| PO_CREATE_DATE | CREATE_DATE | PURCHASE_ORDER |
| PO_UPDATE_DATE | UPDATE_DATE | PURCHASE_ORDER |
| PO_COMPLETE_DATE | COMPLETE_DATE | PURCHASE_ORDER |
| PO_USER | USER_ID | PURCHASE_ORDER |

| BDI Column | Mapped Column | Mapped Column Table |
|---|---|---|
| PO_COMMENTS | COMMENTS | PURCHASE_ORDER |
| PO_CUST_ORD_ID | CUST_ORDER_ID | PURCHASE_ORDER |
| PO_FUL_ORD_EXT_ID | FUL_ORD_EXTERNAL_ID | PURCHASE_ORDER |
| PO_EXT_STATUS | EXTERNAL_STATUS | PURCHASE_ORDER |
| PO_LINE_ITEM | ITEM_ID | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_SUPP_COUNTRY | SUPPLIER_COUNTRY_ID | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_CASE_SIZE | CASE_SIZE | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_EXPECTED | QUANTITY_EXPECTED | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_RECEIVED | QUANTITY_RECEIVED | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_UNIT_COST_CURRENCY | UNIT_COST_CURRENCY | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_UNIT_COST_VALUE | UNIT_COST_VALUE | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_PREFERRED_UOM | PREFERRED_UOM | PURCHASE_ORDER_LINE_ITEM |
| PO_LINE_PUBLISHED | PUBLISHED | PURCHASE_ORDER_LINE_ITEM |

# Data Import Exception Handling

This section describes the error message with which batch would be marked as failed due to the invalid data on transaction import file.

Each error message will be prefixed with following text to denote the error recorded is for which row value in respective BDI transaction import table.

Prefix Text: `Validation failed for Row Sequence: <BDI_SEQ_ID>`

**Table L-5    1. Purchase Order**

| Error | Cause | Data Column with Error |
|---|---|---|
| Purchase Order:<PO_EXT_ID> has invalid store information. | Store identifier is not a valid store for a new purchase order. | PO_STORE |
| Purchase Order:<PO_EXT_ID> has invalid supplier information. | Supplier identifier is not a valid supplier for a new purchase order. | PO_SUPPLIER |
| Purchase Order:<PO_EXT_ID> has no status information. | Status is either null or not valid value for purchase order status for a new purchase order. | PO_STATUS |
| Purchase Order:<PO_EXT_ID> has no external status information. | External status is either null or not valid value for purchase order external status for a new purchase order. | PO_EXT_STATUS |

**Table L-5    (Cont.) 1. Purchase Order**

| Error | Cause | Data Column with Error |
|---|---|---|
| Purchase Order:<PO_EXT_ID> has invalid store information that does not match to existing transaction. | Store identifier doesn't match to the existing purchase order. | PO_STORE |
| Purchase Order:<PO_EXT_ID> has invalid supplier information that does not match to existing transaction. | Supplier identifier doesn't match to the existing purchase order. | PO_SUPPLIER |
| Purchase Order:<PO_EXT_ID> has invalid status code. | Status is either null or not valid value for purchase order status for an existing purchase order. | PO_STATUS |
| Purchase Order:<PO_EXT_ID> has status which does not match the existing data. | Status is not same as that for the existing purchase order. | PO_STATUS |
| Purchase Order:<PO_EXT_ID> has external status information which is not valid for existing transaction. | External status is either null or not a valid value for an existing transaction. | PO_EXT_STATUS |
| Purchase Order:<PO_EXT_ID> Item:<ITEM_ID> is not ranged for store. | Item is not range to the store. | PO_LINE_ITEM PO_STORE |
| Purchase Order:<PO_EXT_ID> Item:<ITEM_ID> is not ranged for supplier. | Item is not ranged to the supplier. | PO_LINE_ITEM PO_SUPPLIER |
| Purchase Order:<PO_EXT_ID> Item:<ITEM_ID> has no supplier country information. | Item Supplier country information is not available on the SIOCS. | PO_LINE_ITEM PO_SUPPLIER PO_LINE_SUPP_COUNTRY |

**Table L-6    2. Transfer**

| Error | Cause | Data Column with Error |
|---|---|---|
| Transfer:<TSF_EXT_ID> has invalid source information. | Transfer source is not valid. | TSF_SRC |
| Transfer:<TSF_EXT_ID> has invalid destination information. | Transfer destination is not valid. | TSF_DEST |
| Transfer:<TSF_EXT_ID> is not a valid store transfer. | Transfer is not valid store transfer that is, both source and destination is not a store. | TSF_SRC_TYPE TSF_DEST_TYPE |
| Transfer:<TSF_EXT_ID> has invalid status information. | Transfer Status is null or not valid value. | TSF_STATUS |
| Transfer:<TSF_EXT_ID> has source identifier which does not match the existing data. | Transfer source identifier does not match to the existing transfer transaction on SIOCS. | TSF_SRC |
| Transfer:<TSF_EXT_ID> has destination identifier which does not match the existing data. | Transfer destination identifier does not match to the existing transfer transaction on SIOCS. | TSF_DEST |

**Table L-6 (Cont.) 2. Transfer**

| Error | Cause | Data Column with Error |
|---|---|---|
| Transfer:<TSF_EXT_ID> has status which does not match the existing data. | Transfer status does not match to the existing transaction state on SIOCS. | TSF_STATUS |
| Transfer:<TSF_EXT_ID> is already approved and use available flag value doesn't match existing data. | Use available flag is changed for the existing approved transfer transaction. | TSF_USE_AVAILABLE |
| Transfer:<TSF_EXT_ID> has no source type information. | Transfer source type is either null or an invalid value. | TSF_SRC_TYPE |
| Transfer:<TSF_EXT_ID> has no destination type information. | Transfer destination type is either null or an invalid value. | TSF_DEST_TYPE |
| Transfer:<TSF_EXT_ID> Item:<ITEM_ID> is not ranged for source. | Item is not ranged for the source. | TSF_LINE_ITEM<br>TSF_SRC |
| Transfer:<TSF_EXT_ID> Item:<ITEM_ID> is not ranged for destination. | Item is not ranged for the destination. | TSF_LINE_ITEM<br>TSF_DEST |

**Table L-7 3. Allocation**

| Error | Cause | Data Column with Error |
|---|---|---|
| Allocation:<TSF_ALLOCATION_ID> has source information which does not match existing data. | Allocation source is not valid for the existing transaction. | TSF_SRC |
| Allocation:<TSF_ALLOCATION_ID> has invalid source information. | Allocation source is either null or not a valid value. | TSF_SRC |
| Allocation:<TSF_ALLOCATION_ID> has invalid destination information. | Allocation destination is either null or not a valid value. | TSF_DEST |
| Allocation:<TSF_ALLOCATION_ID> Item:<ITEM_ID> is not ranged for source. | Item is not ranged to the allocation source. | TSF_LINE_ITEM<br>TSF_SRC |
| Allocation:<TSF_ALLOCATION_ID> Item:<ITEM_ID> is not ranged for destination. | Item is not ranged to the allocation destination. | TSF_LINE_ITEM<br>TSF_DEST |
| Allocation:<TSF_ALLOCATION_ID> has invalid status code. | Allocation status is not valid. | TSF_STATUS |
| Allocation:<TSF_ALLOCATION_ID> has status which does not match the existing data. | Allocation status does not match to the existing transaction status. | TSF_STATUS |
| Allocation:<TSF_ALLOCATION_ID> has no expected quantity. | Allocation quantity is null. | TSF_LINE_EXPECTED |

**Table L-8    4. Direct Store Delivery (DSD)**

| Error | Cause | Data Column with Error |
|---|---|---|
| DSD ASN is null. | DSD ASN is null. | DSD_ASN |
| DSD:<ASN> has invalid store identifier. | Store identifier is either null or not a valid store. | DSD_STORE |
| DSD:<ASN> has invalid supplier identifier. | Supplier identifier is either null or not a valid supplier. | DSD_SUPPLIER |
| DSD:<ASN> has invalid status code. | Delivery status is either null or not a valid status value. | DSD_STATUS |
| DSD:<ASN> has invalid carrier type code. | Specified Carrier Type value is not valid. | DSD_CARRIER_TYPE |
| DSD:<ASN> has store identifier which doesn't matches to existing delivery. | Store identifier does not match to the existing delivery transaction. | DSD_STORE |
| DSD:<ASN> has supplier identifier which doesn't matches to existing delivery. | Supplier identifier does not match to the existing delivery transaction. | DSD_SUPPLIER |
| DSD:<ASN> has invalid status code for existing delivery. | Status code is either null or an invalid value for an existing delivery. | DSD_STATUS |
| DSD:<ASN> has status which does not match the existing data. | Status code does not match the current status of the existing delivery transaction. | DSD_STATUS |
| DSD:<ASN> is in <DELV_STATUS> status and has one or more container in open state. | Delivery is currently in received status however few container records are in open state on import data. | DSD_STATUS DSD_CARTON_STATUS |
| DSD:<ASN> is in <DELV_STATUS> status and has one or more container in open or completed state. | Delivery is in cancel/rejected status however few of the containers on delivery are either completed or open. | DSD_STATUS DSD_CARTON_STATUS |
| DSD:<ASN> Container Identifier is null or invalid. | Container identifier is either null or an invalid value. | DSD_CARTON_EXT_ID |
| DSD Container:<CARTON_EXT_ID> has invalid status code. | Container status is either null or invalid value for status. | DSD_CARTON_STATUS |
| DSD Container:<CARTON_EXT_ID> has invalid store identifier. | Container store is either null or not a valid store. | DSD_CARTON_STORE |
| DSD Container:<CARTON_EXT_ID> has store identifier which doesn't match to delivery information. | Container store does not match the delivery store identifier. | DSD_CARTON_STORE DSD_STORE |
| DSD Container:<CARTON_EXT_ID> has store identifier which doesn't matches to existing carton. | Container store does not match to the store identifier for an existing container. | DSD_CARTON_STORE |

