Oracle® Retail Merchandising Conversion

Implementation Guide

Release 22.1.401.0
F71431-01
November 2022

Oracle Retail Merchandising Conversion Implementation Guide, Release 22.1.401.0

F71431-01

Contents

Send Us Your Comments

Oracle® Retail Merchandising Conversion Implementation Guide, Release 22.1.401.0

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

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

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

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

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

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

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

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

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

Preface

This guide provides information about data conversion for the Oracle Retail Merchandising Suite of applications - primarily for the Merchandising Foundation Cloud Service (or Merchandising) and the Pricing Cloud Service (or Pricing). It also describes the tools available to assist in conversion, including the Data Conversion application. This guide covers the functional scope of conversion, as well as the overall data migration approach and details the tools available. It also details the screens and covers content from a user manual perspective for the Data Conversion Application.

The approach detailed here is suggestive and can be changed based on your business processes, environments, and delivery phases.

Audience

The implementation guide is intended for the Oracle Retail Merchandising Operations Management applications integrators and implementation staff, as well as the retailer's IT personnel.

Documentation Accessibility

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

Access to Oracle Support

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

Customer Support

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

https://support.oracle.com

When contacting Customer Support, please provide the following:

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

- Screen shots of each step you take

Improved Process for Oracle Retail Documentation Corrections

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

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

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

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

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

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

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

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

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

Conventions

The following text conventions are used in this document:

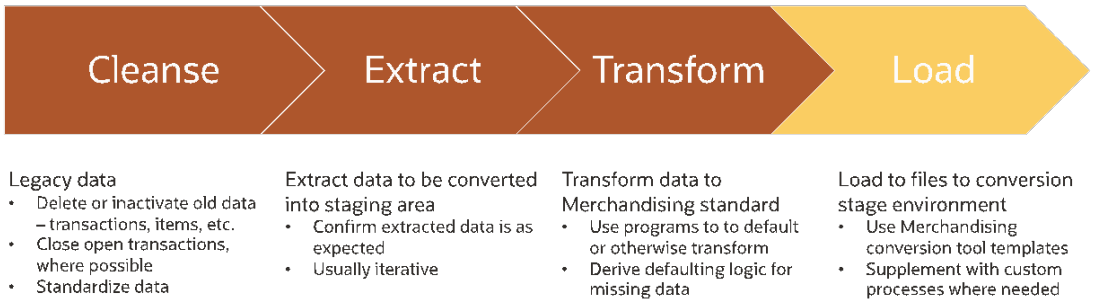
Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates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

1 Introduction

Converting data into Merchandising is a critical step to ensuring a successful implementation of the solution. Depending on your overall approach to the implementation, you may be phasing the solutions in over time - for example, starting with foundation data and then later going live with purchase orders and inventory - or you may be implementing the whole solution at once. This document will speak to the approach of a full end to end implementation of the solution, sometimes referred to as a "big bang" conversion.

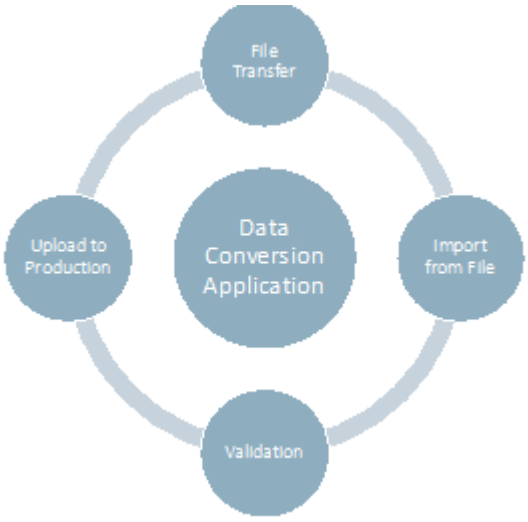
However, if you are performing a phased conversion, there are conversion APIs designed to enable the subsequent phase conversion of certain entities in the production environment. See Appendix: Conversion APIs"Appendix: Conversion APIs" for more information about how to load the subsequent phase data by effectively appending it to the live production data. Also, see the *Phased Data Conversation Reference Paper*, which discusses the approaches that are currently available for Merchandising cloud service customer for a phased rollout.

At a high level, conversions go through the following steps:



The first three steps occur in your legacy solutions. The last step will be to load the cleansed, extracted, and transformed data into the Merchandising solutions.

The vast majority of this migration of data from legacy applications to Merchandising can be achieved through the Data Conversion application. It works by consuming data files from your legacy solutions and loading them into Merchandising tables. From there, data integrity checks and business validations will be invoked to ensure the correctness of data being migrated. The tool contains screens to guide you through the conversion process. Any errors in the file data are reported at a detailed level to enable you to iteratively correct and re-run through the conversion process. The final converted data will then be exported to your production environment.



When your environments for Merchandising are provisioned, the Data Conversion Application will be part of the installation in non-production environments. It will be installed in the same database as Merchandising, but in a separate schema.

Before this occurs or in parallel to the provisioning process, you will need to start analyzing your legacy data to determine how it maps to the Merchandising data model in order to execute the conversion. This includes looking at data cleansing, extract procedures, and transformation logic. To support this process, two tools are provided as part of this application - data templates and an Appendix: Offline File Validator tool. The templates provide detail on the structure of the data files, whereas the offline validator tool can allow you to start pre-validating your transformed data in your legacy environment. These will be described in more detail below.

Then, once you have extracted and transformed your data into the templates, they can be transferred to the conversion environment and loaded into the tool to be validated. The last step, once all conversion related validations and testing is complete, is to load the data into production.

Roles and Responsibilities

The table below summarizes the roles and responsibilities of the customer and their system integration partner and the roles and responsibilities of the Oracle Cloud Operations team as it relates to conversion.

Customer and System Integrator Partners	Oracle Cloud Operations
Determine legacy data to be converted	Provide the Merchandising and Data Conversion Application in development
Plan and design conversion strategy – phased by function, big bang, other	Work with customers and system integrators to confirm plans for data conversion – final production cutover
Cleanse legacy data and determine mapping to Merchandising tables	
Build extract and transformation programs for legacy data	
Import legacy data files via SFTP and utilize the Data Conversion Application to practice data conversion – determine timings and data issues that require correcting	
Validate data in Merchandising development environment after each trial conversion run	
Execute final conversion on cutover day	Export converted data in development instance and import into production database (lift and shift)
Validate data converted into production instance	

2

Data Entities

Base Seeding and Manual Configuration

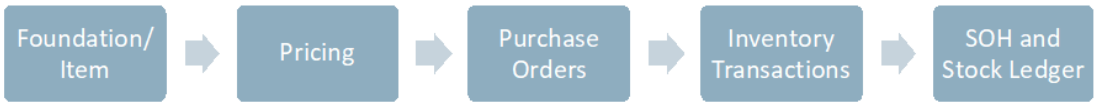
Prior to starting the conversion activities outlined in this document, it is expected that your provisioned Merchandising environment will have the base seeding scripts run, which load core foundational data like valid currency and country codes. Some of these data entities can be tailored for your business, such as calendar, codes, and system options. It is expected this will be done as needed pre-conversion such that any dependent data is configured properly to support the conversion activities. See the Oracle Retail Merchandising Implementation Guide for more details on this configuration.

Functional Conversion

For tables that will be converted, dedicated processes used to load files of data provided by you based on cleansed data from your legacy systems are outlined in body of this document. These programs will validate data and load it into the Merchandising and Pricing tables. It will also have methods to validate and correct data errors as part of the load process.

For certain tables that are impacted as part of the conversion, there will be calculations done as part of the conversion to calculate initial values based on the converted tables.

When converting data into Merchandising, it is expected that the conversion will follow the general flow below, with the vast majority of the data converted falling into the Foundation and Item category.