Appendix L
Open Transaction Data Processing

**Table L-8    (Cont.) 4. Direct Store Delivery (DSD)**

| Error | Cause | Data Column with Error |
|---|---|---|
| DSD Container:<CARTON_EXT_ID> has invalid status code. | Container status value is either null or not a valid status value. | DSD_CARTON_STATUS |
| DSD Container:<CARTON_EXT_ID> has status which does not match the existing data. | Container status does not match the status of existing container. | DSD_CARTON_STATUS |
| DSD Container:<CARTON_EXT_ID> has invalid order related code. | Order Related flag has an invalid code. | DSD_CARTON_CUST_ORD_RELATED |
| DSD Line Item:<ITEM_ID> is not ranged for store. | Item is not ranged to the store. | DSD_LINE_ITEM<br>DSD_LINE_STORE |
| DSD Line Item:<ITEM_ID> is not ranged for supplier. | Item is not ranged to the supplier. | DSD_LINE_ITEM<br>DSD_SUPPLIER |
| DSD Line Item:<ITEM_ID> has invalid store identifier. | Store identifier for line item is either null or not a valid store. | DSD_LINE_ITEM<br>DSD_LINE_STORE |
| DSD Line Item:<ITEM_ID> has store identifier which does not match to carton information. | Store identifier for line item does not match to the container store identifier. | DSD_LINE_STORE<br>DSD_CARTON_STORE |
| DSD Line Item:<ITEM_ID> has invalid purchase order identifier. | Purchase Order external identifier for line item is not valid. | DSD_LINE_PO_EXT_ID |
| DSD Line Item:<ITEM_ID> has store identifier which does not match to existing line item information. | Store identifier for line item does not match the store identifier for an existing line item. | DSD_LINE_STORE |
| DSD Adjustment Item:<ITEM_ID> is not ranged for store. | Adjustment Item is not ranged to the store. | DSD_ADJ_ITEM<br>DSD_STORE |
| DSD Adjustment Item:<ITEM_ID> is not ranged for supplier. | Adjustment Item is not ranged to the supplier. | DSD_ADJ_ITEM<br>DSD_SUPPLIER |
| DSD Adjustment Item:<ITEM_ID> has invalid purchase order external identifier. | Adjustment Item has an invalid purchase order identifier. | DSD_ADJ_PO_EXT_ID |
| DSD Adjustment Item:<ITEM_ID> has invalid adjustment date. | Adjustment data is set to null. | DSD_ADJ_DATE |
| DSD Adjustment Item:<ITEM_ID> has invalid adjustment quantity. | Adjustment quantity is not specified. | DSD_ADJ_QUANTITY |

# Additional Details for BDI CLI File Imports

This section is only applicable if you choose to use BDI CLI File Transmitter to transport the data file. You may skip this section if you choose BDI External Process Flow for data transportation.

**Transaction Interface Specifications**

# DSD Interface Specification

| Name | Type | Optional (Null?) | Description |
|------|------|------------------|-------------|
| BDI_DATASET_ACTION | VARCHAR2(20) | No | The required record action type, valid values are: REPLACE |
| DSD_ASN | VARCHAR2(128) | No | The advanced shipping notification identifier that is associated with the direct delivery. |
| DSD_STORE | NUMBER(10,0) | Yes | The unique identifier of the store that this direct delivery is for. |
| DSD_SUPPLIER | NUMBER(10,0) | Yes | The unique identifier of the supplier that shipped the items. |
| DSD_STATUS | NUMBER(4,0) | Yes | The current status of the direct delivery. The valid values: 0 - New, 1 - In Progress, 2 - Received, 3 - Canceled, 4 - Rejected, 9999 - Active. |
| DSD_EXPECTED_DATE | TIMESTAMP(6) | Yes | The expected date of the delivery. |
| DSD_CREATE_DATE | TIMESTAMP(6) | Yes | The date the direct delivery record was created. |
| DSD_UPDATE_DATE | TIMESTAMP(6) | Yes | The date the direct delivery was last updated. |
| DSD_RECEIVED_DATE | TIMESTAMP(6) | Yes | The received date of the Delivery. |
| DSD_INVOICE | VARCHAR2(128) | Yes | The unique identifier of the invoice that this direct delivery is associated to, as entered by the user. |
| DSD_INVOICE_DATE | TIMESTAMP(6) | Yes | The date of the invoice as entered by the user. |
| DSD_CREATE_USER | VARCHAR2(128) | Yes | The user who created the Delivery. |
| DSD_CUST_ORD_ID | VARCHAR2(128) | Yes | The customer order identifier. |
| DSD_FUL_ORD_EXT_ID | VARCHAR2(128) | Yes | The fulfillment order external identifier. |
| DSD_CARRIER_ENTITY | VARCHAR2(128) | Yes | The carrier name. |
| DSD_CARRIER_TYPE | NUMBER(2,0) | Yes | Indicate carrier type, 0 - Corporate, 1 - Third Party |
| DSD_CARRIER_CODE | VARCHAR2(4) | Yes | Unique code identifier for a carrier. |
| DSD_COUNTRY_CODE | VARCHAR2(3) | Yes | The Country code. |

ORACLE®

| Name | Type | Optional (Null?) | Description |
|---|---|---|---|
| DSD_SOURCE_ADDRESS | VARCHAR2(1000) | Yes | The address of source sending delivery. |
| DSD_LICENSE_PLATE | VARCHAR2(128) | Yes | The license plate. |
| DSD_FREIGHT_ID | VARCHAR2(128) | Yes | The freight identifier. |
| DSD_CURRENCY_CODE | VARCHAR2(3) | Yes | The currency code. |
| DSD_INVOICE_COST | NUMBER(12,4) | Yes | The invoice cost value. |
| DSD_RECEIVED_USER | VARCHAR2(128) | Yes | The user who received the delivery. |
| DSD_UPDATE_USER | VARCHAR2(128) | Yes | The user who last updated the delivery. |
| DSD_BOL_EXT_ID | VARCHAR2(128) | Yes | The Delivery Bill Of Lading from external system or entered by SIM user. |
| DSD_ADJ_PO_EXT_ID | VARCHAR2(128) | Yes | The purchase order external identifier. |
| DSD_ADJ_ITEM | VARCHAR2(25) | Yes | The receiver unit adjustment item identifier. |
| DSD_ADJ_DATE | TIMESTAMP(6) | Yes | The receiver unit adjustment date. |
| DSD_ADJ_QUANTITY | NUMBER(20,4) | Yes | The receiver unit adjustment quantity. |
| DSD_CARTON_EXT_ID | VARCHAR2(128) | Yes | The unique external identifier of the container. |
| DSD_CARTON_STATUS | NUMBER(4,0) | Yes | Container status. valid values- 1 - New, 2 - In Progress, 3 - Submitted, 4 - Received, 5 - Damaged, 6 - Missing, 7 - Canceled |
| DSD_CARTON_STORE | NUMBER(10,0) | Yes | The unique store identifier to which container is associated to. |
| DSD_CARTON_REFERENCE_ID | VARCHAR2(128) | Yes | The reference identifier. |
| DSD_CARTON_DAMAGED_REASON | VARCHAR2(128) | Yes | The reason for container damage. |
| DSD_CARTON_SERIAL_CODE | NUMBER(18,0) | Yes | The serial code. |
| DSD_CARTON_TRACKING_NUMBER | VARCHAR2(128) | Yes | The Tracking number for the container. |
| DSD_CARTON_DAMAGE_REMAINING | VARCHAR2(1) | Yes | Indicates all remaining quantities should be damaged on final receipt. |
| DSD_CARTON_CREATE_USER | VARCHAR2(128) | Yes | The user who created the direct delivery container. |
| DSD_CARTON_UPDATE_USER | VARCHAR2(128) | Yes | The user who last updated the direct delivery container. |
| DSD_CARTON_RECEIVE_USER | VARCHAR2(128) | Yes | The user who received the direct delivery container. |
| DSD_CARTON_CREATE_DATE | TIMESTAMP(6) | Yes | The date when the direct delivery container was created. |

| Name | Type | Optional (Null?) | Description |
|---|---|---|---|
| DSD_CARTON_UPDATE_DATE | TIMESTAMP(6) | Yes | The date when the direct delivery container was updated. |
| DSD_CARTON_RECEIVE_DATE | TIMESTAMP(6) | Yes | The date when the direct delivery container was received. |
| DSD_CARTON_UIN_REQUIRED | VARCHAR2(1) | Yes | Indicates whether the container has uin items. Valid values: Y or N. |
| DSD_CARTON_RCV_AT_SHOP_FLOOR | VARCHAR2(1) | Yes | Indicates whether the inventory on container is to be received at shop floor. Valid values: Y or N. |
| DSD_CARTON_QUALITY_CONTROL | VARCHAR2(1) | Yes | Used to flag containers that require detailed receiving. |
| DSD_CARTON_EXTERNAL_CREATE | VARCHAR2(1) | Yes | Indicates whether the delivery is external created. Valid values: Y or N. |
| DSD_CARTON_ADJUSTED | VARCHAR2(1) | Yes | Indicates whether the delivery is adjusted. Valid values: Y or N. |
| DSD_CARTON_CUST_ORD_RELATED | VNUMBER(4,0) | Yes | Indicates if container contains customer order items. 1 - Yes, 2 - Mix, 3 - No |
| DSD_LINE_ITEM | VARCHAR2(25) | Yes | The unique identifier of the item that is contained in this container. |
| DSD_LINE_CASE_SIZE | NUMBER(10,2) | Yes | The number of units in the case that this item was shipped in. |
| DSD_LINE_EXPECTED | NUMBER(20,4) | Yes | The total number of units expected in this direct delivery. |
| DSD_LINE_RECEIVED | NUMBER(20,4) | Yes | The total number of units received in this direct delivery. |
| DSD_LINE_DAMAGED | NUMBER(20,4) | Yes | The total number of units that were damaged when the direct delivery was received. |
| DSD_LINE_UNIT_COST_CURR | VARCHAR2(3) | Yes | The unit cost currency code for the line item. |
| DSD_LINE_UNIT_COST_VAL | NUMBER(12,4) | Yes | The unit cost value for the line item. |
| DSD_LINE_OVRD_UNIT_COST_CURR | VARCHAR2(3) | Yes | The override unit cost currency. |
| DSD_LINE_OVRD_UNIT_COST_VAL | NUMBER(12,4) | Yes | The override unit cost value. |
| DSD_LINE_STORE | NUMBER(10,0) | Yes | The store identifier. |
| DSD_LINE_PO_EXT_ID | VARCHAR2(128) | Yes | The purchase order external identifier. |
| DSD_LINE_CUST_ORD_ID | VARCHAR2(128) | Yes | The customer order identifier. |
| DSD_LINE_FUL_ORD_EXT_ID | VARCHAR2(128) | Yes | The fulfillment order external identifier. |
| DSD_LINE_VPN | VARCHAR2(256) | Yes | The vendor product number. |

| Name | Type | Optional (Null?) | Description |
|------|------|------------------|-------------|
| DSD_LINE_RECEIVED_OVER | NUMBER(20,4) | Yes | Amount of received inventory over the expected quantities. |
| DSD_LINE_DAMAGED_OVER | NUMBER(20,4) | Yes | Amount of damaged inventory over the expected quantities. |
| DSD_LINE_PREV_RECEIVED | NUMBER(20,4) | Yes | Units previous received when container is reopened for adjustment. |
| DSD_LINE_PREV_DAMAGED | NUMBER(20,4) | Yes | Units previous received as damaged when container is reopened for adjustment. |

**DSD Sample Data File**

Comma Delimited File with fields defined (see Interface Definitions)

```
"REPLACE","ASN-EXT_PO_0",4241,770,1,,2022-04-15 11:00:00.000000000,2022-04-15
11:00:00.000000000,,"",,"SIM","","","",,"","US","","","","USD",,"","SIM","","EXT
_PO_0","100050005",2022-04-15
11:00:00.000000000,20,"",,,"","",,"","","","","",,,,"","","","","",,"",,,,,"",,"
",,,"","","","",,,,
```

# Purchase Order Interface Specification

**Purchase Order Sample File**

Comma Delimited File with fields defined (see Interface Definitions)

"REPLACE","EXT_PO_1",4241,"55",1,2022-04-07 11:00:00.000000000,2022-04-07
11:00:00.000000000,2022-04-07
11:00:00.000000000, ,"TEST_USER","test1","CO0","FOE0",3,"100750010","US",1,10,0,"USD
",10,"EA","N"

# Transfer Interface Specification

| Name | Type | Optional (Nullable) | Description |
|------|------|---------------------|-------------|
| BDI_DATASET_ACTION | VARCHAR2(20) | No | The required record action type, valid values are: REPLACE. |
| TSF_DOC_TYPE | VARCHAR2(2 BYTE) | No | The type of the document type. valid values: T - Transfer, A - Transfer Allocation. |
| TSF_EXT_ID | VARCHAR2(128 BYTE) | No | The external identifier of transfer/ transfer allocation. |
| TSF_SRC_TYPE | NUMBER(2,0) | Yes | Source location type for transfer/ transfer allocation. 1- Store, 3 Warehouse, 4 - Finisher |
| TSF_SRC | NUMBER(10,0) | Yes | The identifier of the source that will send the items on the transfer / Transfer Allocation. |

| Name | Type | Optional (Nullable) | Description |
|------|------|---------------------|-------------|
| TSF_DEST_TYPE | NUMBER(2,0) | Yes | Destination location type for the transfer. 1- Store, 3 Warehouse, 4 - Finisher |
| TSF_DEST | NUMBER(10,0) | Yes | The identifier of the destination that will receive the items on the transfer. |
| TSF_STATUS | NUMBER(2,0) | Yes | The status of transfer/transfer_allocation. Valid Transfer Allocation Status: 1 - Approved,2 - Completed, 3 - Canceled. Transfer Status: 1 - New Request, 2 - Requested, 3 - Request In-Progress, 4 - Rejected Request, 5 - Canceled Request, 6-Transfer In-Progress, 7 - Approved, 8 - In-Shipping, 9 - Completed Transfer, 10 - Canceled Transfer. |
| TSF_CREATE_USER | VARCHAR2(128 BYTE) | Yes | A person who created a transfer. |
| TSF_UPDATE_USER | VARCHAR2(128 BYTE) | Yes | The last user to update the transfer. |
| TSF_APPROVAL_USER | VARCHAR2(128 BYTE) | Yes | The user that approved the transfer. |
| TSF_CREATE_DATE | TIMESTAMP(6) | Yes | The original or create date of a transfer. |
| TSF_UPDATE_DATE | TIMESTAMP(6) | Yes | The date the transfer was last updated. |
| TSF_APPROVAL_DATE | TIMESTAMP(6) | Yes | The date the transfer was approved. |
| TSF_CONTEXT_ID | NUMBER(18,0) | Yes | The context identifier for Transfer / Transfer Allocation. |
| TSF_CONTEXT_VALUE | VARCHAR2(25 BYTE) | Yes | This field holds value for Transfer / Transfer Allocation relating to the context type, for example promotion number. |
| TSF_CUST_ORD_ID | VARCHAR2(128 BYTE) | Yes | Customer order external identifier for transfer. |
| TSF_FUL_ORD_EXT_ID | VARCHAR2(128 BYTE) | Yes | Fulfillment order external identifier for transfer. |
| TSF_PARTIAL_DELIVERY_IND | VARCHAR2(1 BYTE) | Yes | Indicate if partial delivery is allowed in transfer. |
| TSF_AUTH_CODE | VARCHAR2(12 BYTE) | Yes | Authorization code for the transfer. |
| TSF_NOT_AFTER_DATE | TIMESTAMP(6) | Yes | Date after which the transfer is not valid. |
| TSF_REQUEST_DATE | TIMESTAMP(6) | Yes | Request date of the transfer. |
| TSF_REQUEST_USER | VARCHAR2(128 BYTE) | Yes | The user that requested the transfer. |
| TSF_USE_AVAILABLE | VARCHAR2(1 BYTE) | Yes | Indicates if the transfer is for unavailable stock. |
| TSF_DELV_SLOT_ID | VARCHAR2(15 BYTE) | Yes | Delivery time slot id for allocation. |