Foundation Data and Item

Regardless of whether you are planning a "big bang" type conversion or planning to rollout in phases by function, you will need to start with foundation data. The data that can be converted as part of foundation data is listed below and broken into categories to help with understanding how you may want to consider converting your data. The details on what is needed to be included in conversion, are included as part of the templates that are downloaded and discussed more in the "Download Templates" section below. However, some notes are also included with the foundational entities below to add some additional context to help you determine if you need to convert data into these tables. See also the *Merchandising Implementation Guide* for more details on some of these foundational entities.

One other general note: across all entities you will see the inclusion of tables with a _TL extension. These tables are used to house translations of the description fields on the main

table. In general, if your Merchandising users all use a single language, then these tables do not need to be loaded as part of the conversion.

However, there are a couple exceptions where the description is not held on the main table and therefore will need to be loaded with the primary language as well. These exceptions are called out below.

Core Foundation

This is data that is required by nearly every Merchandising implementation, with a few exceptions. All the data in this category should be included in your plan or have a rationale for why it is not needed.

Functional Area	Notes
Banners/Channels	Channel is a mandatory attribute for stores and optional warehouses. A channel is a child of a banner, so both entities are needed prior to store conversion.
Transfer Zones	Transfer zones are required for all stores; if you do not wish to leverage this functionality, you can create a single zone to be associated with all stores.
Diff Types, IDs and Groups	Usually this is only required for items that you want to manage at a "parent" or style level that are differentiated by color or size and are most common for soft-line items. But, they can also be used in grocery or hard-lines for flavor, color, or size as well, if you want to manage attribution and pricing at a parent level.
Freight and Payment Terms	These usually are mastered in your financials system, but also need to be set up in Merchandising. The IDs used for the converted terms of both types should match those used in your financial system to ensure that the details on the purchase orders and invoices sent for payment from Merchandising and Invoice Matching are recognized. Also, if using the Retail Financials Integration (RFI) for integration with PeopleSoft, Oracle Financials, or Cloud Financials, there is a cross-reference that would also need to be maintained and needs to be considered in conversion. See the Merchandising Implementation Guide section on Financials Prerequisites for more details. Additionally, for both types of terms, the description for the primary language is on the translation table not the main entity table, so at least one record must be added in the translation table for your primary language for each term added.
VAT	Value Added Tax, or VAT, is required to be set up if you have configured Merchandising to run in the Simple VAT tax mode. This applies for all regions in the world[1] other than retailers operating in the US only, where VAT is not applicable. Even for US implementations, you may decide to implement using Simple VAT to support future expansion into other regions. In this case, at least one VAT code and region must be defined. US stores and warehouses can be defined as exempt. Also, if using the Retail Financials Integration (RFI) for integration with PeopleSoft, Oracle Financials, or Cloud Financials, there is a cross-reference that would also need to be maintained and needs to be considered in conversion. See the Merchandising Implementation Guide section on Financials Prerequisites for more details.

Functional Area	Notes
Transfer Entity	If you have configured the Intercompany Transfer Basis system option to be Transfer Entity, then this is required to be associated with all stores, virtual warehouses, and external finishers. Otherwise it is not required.
Org Units	All Merchandising implementations will require at least one org unit to be configured. These are usually determined by your financials solution and the codes used in Merchandising should be coordinated with financials for conversion.
General Ledger Setup	The General Ledger setup contains key information by set of books used in the GL mapping. One record should be added per org unit to ensure proper mapping to the chart of accounts in your GL.
General Ledger Account	Supports integration with the general ledger once the stock ledger is initialized. It will hold all the account related information. Set of books in the load file are validated against General Ledger Setup.
General Ledger Cross Reference	This cross-reference supports mapping merchandise hierarchy levels, locations, and transaction codes from Merchandising with the appropriate general ledger accounts within the financial system. This needs to be defined in order to correctly map financial transactions in Merchandising to financials at the end of a day or month. If using the Retail Financials Integration (RFI) for integration with PeopleSoft, Oracle Financials, or Cloud Financials, then the set of books in the load file will be validated against General Ledger Account.
Currency Rates	Currency rates are generally sourced from an external party and integrated into both Merchandising and your financials solution. For conversion, at least one active rate will be needed for each currency that is relevant for your implementation, including stores, warehouses, partners, and suppliers. This includes a record for the primary currency you configured in Merchandising as part of the install options. For that currency you should use a rate of 1.

[1] Brazilian tax requirements are not supported by this functionality.

Optional Foundation

This is data that is foundational in nature but is not systematically required and may not be needed for your implementation. For example, seasons and phases tend to be used more for soft-line retailers, and less so for grocers and hard-lines. This information can still be setup later if not done at conversion, if you decide to leverage some of the functionality.

Functional Area	Notes
Cost Components	Only applicable if you have indicated that you are using estimated landed cost in Merchandising as part of the system options configuration. But, even if using ELC, these are not required for initial conversion.
HTS Setup and Definition	Only applicable if Import Management functionality is being used, as indicated by the installation options that you will have defined for Merchandising.
Seasons/Phases	If loading this data, both seasons and phases should be added.
Outside Locations	Generally, only required if you are using estimated landed cost or import management functionality in Merchandising, but may be used for other purposes as well.

Functional Area	Notes
Buyers/Merchants	Optionally can be associated with levels of the merchandise hierarchy.
Calculation Value Bases	Only applicable if you have indicated that you are using estimated landed cost in Merchandising as part of the system options configuration. But, even if using ELC, these are not required for initial conversion.
Freight Types and Sizes	Only applicable if Import Management functionality is being used, as indicated by the installation options that you will have defined for Merchandising. Additionally, for both types and sizes, the description for the primary language is on the translation table not the main entity table, so at least one record must be added in the translation table for your primary language for each type and size added.
Ticket Types	Optionally can be associated with items.
Brands	Optionally can be associated with items.
Diff Ratios	Used for purchase order distribution in Merchandising - creates a basic diff profile that works best for sizes.
Diff Ranges	Can be used to designate a subset of a diff group or a matrix of diffs in two diff groups. Used in purchase order distribution.
Country Attributes	These are really only required for Brazil implementations. For non-Brazil implementations, this is optional.
Inventory Adjustment Reasons, Statuses, Codes	For all three of these entities, the description for the primary language is on the translation table not the main entity table, so at least one record must be added in the translation table for your primary language for each reason, status, and code added, respectively.
Custom Flex Attributes	If you have configured any custom flex attributes for diff types, ELC components and VAT codes then these can be converted as well. Make sure that the attributes are configured and active prior to conversion. For more information on this functionality, see the *Customization and Extension Guide*.

Hierarchies

This data is required prior to setting up any new items or locations. It also contains some optional components, which are highlighted below.

Functional Area	Notes
Company	Company is seeded during installation and, if required, can be updated during data conversion. Only one company should be loaded; company should always have the ID 1.
Division	Top level of the merchandise hierarchy under company; required.
Groups	Level below division in the merchandise hierarchy; required.
Departments	Level below group in the merchandise hierarchy; required. For retail accounting departments, the Markup % of Cost (BUD_INT) should be used as the initial cumulative mark-on % when initializing stock ledger valuation.