**ORACLE**

| Name | Type | Optional (Nullable) | Description |
| --- | --- | --- | --- |
| TSF_DELIVERY_DATE | TIMESTAMP(6) | Yes | The date that the allocation is expected to be delivered from the warehouse. |
| TSF_LINE_ITEM | VARCHAR2(25 BYTE) | Yes | The identifier of the item that this allocation/transfer is for. |
| TSF_LINE_DIST_PARENT_ID | VARCHAR2(25 BYTE) | Yes | A unique identifier of the parent transfer document that is associated to this allocation. |
| TSF_LINE_CASE_SIZE | NUMBER(10,2) | Yes | The size of the transfer item pack. |
| TSF_LINE_EXPECTED | NUMBER(20,4) | Yes | The total number of units expected for this allocation. |
| TSF_LINE_RECEIVED | NUMBER(20,4) | Yes | The total number of units received for this allocation/transfer. |
| TSF_LINE_DAMAGED | NUMBER(20,4) | Yes | The total number of units that were damaged for this allocation/transfer. |
| TSF_LINE_REQUESTED | NUMBER(20,4) | Yes | Requested quantity for the transfer. |
| TSF_LINE_APPROVED | NUMBER(20,4) | Yes | Approved quantity for the transfer. |
| TSF_LINE_SHIPPING | NUMBER(20,4) | Yes | Shipping quantity for the transfer. |
| TSF_LINE_SHIPPED | NUMBER(20,4) | Yes | Shipped quantity for the transfer. |
| TSF_LINE_PREFERRED_UOM | VARCHAR2(4 BYTE) | Yes | Preferred unit of measure of the transfer line item. |

**Transfer Sample File**

Comma Delimited File with fields defined (see Interface Definitions)

```
"REPLACE","T","EXT_TSF_1",1,4241,1,5251,1,"TEST_USER","TEST_USER","TEST_USER",20
22-04-07 11:00:00.000000000,2022-04-07 11:00:00.000000000,2022-04-07
11:00:00.000000000,1,"PROM_TSF","CO1","FOE1","Y","AUTH_1234",2022-04-07
11:00:00.000000000,2022-04-07
11:00:00.000000000,"TEST_USER","Y","",,"100050005","",1,,0,0,100,100,0,0,"EA"
```

# Setup BDI CLI File Transmitter Tool

This section describes steps to install the BDI Transmitter CLI for 64-bit Linux, using an archive binary file (.zip), the configurations include the Open Transaction Modules from external systems.

For additional BDI CLI Transmitter CLI overview and usages, see RICS documentation:

https://docs.oracle.com/cd/B31315_01/191000/BDI%20Implementation%20Guide/Output/Chapter%207%20-%20cli-tools.htm#sthref346 Oracle® Retail Bulk Data Integration Cloud Service Implementation Guide

1. Download CLI Transmitter

   Download the BdiCliTransmitter<version>ForAll<version>Apps_eng_ga.zip.

   The archive binary can be installed in any location that you can write to.

2. Set CLI tool installation directory

Change directory to the location where you would like transmitter to be installed. Move the .zip archive binary to the current directory.

3. Unpack the transmitter

Unpack the zip and install the Transmitter.

unzip BdiCliTransmitter<version>ForAll<version>Apps_eng_ga.zip

You may delete the .zip file if you want to save disk space.

4. Modify bdi-file-transmitter.properties:

```
….source.system.name=EXTsim.receiver.url=https://<ext-lb> /sim-batch-
job-admin/resources/
receiversim.receiver.url.useralias=simReceiverUrlUserAlias#list of
interface modules#interfaceModules=ExtPurchOrder_Tx,
ExtDsd_Tx,ExtTsf_TxinterfaceModules=ExtPurchOrder_Tx#list of interface
namesExtPurchOrder_Tx.receiver.appname=simExtPurchOrder_Tx.dataset.typ
e=PARTIALExtPurchOrder_Tx.interfaceShortNames=Ext_Purch_OrderExtPurchO
rder_Tx.Ext_Purch_Order.input.filepath= /data-files/
po#ExtDsd_Tx.receiver.appname=sim #ExtDsd_Tx.dataset.type=PARTIAL
#ExtDsd_Tx.interfaceShortNames=Ext_Dsd#ExtDsd_Tx.Ext_Dsd.input.filepat
h=/data-files/dsd#ExtTsf_Tx.receiver.appname=sim
#ExtTsf_Tx.dataset.type=PARTIAL
#ExtTsf_Tx.interfaceShortNames=Ext_Tsf#ExtTsf_Tx.Ext_Tsf.input.filepat
h=/data-files/tsf
```

(Notes: Based on your integration need, you can uncomment out the modules you want to seed into receiver app).

5. Set JAVA_HOME environment variable

For example,

export JAVA_HOME=/vol.rtk/java/oracle_linux/jdk1.8.0_66.64bit

6. Setup credential

**$ ./bdi-file-transmitter.sh -setup-credentials**

On prompt, enter receiver user

Where the receiver is assigned BdiEdgeSimJobOperatorGroup in IDCS), for example, simbdioperator

(Notes: if you add a new module or modify the transmitter properties, you need to run ./ **bdi-file-transmitter.sh -setup-credentials**)

7. Get Interface Metadata

$ ./bdi-file-transmitter.sh -get-interface-metadata sim

(Notes: to verify the Module>_metadata.json generated correctly, please check the contents of the metadata json file contains the interface table definition, which is located at <bdi-cli-transmitter>/conf/meta-data/sim

If the metadata json contains empty columns for specific interfaces, inspect the interface names in the returned <module>_metadata.json file to see if there is an empty space. If there is an empty space, then you need to remove the empty space in bdi-file-transmitter.properties, and then bounce the sim-bdi server to remove the metadata cache, and then re-execute the get metadata step)

# Run BDI CLI File Transmitter

1. Place the data file into the location listed in the bdi-file-transmitter.properties

2. Run transmitter as below:

   **$ ./bdi-file-transmitter.sh -m** <interfaceModuleName>

   For example:

   **$ ./bdi-file-transmitter.sh -m ExtPurchOrder_Tx**

3. Review the Receiver log

   Login to BDI sim-batch-job-admin console -> navigate to System Logs -> Find the Receiver Log (for example, <sourceSyste>_ExtPurchseOrder_Tx_ReceiverService-bdi-server-system.log)

   Note down the source data set id (you will need the data set id later in Importer Job step).

**Figure L-1    Source Data Set**



# Run Importer Job

To run the importer job, follow these steps:

Once the BDI process is completed, you can load data from EICS inbound stage tables into the application tables.

To load stage data to application tables:

From Main Menu, navigate to **Admin**/**Technical Maintenance**/**Job Admin**.

The job would be available for the Admin user who is authorized to access/run the Data Seeding batch processes via EICS user interface. User will need to navigate to following screen to access and run these batch processes.

1. Login to EICS via browser

2. On **Navigation** menu select **Admin/ Technical Maintenance**/**Job Admin**/**Data Seed**.

   User will be taken to **Data Seed** screen.

3. Select the desired **Importer** job from drop down list:

   The supported BDI Open Transaction Imports are:

   • Third Party Purchase Order Import

- Third Party Transfer Import
- Third Party Vendor Delivery Import

4. Enter the data set identifier.

> **Note:**
>
> The data set id is required if importer data is not coming from BDI Process Flow. When importer data is coming from BDI CLI tool, user would need to find the dataset id for running the importer. The receiver source data set id can be identified via BDI SIM Job admin log.

5. Click on **Start Job**.

6. View **Importer Job Execution**:

   Login SIOCS JET UI, navigate to **Job Admin** screen

   By default, the job admin screen lists the failed job execution. To view the completed importer job, you may need to filter the job status by selecting the job status.

   To view job execution details, user can click on the job execution link and navigate to the detail screen to see the exact reason of failure.

# M

# Appendix: Third Party Pricing Integration

## Overview

By default, EICS integrate with RPCS clearance and regular price change via BDI process flow.

EICS does not integrate with Promotion Offer interface due to it does not provide pricing at item/date level, so there is nothing for SIOCS to consume.

As an alternative, SIOCS does have a generic pricing interface for the retailer to integrate promotion price into.

Third Party Pricing Import provides a generic pricing interface for retailer which can provide pricing data for EICS pricing imports from third party. This third-party pricing import provides an alternative for import promotion data into EICS.
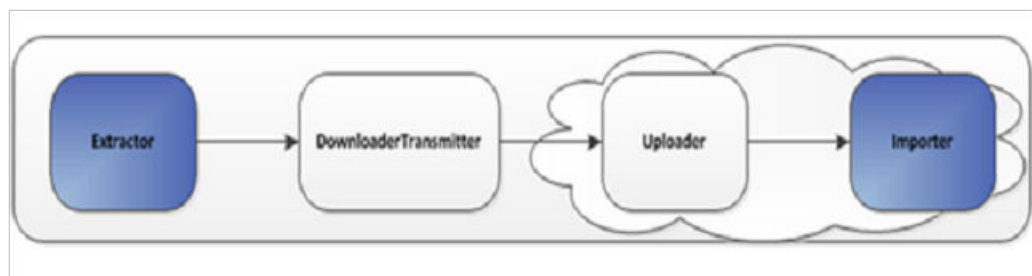
## BDI Implementation Considerations

To load third-party data into EICS via BDI, the external system needs to provide data sourcing implementations.

There are two approaches for data sourcing implementations, Integration Implementer may choose the data provider implementations based on their need.

**Option A: Third Party Pricing Integration Via BDI External Process Flow Implementation**

**Figure M-1    BDI Process Flow Process**



The BDI process:

- An Extractor extracts data from a data producing system into an outgoing intermediate integration interface table.
- A Downloader and transporter services data to receiver service.
- An Uploader uploads the data to an incoming intermediate integration interface table.

- An Importer imports the data from the interface stage tables into SIOCS application tables.

For more information about data seeding, see Data Seeding.

With this approach, the BDI Ext Process Flow will move the data from sourcing system to destination system's application table. BDI process flow provides BDI job scheduling, end to end data transportation from sourcing system's database table into destination application's tables.

Customers need to implement the BDI Ext-EdgeApp to move data into the BDI outbound interface table.

BDI Process Flow: ExtPrice_Tx_ProcessFlow_From_EXTERNAL

See Oracle Retail Enterprise Integration Overview Guide: BDI Third Party Integrations.

**Option B: Third Party Pricing Integration Via BDI CLI File Transmitter**

BDI CLI File Transmitter is a client command line tool, is applicable for smaller data file size which does not require concurrent processing.

BDI CLI file transmitter only transport data files from sourcing systems to destination system's interface tables. It does not provide end to end data transportation. If you choose BDI CLI File transmitter for load third party data into EICS application table, it will require three steps:

**Figure M-2    BDI CLI File Transmitter**
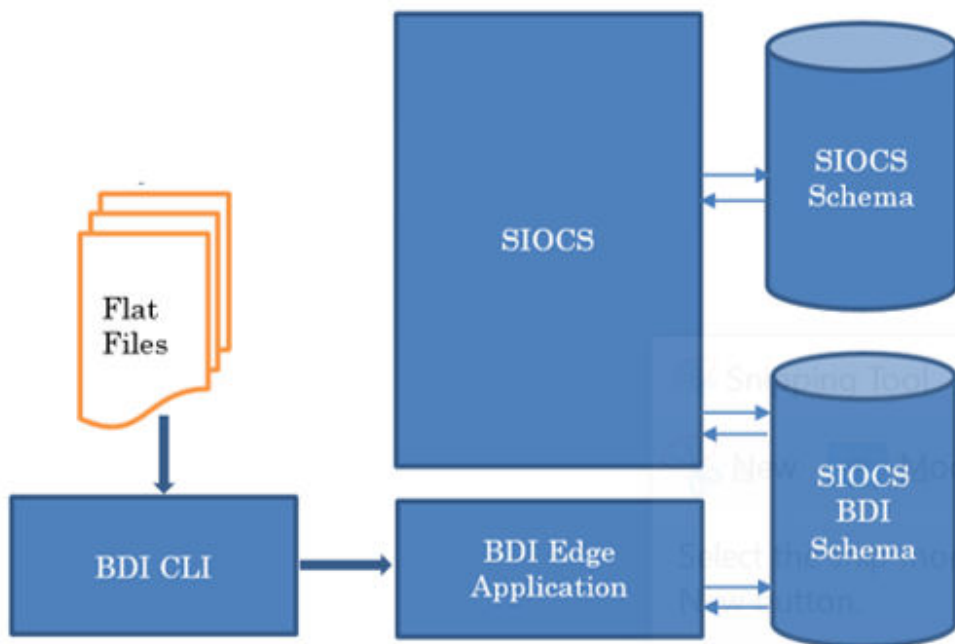


- Download BDI CLI File Transmitter and install on customer's system.
- Run BDI CLI Transmitter batch to load data file from sourcing system into destination system's interface table.

- Once the data are loaded into receiver app's inbound interface tables, EICS's importer job will move the data from inbound interface table into EICS app tables.

> **📝 Note:**
>
> By default, batch scheduler Third Party Price Import in EICS is disabled; this job should only be enabled if you choose to use BDI CLI File transmitter for data sourcing provider.

See Integration.

This section provides BDI CLI File Transmitter as an implementation to describe how to import third partying pricing into EICS via BDI CLI File Transmitter.

# Third Party Pricing Import via BDI CLI File Transmitter

Each file contains price records for a single store, store/item/action date uniquely identify a pricing record.

### 1. BDI CLI File Transmitter Process

The BDI CLI Transmitter process load data files to an intermediate BDI inbound tables in SIOCS BDI Schema.

Once the records are created on the SIOCS BDI table, BDI process also create a control record for the module and interface for importer to process. The module interface along with the data set id will be used by importer to identify the records to be imported into the EICS in importer process.

See Appendix: BDI Transmitter CLI Procedure - Load External Data File for details on how to run BDI CLI File Transmitter.

Sample bdi-file-transmitter.properties for ExtPrice_Tx Module:

```
...

source.system.name=ext-source

sim.receiver.url=https://<bdi-sim-app-hostname>:<port>/sim-batch-job-admin/
resources/receiver sim.receiver.url.useralias=simReceiverUrlUserAlias

interfaceModules=ExtPrice_Tx

ExtPrice_Tx.receiver.appname=sim

ExtPrice_Tx.dataset.type=PARTIAL

ExtPrice_Tx.interfaceShortNames=Ext_Price

ExtPrice_Tx.Ext_Price.input.filepath=/home/bdi/ext_price_tx/ext_price/files
```

### 2. EICS Importer Process

Once the file data is loaded into the SIOCS BDI inbound interface table, the importer process will process the BDI interface tables into SIOCS application tables. You can setup SIOCS

Batch job schedule to import data from inbound interface tables into EICS application tables. For importer details, see Third Party Pricing Import in batch chapter for details.

**3. Job Scheduling**

You can schedule EICS job scheduler to start the third-party price TX importer job automatically, or You can run importer as adhoc job.

By default, EICS batch scheduling for third party pricing TX import job is disabled, as for customers who choose BDI External Process Flow for third party pricing end to end integration, the BDI Job scheduler should be used.

**4. Error Handling and Logging**

When the job execution failed, only the failed record reference ids are stored in the BATCH_DATA_DETAIL table, the data which are successfully processed are removed from the BDI inbound table. Sys ops user can view the batch detail errors via EICS Job admin console. The number of maximum errors (for example 10 errors) per partition are logged for each batch/partition run.

# External Price File Layout

| Field Name | Description | Required | Type |
|---|---|---|---|
| BDI_DATASET_ACTION | EICS only support REPLACE OR DELETE as dataset action for third party pricing. UPDATE type is not supported, use REPLACE for updating a record. | Yes | CHAR(20) |
| ITEM_ID | The unique alphanumeric value for the transaction level item. | Yes | CHAR(25) |
| STORE_ID | The number that uniquely identifies the store. | Yes | Number(10) |
| EFFECTIVE_DATE | The date on which the price change became effective. yyyy-mm-dd hh:mm:ss.ffffffffff for example, 2021-04-09 11:00:00.000000000 | Yes | Timestamp |
| END_DATE | Promotion or clearance price end date. For price change with an end date, if the clearance indicator is 'Y', the end date is for clearance end date; otherwise, the end date is promotional ending date. yyyy-mm-dd hh:mm:ss.ffffffffff for example, 2021-04-09 11:00:00.000000000 | No | Timestamp |
| PRICE_TYPE | The item price type. Valid values: 200- Clearance 201- Promotional 202- Regular | Yes | NUMBER(3) |
| PROMOTION_NAME | Promotion name. | No | CHAR(160) |

| Field Name | Description | Required | Type |
|---|---|---|---|
| SELLING_UNIT_RETAIL | Contains the current single unit retail in the selling unit of measure. | Yes | NUMBER(20,4) |
| SELLING_UOM | Contains the selling unit of measure for an items single-unit retail. | Yes | CHAR(4) |
| SELLING_UNIT_RETAIL_CURRENCY | Contains the selling unit retail currency. | Yes | CHAR(3) |
| MULTI_UNITS | Contains the current multi-units. If the record is being written as a result of a change in the multi-unit retail, then this field contains the new multi-units. | No | NUMBER(12,4) |
| MULTI_UNIT_RETAIL | Contains the current multi-unit retail in the selling unit of measure. | No | NUMBER(20,4) |
| MULTI_UNIT_RETAIL_CURRENCY | Contains the multi-unit retail currency. | No | CHAR(3) |
| MULTI_UNIT_SELLING_UOM | Contains the selling unit of measure for an items multi-unit retail. | No | CHAR(4) |
| CREATE_DATETIME | Contains the record creation date. yyyy-mm-dd hh:mm:ss.fffffffff for example, 2021-04-09 11:00:00.000000000 | No | Timestamp |
| REC_ID | The id of the record. | Yes | NUMBER(15) |
| RETAIL_CHANGE_IND | Indicates whether the retail changed with this price change. Valid values are: 0 - retail price not changed 1 - retail price changed | No | NUMBER(6) |
| MULTI_UNIT_IMPACT | Indicates if the Price Change has impact to Multi Unit retail. Valid values are: AU - Multi Unit information is added or updated R - Multi Unit information is removed N - Multi unit information is not changed. | Yes | CHAR(4) |
| PRICE_EVENT_ID | The id of the price event. | Yes | NUMBER(15) |