Functional Area	Notes
Classes	Level below department; required. There are two IDs on this table. Class (CLASS) is the "display ID" for the class that is seen in Merchandising screens. It is unique for a specific department. Class ID (CLASS_ID) on this table is not displayed in Merchandising screens, but is a unique ID across all departments to identify this class. It is used primarily in integration.
Subclasses	Level below class; required. There are two IDs on this table. Subclass (SUBCLASS) is the "display ID" for the subclass that is seen in Merchandising screens. It is unique for a specific department/class. Subclass ID (SUBCLASS_ID) on this table is not displayed in Merchandising screens, but is a unique ID across all departments to identify this subclass. It is used primarily in integration.
Chain	Top level of the organizational hierarchy below company; required.
Area	Level below chain in the organizational hierarchy; required.
Region	Level below area in the organizational hierarchy; required.
District	Level below region in the organizational hierarchy; required.
Stores and Warehouses	For stores and warehouses, there are multiple templates that can be used, depending on how you plan to configure your locations. If you plan to define a default warehouse for your stores, then you'll want to use the Warehouse entity for loading your warehouses first and then load your stores using the Store Add with Default WH entity. This will make sure when the business validations are run that they are processed in the correct order. For the first store loaded, you will need to leave the pricing store value null, since the system will attempt to validate the value included. It is not required for the first store created. The store can be added to the correct price zones as part of the Pricing Foundation conversion described below. For warehouses, if you plan to default in a pricing store for the virtual warehouses, then you'll want to add these after the stores using the Warehouse with Pricing Store entity.
Optional	
UDAs	Defaults can also be defined as part of the conversion for departments, classes, and subclasses to either make the UDAs you define mandatory or default a value for all new items created in the hierarchy.
Transit Times	Used for replenishment and in the Allocation Cloud Service; not needed if not implementing those functions
Store Format	Optional attribute of the store.
Warehouse/ Department	Only needed if implementing Investment Buy functionality in Merchandising
Franchise Customers	Only needed if you have stores that have franchise locations and plan to use the franchise functionality in Merchandising. If so, ensure that the franchise system option is set accordingly.

Functional Area	Notes
Cost Zone Groups, Zones, and Zone Locations	If you have indicated that you will use Estimated Landed Cost functionality as part of the system options configuration for Merchandising, then cost zone groups will be required for all new items created, and all stores and warehouses will be associated to cost zones in each group defined. Zone groups and zones are used to manage expenses associated with purchasing product and moving it from a lading port to your locations. Two zone groups are added during the installation by default and should be accounted for as part of your conversion, and more can be added, if needed. At a minimum, zones and locations should be added for the seeded groups and locations should be associated with the zones as part of the conversion. See the Merchandising Implementation Guide section on Estimated Landed Cost for more on the cost zone groups seeded as part of the solution.
Department Up-charges	Only applicable if you have indicated that you are using estimated landed cost in Merchandising as part of the system options configuration, which also enables the cost components used to setup up-charges. But, even if using ELC, these are not required for initial conversion.
Custom Flex Attributes	If you have configured any custom flex attributes for the merchandise or organizational hierarchy, including CFAS attributes for associated addresses, then these can be converted as well. Make sure that the attributes are configured and active prior to conversion. For more information on this functionality, see the *Customization and Extension Guide*.

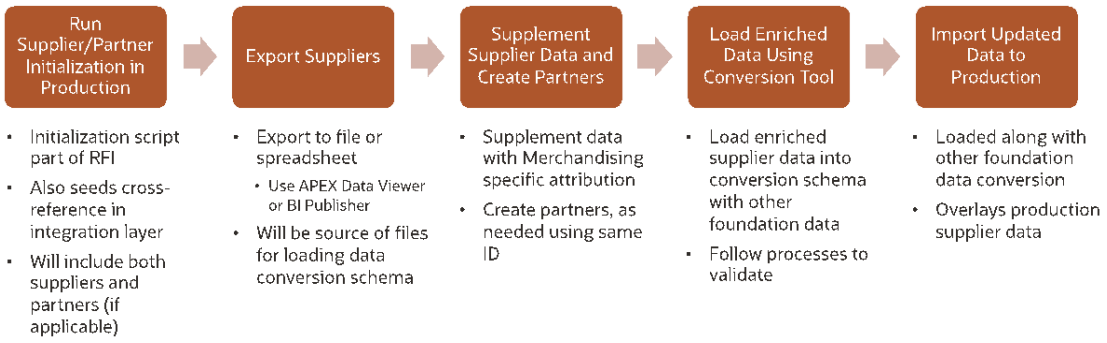
Pricing Foundation

This is required prior to creating items, but after setting up locations, as the zones are used for initial pricing. This should be setup regardless of whether you are implementing the full Pricing Cloud Service or only simplified mode.

Functional Area	Notes
Zone Group	Create one or more to define how locations will be grouped together for pricing. Needed for initial pricing for your items and for future price changes and clearances.
Initial Price Zone Definitions	Every department, class, or subclass that you have converted into Merchandising should have an initial price zone definition that defines the zone group used for new items in that department and the details on how markup is calculated. Additionally, rounding rules can be included as part of this definition, if applicable. If you choose to setup rounding rules for the initial price zone definition, this must be done in the Pricing screens manually. Rounding rules can also be added later as well.
Zones and Zone/ Locations	Define the zones you want for the zone groups created as described above. And then add rows for your locations. Not all locations need to be in every zone group. Also, warehouses should only be added to zones if you have configured Pricing to manage price for warehouses (via the Recognize Warehouses as Locations system option).

Suppliers and Partners

If integrating with a financials solution, a slightly different process for converting is recommended based on the dependency for financials as part of the creation process. The diagram below outlines the recommended process.



It is recommended you start with setting up these parties in financials and integrate them to the production version of Merchandising as a way to convert the financials owned data into the solution[1]. Oracle Retail Financial Integration (RFI) has a seeding process that is used for this purpose. This also means that any dependent data for suppliers, like org units, freight terms, and payment terms, should be loaded into production. This could be done using either spreadsheet upload or the data conversion tool.

Loading the suppliers and partners from financials into production directly will generate the cross references and Merchandising IDs needed for these entities, however there still will be some Merchandising specific attributes you will need to configure, as well as you will need to copy the entities that are really partners to their own table.

For suppliers, the recommended approach would be to use the APEX Data Viewer functionality[2] to query the supplier table to extract the data that was loaded from financials and use this data as a basis to build the conversion data dat files for both suppliers and supplier sites. The same IDs should be used in conversion to reload these suppliers, but they can have the retail specific data added as well.

For partners, the recommended approach also is to extract the data from the production supplier table and use the data to convert into the partner table. The same IDs should also be loaded into the partner table in Merchandising, along with the attribution from Financials and any new attribution that is retail specific as part of the conversion.

For both of these entities, when the converted data is promoted to production, it will overwrite the data in the Merchandising tables, but the key data setup in the cross references would still be accurate and now the data would include the retail specific attributes as well.

1 The RFI Implementation Guide will have details on how to query the suppliers to integration to Merchandising from financials for this initial load.

2 Application Express (APEX) Data Viewer is a tool that allows you to query and update data in your pre-production SaaS environment. Details on how to access and configure users for this environment can be found in the Oracle Retail Merchandising Foundation Administration Guide.

Functional Area	Notes
Partner	This includes both the partner and address information.
Supplier	This includes the suppliers, sites, and addresses. Both suppliers and supplier sites are setup here and should leverage information that you get from financials, if you are using the RFI integration described above. Supplier sites would have the supplier ID in the supplier parent column.
Partner/Org Unit	One row should exist in this table for every supplier site and partner. A supplier site and partner can only belong to one org unit.
Optional	
Import Attributes	Only required if you are using import management in Merchandising.
Inventory Management Parameters	This is primarily used for configuring defaults for suppliers for replenishment purposes. But, also has some PO and item defaults that you may want to configure.
Bracket Costing	Only required if your supplier is flagged as supporting bracket costing.
Delivery Schedules	This is used for replenishment only.
Custom Flex Attributes	If you have configured any custom flex attributes for supplier or partner, including CFAS attributes for associated addresses, then these can be converted as well. Make sure that the attributes are configured and active prior to conversion. For more information on this functionality, see the *Customization and Extension Guide*.

Core Item

This section outlines the data that is either required for all items or needed for the vast majority of them that you will be converting. All the data in this category should be included in your plan or have a rationale for why it is not needed. If you are using parent/child item structures relationships setup at the parent level will not automatically default to the child level when converting using the Data Conversion application. So, if they are applicable to both levels, they will need to be added to both the parent and the child. Some of the attributes below are not applicable for sub-transaction items (such as barcodes).