**Sample CSV File:**

ext_price_<YYYYMMDDHHMMSS>.csv

```
"REPLACE","100637113",5000,2021-04-09
11:00:00.000000000,,202,"",149.99,"USD","EA",,,"","",2021-04-07
11:00:00.000000000,1,1,N,9999"REPLACE","100637148",5000,2021-04-09
11:00:00.000000000,,202,"",145.99,"USD","EA",,,"","",2021-04-07
11:00:00.000000000,1,1,N,9999
```

# File Contents Explanation

- Create CSV file: ext_price _<YYYYMMDDHHMMSS>.csv

- It is expected that the pricing provider to ensure the record uniqueness (A unique record is identified by store/item/effective date time), within a file, each record must be unique. The record action is denoted by action type, only a dataset action is allowed for unique store/item/date.

  For example, for store 5000, item A, a price on date 2018 Dec 10 00:00:00 record in the file can be one of the (REPLACE, DELETE), the same record with more than one dataset action will be rejected. EICS only supports REPLACE OR DELETE as dataset action for third party pricing. UPDATE type is not supported, use REPLACE for updating a record.

- To end clearance price: To end a clearance price for a store/item, the pricing provider can either include the end date on the clearance item record; or sending a new price record (regular or promotional).

- The same file can't have two records with this combination store/item/effective with different price type, if clearance need to be on today, then this file should only have a single record for clearance type; to end the clearance (for example, tomorrow), the file need either mark the end date for that clearance or need to include a regular price record with a different effective date (for example, tomorrow).

- In EICS, there is no client UI which needed the prom id, clearance id or price change id. For data import integration backend processing, the record is uniquely identified by item/store/effective date time and price type external pricing change identifier has no meaning to our system. Promotion name is used in EICS as context type, therefore include in the integration interface.

- **Split the Data into Multiple Files**

  EICS loads the data in parallel from multiple files. Loading files from multiple files in parallel provides performance advantage than loading from a single file. It is recommended to file provider to split the data into multiple files to load data efficiently in parallel loading, each file contains single store is recommended.

# N

# Appendix: Third-Party RFID Integration

## Overview

EICS allows import of RFID information via a BDI process flow (ExtRfid_Tx) from a third-party system. The External RFID import is an operational batch which can run as often as needed. This batch is used for importing bulk amount of RFID information. For small handful of RFID tags, RFID Inventory WebService can be used.

The RFID importer first sets all the present attribute to 'N' for all existing RFID tags at the location thereby removing them from inventory. It then inserts or replaces the RFID data from inbound interface table into RFID table in SIOCS, setting the present attribute to Y (yes) for each RFID tag in the import.

In this fashion, it operates much like a stock count and will generate inventory adjustments for external system to have their SOH updated for any discrepancies found.

## BDI Implementation Considerations

To load third-party data into EICS via BDI, external system need to provide data sourcing implementations.

There are two approaches for data sourcing implementations, Integration Implementer may choose the data provider implementations based on their need.

**Option A: Third-Party RFID Integration Via BDI External Process Flow Implementation**

**Figure N-1    BDI Process Flow Process**
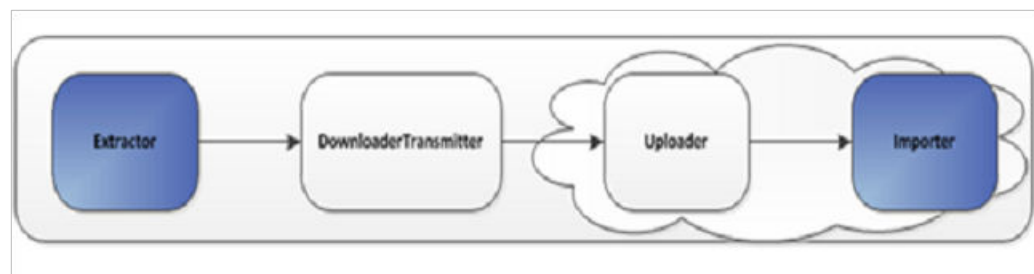


The BDI process:

- An Extractor extracts data from a data producing system into an outgoing intermediate integration interface table.
- A Downloader and transporter services data to receiver service.
- An Uploader uploads the data to an incoming intermediate integration interface tables.

- An Importer imports the data from the interface stage tables into SIOCS application tables.

For more information about data seeding, see Data Seeding.

With this approach, the BDI Ext Process Flow will move the data from sourcing system to destination system's application table. BDI process flow provides BDI job scheduling, end to end data transportation from sourcing system's database table into destination application's tables.

Customer need to implement the BDI Ext-EdgeApp to move data into the BDI outbound interface table.

BDI Process Flow: ExtRfid_Tx_ProcessFlow_From_EXTERNAL

See *Oracle Retail Enterprise Integration Overview Guide*: BDI Third-Party Integrations.

**Option B: Third-Party RFID Integration Via BDI CLI File Transmitter**

BDI CLI File Transmitter is a client command line tool, is applicable for smaller data file size which does not require concurrent processing.

BDI CLI file transmitter only transport data files from sourcing systems to destination system's interface tables. It does not provide end to end data transportation. If you choose BDI CLI File transmitter for load third-party data into SIOCS application table, it would require three steps:

**Figure N-2    BDI CLI File Transmitter**
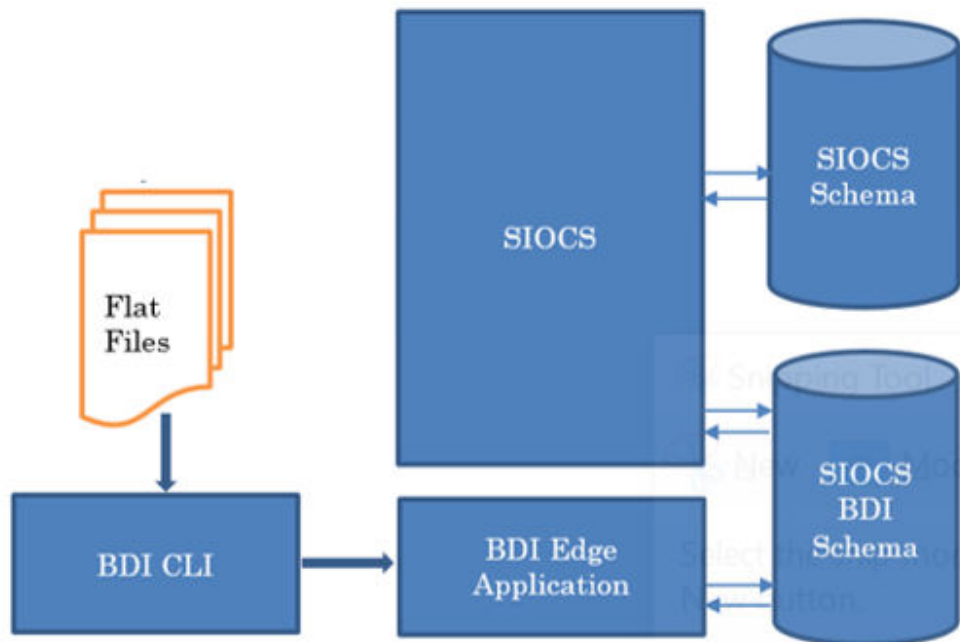


- Download BDI CLI File Transmitter and install on customer's system.
- Run BDI CLI Transmitter batch to load data file from sourcing system into destination system's interface table.

- Once the data are loaded into receiver app's inbound interface tables, EICS's importer job will move the data from inbound interface table into EICS app tables.

> **Note:**
>
> By default, batch scheduler Third-Party RFID Import in EICS is disabled; this job should only be enabled if you choose to use BDI CLI File transmitter for data sourcing provider.

See Integration.

This section provides BDI CLI File Transmitter as an implementation to describe how to import third-party RFID into SIOCS via BDI CLI File Transmitter.

# Third-Party RFID Import via BDI CLI File Transmitter

Each file contains RFID information for a single store, store/item/action date uniquely identify a RFID record.

**1. BDI CLI File Transmitter Process**

The BDI CLI Transmitter process load data files to an intermediate BDI inbound tables in SIOCS BDI Schema.