Functional Area	Notes
Item	The required data for item includes the item master table as well as the following tables: Item/supplier - for orderable items Item/supplier/origin country - for orderable items Item/supplier/manufacturing country - only required if the Merchandising system option for HTS tracking level is set to Manufacturing (the default is Sourcing if not otherwise configured) Item/VAT - It is defaulted from Department/VAT during item conversion if Simple VAT is configured as the default tax type. Any additional records or changes in VAT rates can be loaded, if desired. Only approved, active items will be converted. It is highly recommended that you do not load all items in one large file, but instead consider building separate files (such as, by department, class, or subclass) for better management of data volumes during the validation stages.

Functional Area	Notes
Price History	Load initial record with a tran type of 0 on this table for the item with a location of 0 using the Price Hist entity. Location-level records added in the item/location ranging are described below.
Item Zone Price	The zone-level pricing details will be used by the item/location ranging process described below to default the regular retail price for the item/location combinations if the retail price is not provided in the input file. If any retails need to be corrected, or if any of the items are on clearance, the selling retail price at the item/location combinations will need to be updated after converting zone pricing and item/location ranging. See also the "Pricing" section below for details on seeding clearance events and future retail.
Item/Location Ranging	To associate stores and warehouses with your items as part of conversion, use the Item Ranging entity. Pricing for your location can be provided in the input file during the item/location ranging process. If unit retail is not specified, it will default based on the zone pricing. This will take care of inserting records for the following entities in Merchandising: Item Location (ITEM_LOC) Item Location Stock on Hand (ITEM_LOC_SOH)[1] Item Supplier Country Location (ITEM_SUPP_COUNTRY_LOC) Future Cost (FUTURE_COST) Price History (PRICE_HIST) for tran type 0 It is assumed that only active item/location combinations will be converted.
Update Item/Location	Pricing for your location will be established by the item/location ranging process as described above. As needed, you can use this entity to update those values. This also includes setting of the clearance flag and selling unit retail for items on clearance at the time of conversion, as needed. If you update the item/location level price you will also want to add a row in the Price History table for the new price as well. See the "Pricing" section below for more on initializing items on clearance within the Pricing Cloud Service.

1 All stock on hand values will be inserted as zeros. The on hands, along with any adjustments to average cost and weight can be updated later in the process as described in the section on Stock on Hand and Stock Ledger.

Optional Item

This is additional data related to items that may or may not be relevant for your implementation. It can also be added later using the basic Merchandising functionality and mass update capabilities if there is not anything to convert initially. If you are using parent/child item structures relationships setup at the parent level will not automatically default to the child level when converting using the Data Conversion application. So, if they are applicable to both levels, they will need to be added to both the parent and the child. Most of the attributes below are not applicable for sub-transaction items (such as barcodes).

Functional Area	Notes
Supplier/Country Dimensions	This is not required for most items, but should be defined for catch weight items for cases and eaches. If loading this data, both primary and non-primary supplier/country should be loaded.

Functional Area	Notes
Pack Item Details	There are two tables that are required for any converted pack items - one is the basic pack details (PACKITEM) and the other goes one level deeper in detail, if the pack contains any inner packs or uses a pack template (PACKITEM_BREAKOUT). Both tables are required for all packs.
UDAs	If any UDA defaults were converted for departments, classes, and/or subclasses that make certain UDAs required at the item level, make sure that you convert values for those UDAs for all items in the hierarchy as part of your conversion.
Seasons	If loading this data, both seasons and phases should be added. Requires conversion of season information as described above.
Images	At least one image link must be flagged as primary, if added.
Translations	Translations can be added for item master level records and for item images for languages other than primary, if desired.
Expenses	Both header and detail are required if you choose to load zone and country level expenses associated with items.
HTS	Only applicable if Import Management functionality is configured on. Requires conversion of HTS information as described above.
Custom Flex Attributes	If you have configured any custom flex attributes for item, item/supplier, item/supplier/country, item/supplier/country/location, or item/location then these can be converted as well. Make sure that the attributes are configured and active in advance of conversion. For more information on this functionality, see the *Customization and Extension Guide*.
Upcharges	Only needed if planning to use Estimated Landed Cost functionality, which enables the cost components used to setup up-charges.
Item Location Traits	Not required, but there are some key attributes on this table you may want to set for your implementation that are used to drive customer ordering - including whether or not an item can be backordered.
Tickets	Requires conversion of ticket types as described above.
Related Items	Both header and detail information are required, if you choose to use this functionality.
Pack Templates	Only used in the creation of some pack types.

Other

This section outlines some other entities that are related foundation and item that may be something you consider for conversion. Everything in this section is optional.

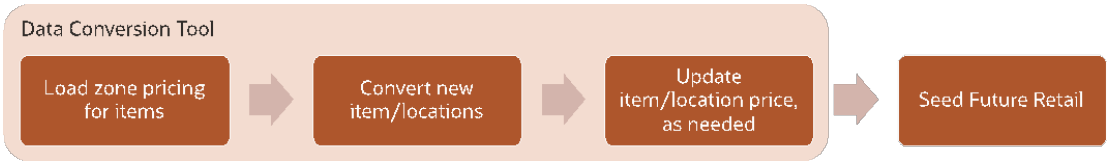
Functional Area	Notes
Substitute Items	Only applicable if you are using Replenishment functionality in Merchandising.
Item Forecasts	Only applicable if you have a forecasting solution and wish to initialize forecasting in conversion. Data used for replenishment and allocation.
Master Replenishment Attributes	Only applicable if you are using Replenishment functionality in Merchandising.
Cost Changes	If you have any pending cost changes that are due to go into effect in your legacy systems that you would like to convert into Merchandising, the scripts in this section will support that conversion. This is optional.

Pricing

In addition to the pricing related data called out above, there are a couple other components of pricing that you will need to consider for your conversion plan regardless of whether you are planning to use the full Pricing Cloud Service functionality, or operate in simplified mode.

Future Retail

First, you will need to seed the Future Retail table to establish the initial retail prices that were included as part of your item/location conversion.



The diagram above shows how the steps that were performed in the data conversion tool are used as a basis for this process. Seeding the future retail values will be done by executing the `DC_RPM_SEED_FUTURE_RETAIL` task from the System Administration screen in the data conversion tool. It is required that this be run after your foundation data and item data is loaded, but before your conversion of stock on hand and stock ledger. This will establish seed records in Pricing for all active item/locations in the system at the time the conversion is run. The task is intended to run a **single time**, and should be run after the Item Location conversion has been validated. The task looks directly at the `ITEM_LOC` table to determine the records to be processed.

Clearance Markdowns

The other conversion for pricing to consider is any items that are currently on clearance. If you will have items that are actively on clearance when you are converting into Merchandising and Pricing, then you will need to create clearance markdowns in Pricing to initialize the markdown to allow it to be used for a future reset of clearance price, or for any inheritance for new item/location relationships. The conversion of active clearances is based on converting data from an input file using the `RPM_CLEARANCE` entity in the Data Conversion application. See the "Key Data File Assumptions" section in "Prepare for Conversion" for more details on file layout and assumptions.

Column	Required?	Data Type	Notes
Item	Yes	Char (25)	Must be a valid sellable, approved item already converted into Merchandising. It can be a parent or transaction level item.
Diff ID	No	Char (10)	If this is included, the item must contain a parent item ID and the diff must be a valid diff for one or more child items for that parent.
Location Type	Yes	Number (1)	Valid values are 0 (store), 1 (zone), or 2 (warehouse).