Once the records are created on the SIOCS BDI table, BDI process also create a control record for the module and interface for importer to process. The module interface along with the data set id will be used by importer to identify the records to be imported into the EICS in importer process.

See Appendix: BDI Transmitter CLI Procedure - Load External Data File for details on how to run BDI CLI File Transmitter.

**2. EICS Importer Process**

Once the file data is loaded into the SIOCS BDI inbound interface table, the importer process will process the BDI interface tables into SIOCS application tables. You can setup SIOCS Batch job schedule to import data from inbound interface tables into SIOCS application tables. For importer details, see Third-Party RFID Import in batch chapter for details.

**3. Job Scheduling**

You can schedule EICS job scheduler to start the third-party RFID TX importer job automatically, or you can run importer as adhoc job.

By default, SIOCS batch scheduling for third-party RFID TX import job is disabled, as for customers who choose BDI External Process Flow for third-party RFID end to end integration, the BDI Job scheduler should be used.

**4. Error Handling and Logging**

When the job execution failed, only the failed record reference ids are stored in the BATCH_DATA_DETAIL table, the data which are successfully processed are removed from the BDI inbound table. Sys ops user can view the batch detail errors via EICS Job admin

console. The number of maximum errors (for example 10 errors) per partition are logged for each batch/partition run.

# EXT_RFID_IN

| Field Name | Description | Required | Type |
|---|---|---|---|
| BDI_SEQ_ID | Bulk Data Integration data sequence. | Yes | NUMBER(19) |
| BDI_APP_NAME | Bulk Data Integration Application Name. | Yes | VARCHAR2(50) |
| BDI_DATASET_TYPE | Bulk Data Integration Data Set Type, valid values are FULL, or PARTIAL. | No | VARCHAR2(20) |
| BDI_DATASET_ACTION | EICS only support REPLACE and DELETE as dataset action for third-party RFID, UPDATE type is not supported, use replace for updating a record. | No | VARCHAR2(20) |
| EPC | Electronic product code (SGTIN-96). | Yes | VARCHAR(256) |
| ITEM_ID | Identifier of the item/sku. | Yes | VARCHAR2(25) |
| LOCATION_ID | Location identifier. | Yes | NUMBER(10) |
| LOCATION_TYPE | Location Type, 1 - store, 2 - warehouse. | Yes | NUMBER(2) |
| ZONE_ID | The zone within the location that the RFID is located. | No | NUMBER(15) |
| EVENT_DATE | The timestamp of the RFID read. | No | TIMESTAMP(6) |

# External RFID CSV File Fields

The CSV data fields matches the non-BDI heading columns in EXT_RFID_IN (four BDI heading columns are: BDI_SEQ_ID, BDI_APP_NAME, BDI_DATASET_TYPE, BDI_DATASET_ACTION).

For example, the first fields in the file would match the fourth column in EXT_RFID_IN table, the second field in the file matches the fifth column in the table.

**Sample CSV File:**

ext_rfid_<YYYYMMDDHHMMSS>.csv

```
"REPLACE","1111111111111111111111","100637113",5000,1,1001,"03-07-2021
0:00""REPLACE","11111111111111111111112","100637148",5000,2,1022,"05-10-202
1 0:00"
```

# File Contents Explanation

- Create CSV file: ext_rfid _<YYYYMMDDHHMMSS>.csv

- It is expected that the RFID provider to ensure the record uniqueness (A unique record is identified by store/item/effective date time), within a file, each record must be unique. The record action is denoted by action type, only one dataset action is allowed. EICS only support REPLACE OR DELETE as dataset action for third-party rfid, UPDATE type is not supported, use replace for updating a record.

- **Split the Data into Multiple Files**

  EICS loads the data in parallel from multiple files. Loading files from multiple files in parallel provides performance advantage than loading from a single file. It is recommended to file provider to split the data into multiple files to load data efficiently in parallel loading, each file contains single store is recommended.

- **Compress the data files**

  If data file contains large datasets, it is recommended that compress the load files individually, when load the data file. Use EICS System Configuration Console to specify the file suffix (for example, gzip, or zip).

# O

# Appendix: Item Ticket

EICS allows ticket generation and printing from web client as well as MAF client. The ticket created from MAF client can be printed directly to a Bluetooth printer. The set of ticket attributes when printed from a Bluetooth Printer differs from that sent to external web service. The list for each is given below.

**Table O-1    Ticket Attributes - Print to Bluetooth Printer**

| Field Name | Description |
| --- | --- |
| item ID | Item Identifier. |
| shortDescription | Item short description. |
| longDescription | Item long description. |
| shortDescriptionLang | The short description in the language of the store. |
| longDesccriptionLang | The long description in the language of the store. |
| departmentId | Merch hierarchy department Identifier. |
| departmentName | Merch hierarchy department name. |
| classId | Merch hierarchy class Identifier. |
| className | Merch hierarchy class name. |
| subclassId | Merch hierarchy sub-class Identifier. |
| subclassName | Merch hierarchy sub-class name. |
| differentiatorType1 | The description of the type for differentiator 1. |
| differentiatorType2 | The description of the type for differentiator 2. |
| differentiatorType3 | The description of the type for differentiator 3. |
| differentiatorType4 | The description of the type for differentiator 4. |
| differentiator1 | The description of differentiator 1. |
| differentiator2 | The description of differentiator 2. |
| differentiator3 | The description of differentiator 3. |
| differentiator4 | The description of differentiator 4. |
| printQuantity | The quantity to be printed on each ticket. |
| overridePrice | The override price given while creating the ticket. |
| printPrice | The ticket price. |
| printPriceType | The type of the ticket price. |
| printPriceActiveDate | The date the ticket price became active. |
| printPriceExpireDate | The date the ticket price expires. |
| printPriceUom | The unit of measure of the ticket price. |
| multiUnitPrice | Ticket multi-unit price. |
| multiUnitPriceUom | The unit of measure of the ticket multi-unit price. |
| multiUnitQuantity | The multi-unit quantity associated to the price. |

**Table O-2    Ticket Attributes - Print to Web-service**

| Field Name | Description |
| --- | --- |
| ticketId | Ticket Identifier. |
| item Id | Item Identifier. |
| primaryUpc | The Unique Product Code for the item. |
| originType | The origin type of the ticket. Valid values are:<br>1- External Batch<br>2- Price Change,<br>3- Foundation,<br>4- Manual |
| sequenceNumber | The sequence number of a ticket within its grouping. |
| ticketCount | The number of instances of this ticket to print. |
| shortDescription | Item short description. |
| longDescription | Item long description. |
| shortDescriptionLang | The short description in the language of the store. |
| longDesccriptionLang | The long description in the language of the store. |
| departmentId | Merch hierarchy department Identifier. |
| departmentName | Merch hierarchy department name. |
| classId | Merch hierarchy class Identifier. |
| className | Merch hierarchy class name. |
| subclassId | Merch hierarchy sub-class Identifier. |
| subclassName | Merch hierarchy sub-class name. |
| differentiatorType1 | The description of the type for differentiator 1. |
| differentiatorType2 | The description of the type for differentiator 2. |
| differentiatorType3 | The description of the type for differentiator 3. |
| differentiatorType4 | The description of the type for differentiator 4. |
| differentiator1 | The description of differentiator 1. |
| differentiator2 | The description of differentiator 2. |
| differentiator3 | The description of differentiator 3. |
| differentiator4 | The description of differentiator 4. |
| printQuantity | The quantity to be printed on each ticket. |
| overridePriceCurrency | The currency code of an override price. |
| overricePriceValue | The currency value of an override price. |
| priceCurrency | The currency code of the ticket price. |
| priceValue | The value of the ticket price. |
| printPriceType | The type of the ticket price. |
| printPriceActiveDate | The date the ticket price became active. |
| printPriceExpireDate | The date the ticket price expires. |
| printPriceUom | The unit of measure of the ticket price. |
| multiUnitPriceCurrency | The currency code of a ticket multi-unit price. |

**Table O-2    (Cont.) Ticket Attributes - Print to Web-service**

| Field Name | Description |
| --- | --- |
| multiUnitValue | The value of the ticket multi-unit price. |
| multiUnitPriceUom | The unit of measure of the ticket multi-unit price. |
| multiUnitQuantity | The multi-unit quantity associated to the price. |
| countryOfManufacture | A two letter country code denoting the country of manufacture for the item. |

# P

# Appendix: SFTP File Directory and Filename Format

This section lists the sFTP directories where user can upload and download batch files via sFTP server.

For additional details on sFTP account setup, see *Oracle® Retail Enterprise Inventory Cloud Service Security Guide* Nightly Batch File Uploads and Export File Downloads sections.

## File Upload Directory and Filename Format

**Table P-1    File Upload Directory and Filename Format**

| Import File Name | sFTP Directory | Filename Format |
|---|---|---|
| Clearance Price Change Import File | User logs into the SFTP server, and transfer data files to the directory:<br><br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br><event type>_<date in YYYYMMDDHH24MISS format>_<loc type>.zip<br>Where event type value is "CLRPC". The loc type field would be either 'S' or 'W'.<br>The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>CLRPC_20180129133250_S.zip<br>CLRPC_20180129133250_S.zip.complete<br>**Filename Inside Zip**<br><event type>_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>.dat<br>Where event type value is "CLRPC" (clearances), the loc type field would be either 'S' or 'W'.<br>Example:<br>CLRPC_20180129133250_1111_S.dat |

**Table P-1    (Cont.) File Upload Directory and Filename Format**

| Import File Name | sFTP Directory | Filename Format |
|---|---|---|
| Initial Inventory Import File | User logs into the SFTP server, and transfer data files to the directory:<br><br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br><br><file prefix>_<date in YYYYMMDDHH24MISS format>.zip<br><br>Where file prefix value is EXTSTK_. The zip file can contain one or more .date files.<br><br>The complete file needs to be added for zip file for job to pick it for processing.<br><br>Example:<br><br>EXTSTK _20180129133250.zip<br>EXTSTK_20180129133250.zip.complete<br><br>**Filename Inside Zip**<br><br><file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat<br><br>Where file prefix value is EXTSTK_ and loc id is the store identifier. This allows file to be unique for every upload.<br><br>Example:<br><br>EXTSTK_20180129133250_1111.dat |
| POS Sale Transaction Import File | User logs into the SFTP server, and transfer data files to the directory:<br><br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br><br><file prefix>_<date in YYYYMMDDHH24MISS format>.zip<br><br>Where file prefix value is "SIMTLOG". The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.<br><br>Example:<br><br>SIMTLOG_20180129133250.zip<br>SIMTLOG_20180129133250.zip.complete<br><br>**Filename Inside Zip**<br><br><file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat<br><br>Where file prefix value is "SIMTLOG" and loc id is the store identifier. This allows file to be unique for every upload.<br><br>Example:<br><br>SIMTLOG_20180129133250_1111.dat |

**Table P-1    (Cont.) File Upload Directory and Filename Format**

| Import File Name | sFTP Directory | Filename Format |
|---|---|---|
| Promotion Price Change Import File | User logs into the SFTP server, and transfer data files to the directory:<br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br><event type>_<date in YYYYMMDDHH24MISS format>_<loc type>.zip<br>Where event type value is "PRMPC". The loc type field would be either 'S' or 'W'.<br>The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>PRMPC_20180129133250_S.zip<br>PRMPC_20180129133250_S.zip.complete<br>**Filename Inside Zip**<br><event type>_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>.dat<br>Where event type is "PRMPC" (promotional price change). The loc type field would be either 'S' or 'W'.<br>Example:<br>PRMPC_20180129133250_1111_S.dat |
| Regular Price Change Import File | User logs into the SFTP server, and transfer data files to the directory:<br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br><event type>_<date in YYYYMMDDHH24MISS format>_<loc type>.zip<br>Where event type value is "REGPC". The loc type field would be either 'S' or 'W'.<br>The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>REGPC_20180129133250_S.zip<br>REGPC_20180129133250_S.zip.complete<br>**Filename Inside Zip**<br><event type>_<date in YYYYMMDDHH24MISS format>_<loc id>_<loc type>.dat<br>Where event type is "REGPC" (regular price changes). The loc type field would be either 'S' or 'W'.<br>Example:<br>REGPC_20180129133250_1111_S.dat |

**Table P-1    (Cont.) File Upload Directory and Filename Format**

| Import File Name | sFTP Directory | Filename Format |
|---|---|---|
| Retail Sale Audit Import File | Notes:<br><br>When Retail Sale Audit Import File from MFCS will be auto-pushed to SIOCS sFTP server directory<br><br>/{sftp user}<br><br>For third-party provided ReSA file, user logs into the SFTP server, and transfer data files to the directory:<br><br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br>&lt;file prefix&gt;_&lt;date in YYYYMMDDHH24MISS format&gt;.zip<br>Where file prefix value is "SIMT". The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>SIMT_20180129133250.zip<br>SIMT_20180129133250.zip.complete<br>**Filename Inside Zip**<br>&lt;file prefix&gt;_&lt;date in YYYYMMDDHH24MISS format&gt;_&lt;loc id&gt;.dat<br>Where file prefix value is "SIMT" and loc id is the store identifier. This allows file to be unique for every upload.<br>Example:<br>SIMT_20180129133250_1111.dat |
| Store Sequence Data Parser Import File | User logs into the SFTP server, and transfer data files to the directory:<br><br>/{sftp user}/ from_thirdparty | **Zip Filename Format**<br>&lt;file prefix&gt;_&lt;date in YYYYMMDDHH24MISS format&gt;.zip<br>Where file prefix value is "SSEQ". The zip file can contain one or more files from same or different stores. The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>SSEQ_20180129133250.zip<br>SSEQ_20180129133250.zip.complete<br>**Filename Inside Zip**<br>&lt;file prefix&gt;_&lt;date in YYYYMMDDHH24MISS format&gt;_&lt;loc id&gt;.dat<br>Where file prefix value is "SSEQ" and loc id is the store identifier. This allows file to be unique for every upload.<br>Example:<br>SSEQ_20180129133250_1111.dat |

**Table P-1    (Cont.) File Upload Directory and Filename Format**

| Import File Name | sFTP Directory | Filename Format |
|---|---|---|
| Third Party Stock Count Parser Import File | User logs into the SFTP server, and transfer data files to the directory: /{sftp user}/ from_thirdparty | **Zip Filename Format**<br><file prefix>_<date in YYYYMMDDHH24MISS format>.zip<br>Where file prefix value is STK. The zip file can contain one or more .date files (for example, STK_20180129133250_1111.dat).<br>The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>STK_20180129133250.zip<br>STK_20180129133250.zip.complete<br>**Filename Inside Zip**<br><file prefix>_<date in YYYYMMDDHH24MISS format>_<loc id>.dat<br>Where file prefix value is STK and loc id is the store identifier. This allows file to be unique for every upload.<br>Example:<br>STK_20180129133250_1111.dat |
| Warehouse Stock Import File | User logs into the SFTP server, and transfer data files to the directory: /{sftp user}/ from_thirdparty | **Zip Filename Format**<br><file prefix>_<date in YYYYMMDDHH24MISS format>.zip<br>Where file prefix value is "WHIMS". The zip file can contain one or more files. The complete file needs to be added for zip file for job to pick it for processing.<br>Example:<br>WHIMS_20180129133250.zip<br>WHIMS_20180129133250.zip.complete<br>**Filename Inside Zip**<br><file prefix>_<date in YYYYMMDDHH24MISS format>.dat<br>Where file prefix value is "WHIMS". The prefix and date combination allows file to be unique for every upload.<br>Example:<br>WHIMS_20180129133250_1111.dat |

# File Download Directory and Filename Format

**Table P-2    File Download Directory and Filename Format**

| Export File Name | sFTP Directory | Filename Format |
|---|---|---|
| Inventory Extract | User logs into the SFTP server, and download data files from directory: /{sftp user}/ EXPORT/ to_thirdparty | <file prefix>_<store Id>_<extract date in yyyyMMddHHmmss>.TXT<br>Where file prefix value is PRODUCT_LOCATION_INV, store Id is the store identifier.<br>Example:<br>PRODUCT_LOCATION_INV_1111_20180129133250.TXT<br>PRODUCT_LOCATION_INV_1111_20180129133250.TXT.complete |

**Table P-2    (Cont.) File Download Directory and Filename Format**

| Export File Name | sFTP Directory | Filename Format |
| --- | --- | --- |
| Stock Count Export File | When Stock count is unit and amount and authorization succeeded, the stock count authorize process attempts to create an export file.<br><br>By default, SIOCS system stock count export file is sent to MFCS sFTP server when sftp push is enabled, and stock count export system configuration is as /<mfcsFtpUser><br><br>From where, MFCS batch will process the stock count file. | **Zip Filename Format**<br><br><file prefix>_<store id>_<stock count schedule date in yyyyMMddHHmmss> _<export date in yyyyMMddHHmmss>.zip<br><br>Where file prefix value is STK_. The zip file can contain one or more .date files.<br><br>The complete file is added for job to pick it for processing.<br><br>Example:<br><br>STK _1111_20180128133250_20180129133250.zip<br><br>STK_1111_20180128133250_20180129133250.zip.complete<br><br>**Filename Inside Zip**<br><br><file prefix>_<store id>_<stock count schedule date in yyyyMMddHHmmss> _<export date in yyyyMMddHHmmss>.DAT<br><br>The complete file is added for job to pick it for processing.<br><br>Example:<br><br>STK _1111_20180128133250_20180129133250.dat |