Column	Required?	Data Type	Notes
Location	Yes	Number (10)	Must be a valid location ID for the location type specified.
Clearance Retail	Yes	Number (20,4)	Must be greater than zero. Assumed to be in the currency of the location.
Grouping ID	No	Number (15)	Optional; could be used to group certain converted markdown records together. If not assigned, this will be systematically assigned when the processing occurs.
Clearance Group Description	No	Char (250)	Optional; could be used to describe the grouping of certain converted markdown records together. If not assigned, this will be systematically assigned when the processing occurs.
Markdown Value	Conditional	Char (6)	If the system option to require a markdown number is set to Yes, then this should be included for data consistency. Valid markdown value codes are defined using code type MKDN in Merchandising.
Reason	No	Char (6)	If you choose to include a reason code here, it needs to be a valid reason code. Valid clearance reason codes are defined using code type CMRC in Merchandising.

This process will validate all the required fields in the clearance file are present and that all the data included is valid. It will then insert records into the markdown tables in Pricing based on this data as executed markdowns with an effective date of the current system date. It will also insert a record into the Price History tables to record the markdown details and add future retail records related to the markdowns.

Prerequisites

- Configure system options for Pricing; unlike Merchandising, there aren't any seeded values for Pricing, so the system options should be reviewed and setup using the Pricing screen.
- Complete the required Pricing and Item conversion described above to load zone groups, zones, items, and locations fully into the Merchandising tables, not just held in the conversion staging tables.
- Run the future retail seeding before you do any clearance conversion.

Key Assumptions

- The future retail data that is seeded as part of the process described above will use the current date -1.
- The effective date for all converted markdowns will be the **current system virtual date**. Converted markdowns must be unique by item/location/effective date.
- All converted clearance markdowns will be created in Executed status.
- The clearance conversion process is intended to be run a **single time** and assumes no data will be present on the Pricing Clearance table when the program is run.

- The clearance conversion process does not update the clearance flag or selling unit retail on the ITEM_LOC table with the clearance prices. The clearance retail should be updated for the item/location combination using the ITEM_LOC entity in the conversion tool as described above.
- Price changes are not supported as part of this conversion process. It is assumed all price changes will be executed prior to your cutover and any future price changes will be entered in Pricing after cutover. The spreadsheet upload or bulk loads could be used to assist with this process, if there are too many to re-create manually.
- Promotions are not supported as part of this conversion process. It is assumed that promotions will be halted during the conversion window for conversion and new future dated promotions will be setup in Pricing after cutover. If any promotions will be ongoing during the conversion window, then they can be created with a start date of the current date in order to continue in the new solution. At this time, the only method of creating promotions in Pricing is using the screens.

Purchase Orders

Any open purchase orders that cannot be closed prior to conversion will be able to be converted using this process. This conversion includes the conversion of the order and all line items, as well as related data about the order, such as expenses and assessments. If there is a letter of credit associated with the order, this relationship can also be converted. However, it should be noted that the letters of credit themselves are not supported in the Data Conversion Application (see assumptions below). This will be done using the Create Purchase Orders entity in the Data Conversion Application.

Additionally, if the order has been partially shipped and/or partially received, entities also exist in the Data Conversion Application to convert these as well, using the Ship Purchase Orders and Receive Purchase Orders functions, respectively. The Ship Purchase Orders scripts will create ASN records for the order, while the Receive Purchase Orders function will update the shipment and purchase orders with the receipt details.

There is also a Close Purchase Orders option that will allow for any orders that need to be converted as closed, such as if there are unpaid invoices that have not yet been received, to be updated to closed after the prior steps are completed.

If you have configured any custom flex attributes for Purchase Orders at the header or detail level, then these can be converted as well. Make sure that the attributes are configured and active prior to conversion. For more information on this functionality, see the *Customization and Extension Guide*.

Key Assumptions

- All foundation data needed to support purchase orders and invoices has already been converted in a previous phase of conversion. This includes configuration of the following:
  - Procurement related system options
  - Non-Merch Codes
  - Order Context/PO Types
- PO numbers should be able to be converted from legacy, assuming the legacy number is unique and fits the data requirements of Merchandising. If not, the Vendor Order number could be used to hold the legacy number and a unique PO number created to fit the Merchandising numbering scheme.

- All POs that can be closed in legacy will be closed prior to conversion.
- The following functional areas do not have conversion support at this time: Contracts, Deals, Customs Entry, Letter of Credit, Transportation, Obligations, Actual Landed Cost, Documents, Timelines, Work Orders.
- Any off-invoice deals that exist in legacy should be netted out in the converted PO cost.
- POs received in legacy systems will be reconciled, matched, and paid in legacy. For any open invoices, a process should be run post conversion of purchase orders to load the invoices using the Invoice Matching EDI upload. Invoices will not be converted in this tool.
- Inventory updates that occur during the receipt processing for POs will update inventory; it is assumed that these will be reset during the inventory conversion to the correct values.
- Transactional stock ledger (TRAN_DATA) records created out of the shipping and receiving of purchase orders should be cleared prior to promoting your data to production. This can be done using APEX Data Viewer.
- The relevant item/location records needed to support purchase orders have already been established through the item/location ranging process.

Inventory Transactions

It is expected that inventory transactions will be part of the third phase of conversion, following purchase order conversion. Most inventory transactions are not expected to be converted, as it is expected that transactions in your control, such as RTVs, transfers and allocations are closed in legacy before cutover and that there is a hold on creating new transactions until you have moved over to the new system. The exceptions to this are usually customer orders and some sales history.

Customer Orders

It is likely not possible to put a hold on customer order fulfillment during your cutover to Merchandising. So, in order to support any in process orders you may have, a set of conversion scripts are provided to include the data in Merchandising needed to reserve inventory for the orders.

It is generally assumed that only in process customer orders will be converted into Merchandising and that the history of customer orders for returns purposes would be available in your order management system (OMS).

Create Customer Orders

Customer orders will be created in a number of different ways based on the details you include in the Create Customer Orders template. These are the types supported in this conversion and in Merchandising:

1. Orders to be sourced and fulfilled from the same store (via pickup or shipment from store)
2. Orders to be sourced and fulfilled from the warehouse (shipping to the customer)
3. Orders to be sourced and fulfilled from the supplier (shipping to the customer)

4. Orders to be sourced from a warehouse or store to be fulfilled from another store (via pickup or shipment from the store)
5. Orders to be sourced from a supplier to be fulfilled from a store (via pickup or shipment from the store)

For type 1, only customer order header and detail records are required. For types 2 and 4, a customer order header record is created along with a customer order type of transfer to orchestrate the movement of goods between locations or to move the inventory to a virtual store for sales processing. Types 3 and 5 use a customer order header record, along with a purchase order, to create the details of the order. These additional transactions will be created based on the data you send in the provided templates.

The details below are provided to help you with how to setup the fulfillment orders as part of the conversion for each type.

	Type 1	Types 2 & 4	Types 3 & 5
Customer Order Number	The overall order number - usually the customer facing number		
Fulfillment Order Number	The child order number representing this specific fulfillment location combination and items (at detail level); can be one to many with the customer order number		
Source Location Type		For Type 2 = WH For Type 4 = WH or ST	SU
Source Location ID		The warehouse or store ID where the goods will be shipped to the fulfillment location or the customer	Supplier Site ID where the goods will be shipped to the fulfillment location or the customer
Fulfillment Location Type	S	Type 2 = V Type 4 = S	Type 3 = V Type 5 = S
Fulfillment Location ID	Physical store where the order will be picked up or shipped to the customer	Type 2 = ecommerce store[1] Type 4 = physical store where the order will be picked up or shipped to the customer	Type 3 = ecommerce store Type 5 = physical store where the order will be picked up or shipped to the customer
Order Placed Store	Either the ecommerce store or the physical store locations where the customer placed their order		

1 This should be configured as a non-stockholding store.

Ship and Receive Customer Orders

For customer order types 2-5, if there are any open shipments, or if the order was partially received in legacy, then conversion processes should also be run to convert these shipments and receipts. To do this, utilize the Ship Customer Orders and Receive Customer Orders entity templates.

First, the templates to convert any shipments should be run. If carton data has been received as part of the ASN, then that should also be able to be included in the ASN conversion. This template will support both future and backdated shipments.

Then, for those shipments that were previously received, scripts will be run that will allow receipt quantities at the transfer or PO detail level to be entered (along with dates) to process the receipts.

Key Assumptions

- The customer order number, fulfillment order number, and fulfillment location for type 1 determines a unique order.
- The customer order number, fulfillment order number, and sourcing location for types 2-5 determine a unique order.
- Only in progress customer orders will be converted.
- The conversion of customer order reserved quantities would be part of the stock on hand conversion, not covered by this process.
- When converting customer orders that involve a warehouse, the warehouse used will be a virtual warehouse.
- The customer orders have already been sent to the fulfilling locations (store, warehouse, or supplier). The records in Merchandising are just to finalize the fulfillment of the order.
- The Create Customer Orders process will attempt to validate inventory in Merchandising, if the system option Validate Availability for Customer Orders should be set to unchecked (N) during conversion.
- Inventory updates that occur during the shipment and receipt processing will update inventory; it is assumed that these will be reset during the inventory conversion to the correct values.
- Transactional stock ledger (TRAN_DATA) records created out of the shipping and receiving of customer orders should be cleared prior to promoting your data to production. This can be done using APEX Data Viewer.

Sales History and Warehouse Issues

Merchandising doesn't require sales history or warehouse issues (outbound transfers or allocations from a warehouse) in order to operate, but some retailers have chosen to convert a certain amount of history in order to help support forecasting, replenishment, or allocation requirements when first implementing Merchandising. You can do that by utilizing the Item Location History (ITEM_LOC_HIST) template. Sales and issue history are held by item/location/week.

After conversion of sales history, you will want to schedule the history roll up batches to run to roll up history to the subclass, class, and department levels if you are using the Allocation Cloud Service as well, or if you wish to use this information for reporting purposes. Details on these can be found in the *Oracle Retail Merchandising System Operations Guide, Volume 1*.

Key Assumptions

- VAT History and Daily Sales Discount tables will not be converted, as they are used for reporting purposes only. They will start to accumulate data after go live when sales begin to be processed.
- Both store sales and warehouse issues will be able to be converted.

- The value, gross profit, retail, and average cost columns can be included in the conversion, but as they are not required, they can be left null.

Stock on Hand and Stock Ledger

The final step in conversion is to initialize the stock on hand and stock ledger. This should be done after all the other data described above is loaded. Before this step you should also ensure that you have cleaned up all stock on hand information prior to conversion to ensure that stock levels are accurate and you should validate accurate unit cost and unit retail information has been set in the system as part of the initial item/location conversion.

Non-Sellable Inventory

Before the stock on hand is converted, if there is any non-sellable inventory for item location combinations that you need to convert, it should be done first. The below table outlines the data that is included in the conversion file.

Column	Notes
Inventory Status	This should be from pre-configured non-sellable inventory status types in Merchandising.
Quantity	This should be loaded as positive quantity in standard UOM for the inventory status.

Stock on Hand

To do this, you'll want to export the Item Location Stock on Hand records that were generated in the Item Location Ranging process described above. Then you will need to update those records with the inventory snapshot values from your legacy solution. Values that should be initialized as part of this process include:

Column	Notes
Average Cost	If your cost method is configured as standard cost on Merchandising system options, set this to the unit cost value that you used during the item location ranging conversion. Otherwise, set this to the current weighted average cost of the item at this location.
Stock on Hand	Set this value to the number of units you have on hand that are not part of pack items for this item at this location. This should be inclusive of all reserved quantity, but not include in transit, expected, or backorder.
In Transit	Assumed to be zero unless you have shipped/not received customer orders of type 2 or 4. If so, this should be the total number of units shipped but not received for the receiving store on the customer order transfer.
Pack Component In-Transit	Assumed to be zero even if open customer orders exist, as stores don't track pack inventory.
Pack Component Stock on Hand	Set this value to the total number of units you have on hand that are part of pack items for this item at this location. If the item is a pack and/or the location is a store, it should always be zero.
Transfer Reserved	Assumed to zero unless you have customer orders that have not yet been shipped of types 2 or 4; this would have a value for the shipping store or warehouse only.

Column	Notes
Pack Component Reserved	Assumed to zero unless you have customer orders that have not yet been shipped of types 2 or 4 involving pack items; this would have a value for the shipping warehouse only. If the item is a pack and/or the location is a store, it should always be zero.
Transfer Expected	Assumed to zero unless you have customer orders that have not yet been shipped of types 2 or 4; this would have a value for the receiving store only.
Pack Component Transfer Expected	Assumed to be zero even if open customer orders exist, as stores don't track pack inventory.
Non Sellable Quantity	Set the quantity to the number of non-sellable units across all inventory statuses for this location. This should match the total converted for the item/location on the INV_STATUS_QTY table.
Customer Reserved	Assumed to be zero unless you have customer orders of type 1 that are open.
Customer Backorder	Set the quantity to the number of units on backorder you have for this location. Backorders are assumed to be managed in your OMS, there are no other records in Merchandising other than this quantity.
Pack Component Customer Backorder	Set the quantity to the number of units for a component of a sellable pack at this location that are on backorder. Backorders are assumed to be managed in your OMS, there are no other records in Merchandising other than this quantity. If the item is a pack and/or the location is a store, it should always be zero.
First Received	Optional - could be used for reporting purposes
Last Received	Optional - could be used for reporting purposes
First Sold	Optional - could be used for reporting purposes
Last Sold	Optional - could be used for reporting purposes
Average Weight	Should be seeded for catch weight simple pack items in the warehouse only.
Finisher Average Retail	Only needed for external finisher locations if inventory is present at the finisher.
Finisher Units	Only needed for external finisher locations if inventory is present at the finisher.

Stock Ledger

Once you have created the files based on the information as described above and it has been loaded and validated by the Data Conversion application, execute the `STOCKLEDGER_PROCESSING` task in the System Administration screen to calculate the opening balances for the stock ledger. Stock ledger records will have been seeded based on the conversion of subclasses and locations, but the values will all be zero. This task will calculate an opening inventory balance in terms of local currency for both the month and week (depending on your system option settings) for every subclass/location combination where there is inventory to establish. Details on how to do this can be found in the "Task Execution Engine" section of this document.

Key Assumptions

- RTVs, allocations, and transfers not related to customer orders are not currently supported in the Data Conversion application. Oracle Retail highly recommends that you close all open transactions in your legacy solutions and hold new activity

during the conversion cutover window and then start any new activity once you are live in Merchandising.

- The relevant item/location records have already been established at this phase of conversion through the item/location ranging process. If not, then this should be a pre-conversion step for stock on hand conversion.
- Ensure the `SYSTEM_VARIABLES` and `PERIOD` tables have correct data before loading store and warehouse data. The stock ledger tables records will be inserted based on this and any incorrect data will result in issues during the stock ledger initialization.
- Stock Ledger conversion should be done in conjunction with the beginning of an accounting period in order to establish a full month's worth of transactions after cutover. There will be no historical balances converted.
- A primary currency version of opening or closing stock values in the stock ledger is not recorded in Merchandising and so is not required as part of conversion.
- Half Data records will be added for all subclasses and locations converted as part of the foundation conversion. Additional conversion for this table is not required - it will grow over time.
- Half Data Budget will be added for all departments and locations converted with null values in the calculated columns in the foundation conversion. If you wish to update the cumulative mark-on percent in advance of the stock ledger conversion, to use that value for calculating the cost complement, then this can be done in the Merchandising UI using the spreadsheet upload functionality. For more details on this, see the *Oracle Retail Merchandising Foundation Cloud Service Finance User Guide* section on "Managing Budgets".
- Week Data will only be updated if the system option for Stock Ledger Time Interval is Week (W).
- Last update datetime and ID will be updated automatically as part of the conversion.

Converting Other Data

For the data entities that are not supported by the Data Conversion application, the following can be leveraged to load data, as needed:

- Leverage **spreadsheet uploads** for loading foundational data not supported in the tool. This may be especially helpful if you want to configure some of the data that Merchandising seeds automatically at provisioning, such as countries and states, or in cases where you do not have a data source in legacy that can easily be used for programmatic conversions, like Sales Audit reference fields.
- Leverage the vast **API library** for converting data not supported in the conversion tool, such as transfers or promotions. All the Merchandising cloud solutions have a number of different APIs that could be leveraged via different integration methods - web services, RIB messages, and batch uploads.
- Write **custom scripts** that can be run in the non-production environment that you are using for conversion. This is similar to the approach described above for Stock Ledger seeding. The APEX Data Viewer in non-production environments allows for the running of PL/SQL scripts that insert, update, or delete records in the Merchandising non-production database. It should be noted, however, that you will not be able to create tables or call base package functions as part of this process.

There are also several areas that are excluded from the eventual export from the non-production environment that you are using for data conversion to the production environment

because of dependencies that require them to be set up differently than the data described above.

Sales Audit Totals and Rules

Sales Audit Totals and Rules are normally excluded from the lift and shift process because there are additional dependencies to fully set them up in a new environment. If you have setup and tested totals and rules up in your **non-production** environment and want to have them migrated to **production**, then you will need to migrate them separately using these steps:

1. Export the total and rule configurations from the non-production environment by running SA_RULES_TOTAL_EXTRACT_JOB from POM. This will generate sartexp_<table_name>.dat files.
2. Validate the extract and, as necessary, remove unwanted totals and rules.
3. Log an SR with the Oracle Cloud Operations team to upload the extracts to the production environment. As part of the SR, they will load the extract files into production, and then execute the batch required to generate the procedures used by Sales Audit.

For migrating the totals and rules data from one non-production environment to another non-production environment, you are able to execute the full migration process. This is done by executing SA_RULES_TOTAL_EXTRACT_JOB and SA_RULES_TOTAL_UPLOAD_JOB from POM to extract and upload the data respectively in the appropriate environments, using the extracted file similar to what is described above. If there are existing totals and rules records in the target non-production environment, it is recommended that you first purge them using APEX Data Viewer.

Totals and rules can be migrated in worksheet, submitted, or approved status.

Custom Flex Attributes (CFAS)

To migration CFAS attributes from a pre-production to a production environment, leverage CFAS download/upload spreadsheet feature from the UI. Using this approach, download the CFAS foundation data (that is, group sets, groups, record groups, and/or their labels from the pre-prod environments) and upload this spreadsheet to the production environment. Migrate the CFAS attributes in the similar manner. The last step is to create CFAS view by activating the migrated attributes in the production environment. This will also make the attributes visible from the UI. For more information on this process, see the *Merchandising Customization and Extension Guide*.

Custom Validation Rules

As with Sales Audit Totals and Rules, migration of Custom Validation Rules from a pre-production to a production environment will require you to log an SR to coordinate with the Oracle Cloud Operations team.

Data Filtering

If you plan to configure Merchandising to filter data based on merchandise or the organizational hierarchy and have set up the configuration for your users in non-

production, you can migrate this to your production environment by downloading the configurations into a spreadsheet (Foundation Data > Download Foundation Data) using the spreadsheet templates under template type Data Filtering and then uploading the same into the production environment.

Duty and Privilege Configuration for Roles

Often the users and/or the roles and duties of users differ between non-production environments and production environments. However, if you are using a non-production environment to test out the configurations for each of the roles to be used in Merchandising solutions, you can migrate the configuration between instances by exporting the configuration from the non-production environment and then re-importing them into production. This is performed in the Oracle Retail Application Administrator Console using the following steps:

1. **Download configurations** by accessing Settings > Security > Policy Backups and selecting the Download action.
2. Edit the XML file that results to remove the _PREPROD extension from the user roles, so that they will be valid production role names.
3. Import configurations by accessing Settings > Security > Policy Patching and then selecting Synch with Patch for the latest patch applied in your environment. This will take you to the Base Policies page, where there is a button, Import Custom Policies, that will allow you to import the file created in the download step.

For more on these functions, see the *Merchandising Administration Guide* chapter on Security.

Converting Non-Merchandising Solutions

If you are implementing other Oracle Retail solutions at the same time as your Merchandising implementation, or shortly thereafter, Merchandising can be used to seed the data into those environments. It should be planned that those solutions are converted after Merchandising conversion has been completed.

To convert other solutions, such as store solutions (SIOCS, Xstore), planning (MFPCS, A&IPCS), or omni-channel (OROMS, OROB), Merchandising has a number of bulk data integration (BDI) scripts intended to support this. For Retail Insights (ORMI), you can run the full load integration using RDE (Retail Data Extractor) from Merchandising.

3
Getting Started

As part of the provisioning process, you should receive the links to access the application from the cloud engineering team, along with information on how to upload files into the environment. The link for accessing the tool is like the below, with the Region Name and customer subnamespace being replaced by what is applicable for your environments.

```
https://rex.retail.<Region Name>.ocs.oraclecloud.com/<Customer Subnamespace>/
DataConversion/faces/Home
```

The Merchandising File Transfer service will be used for uploading the Data Conversion input files. While uploading the files, use `dataconversion/incoming` as the object storage (OS) prefix. Details on Merchandising File Transfer service can be found in the *Oracle Retail Merchandising System Operations Guide, Volume 2*.

> **Note:**
>
> The File Transfer Services allow you to manage uploading and downloading files to Object Storage that are processed by the Data Conversion Application. These Object Storage buckets are automatically created when the environment is provisioned. You do not need access to the OCI Console because these buckets can be directly used through the File Transfer Service to upload/download files for data conversion.

To get started with using the Data Conversion application, you will first need to configure users for the roles, duties and privileges of the application. There are two roles by default that have been configured for this application:

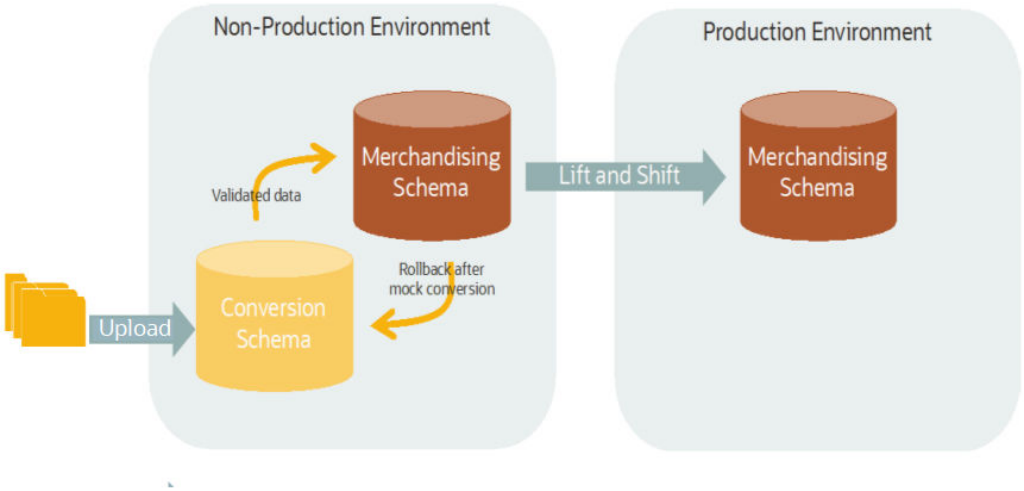
- Data Conversion Operator (DATACONV_OPERATOR_JOB)
- Data Conversion Administrator (DATACONV_ADMIN_JOB)

Duty	Priv	Description
Data Clean-up Duty	Data Clean-up Priv	By default, this is assigned to the administrator role only. This privilege allows for access to the System Administration screen.
Data Inquiry Duty	Data Viewer Priv	This privilege allows for access to the View Uploaded Data screen.
Data Validation Duty	Basic Data Validation Priv	This privilege allows for access to the Data Validation screen.
	Business Data Validation Priv	This privilege allows for access to the Business Validation screen.
Mass Upload Duty	Mass Upload Priv	This privilege allows for access to the Mass Upload screen.
Report Viewer Duty	Report Viewer Priv	This privilege allows for access to the in-context reports.

As with other roles, the data conversion roles are associated with users in the Identity Cloud Service (IDCS)[1] and the duties and privileges assigned to the roles are managed in the Oracle Retail Application Administration Console (ORAAC)[2]. Each of the screens and functions noted above will be described in more detail later in this document.

The diagram below shows, at a high level, how the conversion schema is set up in relation to your non-production environment. Files are imported using the Merchandising File Transfer Service into the conversion schema and then validated data is loaded into the non-production Merchandising schema.

It is expected that you will iterate multiple times, loading data and validating between the conversion schema and the Merchandising schema. Once you have validated, the data is all loaded correctly in the non-production environment, including the running of any custom scripts to load data not supported in the tool, you will coordinate with the Oracle Cloud Operations team to load the data into production, often referred to as "Lift and Shift". The rest of this document will describe in more details how the conversion tool supports this high-level flow of data.



Note:

Unlike other Merchandising applications, the data conversion roles should not be prefixed with `_PREPROD`. The classification is intended to distinguish between pre-production and production environments. The Data Conversion tool is available only for the non-production environment.

Key Assumptions

- Data filtering in Merchandising should be turned off during data conversion activities. It can be re-enabled after conversion is completed.
- All system option configurations in Merchandising and Pricing are updated as appropriate for your implementation prior to starting the conversion. You can perform this via the UI or by updating directly in the non-production environment schema using APEX Data Viewer.

[1] For more information on managing users and roles in IDCS, see Managing Users, User Accounts, and Roles

[2] For more information on configuring duties and privileges for roles, see the *Merchandising Administration Guide*.

- Because of changes that may occur to foundation data structures during a major update, it is recommended that you clear the data conversion environment using the Mock Conversions functionality and reload what you previously had converted (with updates to templates as needed) before continuing the conversion.
- The Data Conversion application is supported in English only. However, the converted data can and should be loaded in the primary language for your implementation, with additional translations included where applicable.

4

Prepare for Conversion

Overall Customer Responsibilities

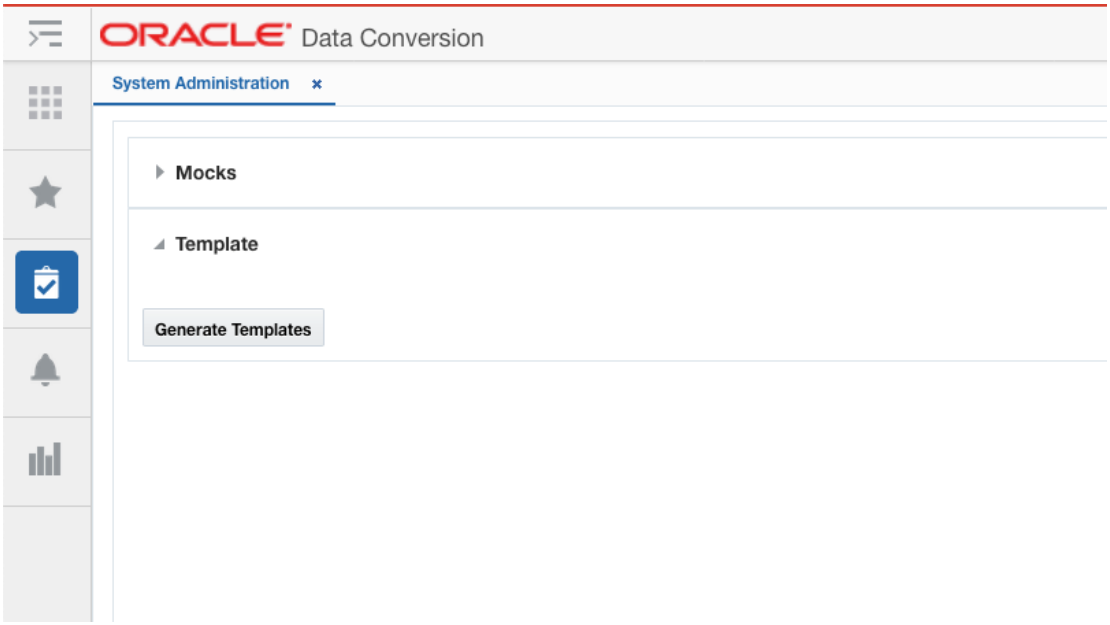
Across all functional areas of conversion, there are some basic customer responsibilities as it relates to data conversion that you and your system implementation partner need to consider. You will be responsible for:

- Defining rules to determine which data in legacy systems will be converted.
- Cleaning up data in legacy and closing all open transactions that can be closed prior to conversion to minimize the data needing to be converted. It is generally assumed that any closed transactions in legacy will not be converted, bringing only active data forward into Merchandising.
- Building extracts and transformation programs to format all data into the structure needed to utilize the conversion tool outlined later in this document. Where necessary, these programs should also include defaults for attributes not available in legacy.

The method of extraction and data cleansing in your legacy solutions can use various tools and methods in order to transform it into the format expected in Merchandising. However, the transformed data must be formatted into the format described in the provided templates in order to load properly using this tool.

Download Templates

To download the templates that give the format for the various data entities to be converted, access the System Administration screen in the Data Conversion application.



Click the **Generate Templates** button to download the templates. This will prompt you to save the generated zip file containing all the .dat files to your local machine.

There will be one file per table that can be supported in the conversion and each file will contain details on:

- File naming pattern
  - For example, the naming pattern for the DEPS table is defined as deps*.dat. The * in the name can be replaced by a number (such as deps1.dat, deps2.dat) or just left off, depending on your conversion plans.
  - If the filename does not match the pattern it will be ignored and will not be picked for loading.
- Column sequences
  - Your data files must follow the sequence of the columns specified in the template. If any column data is to be left as null, then it should be present, but just left blank.
- Data type and length of each column in the table, including a short description of the columns and table
- Whether the column is mandatory
- Primary key for the table, along with any foreign or unique keys and any check constraints

> **Note:**
>
> This file is generated based on the Merchandising data model and only the validations aligned with the Merchandising data model are listed. The business-level validations are not called out.

```
COMMENT ###############################################################################
COMMENT Oracle RMS Data Converter sample file.The lines beginning with 'COMMENT' will be ignored
COMMENT Entity : DEPS
COMMENT Table Comments : This table contains one row for each department within the company.  Whenever a department name etc., is used by Oracle Retail,
or a department number is validated, it is always COMMENT selected from this table.  The values C and R represent whether the markup % was based on a %
of Cost or Retail.C is markup based on percentage of Cost: ((Retail - Cost)/Cost) * 100  R is markup COMMENT based on percentage of Retail ((Retail -
Cost)/Retail) * 100
COMMENT ###############################################################################
COMMENT Delimiter to be used: ,
COMMENT ###############################################################################
COMMENT File name patter : deps*.dat
COMMENT ###############################################################################
COMMENT COLUMN DETAILS:
COMMENT ###############################################################################
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:1
COMMENT COLUMN_NAME:DEPT
COMMENT DATATYPE:NUMBER(4,0)
COMMENT MANDATORY:Y
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains the number which uniquely identifies the department.
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:2
COMMENT COLUMN_NAME:DEPT_NAME
COMMENT DATATYPE:VARCHAR2(120)
COMMENT MANDATORY:Y
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains the name of the department.
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:3
COMMENT COLUMN_NAME:BUYER
COMMENT DATATYPE:NUMBER(4,0)
COMMENT MANDATORY:N
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains the number which uniquely identifies the buyer for that department.
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:4
COMMENT COLUMN_NAME:MERCH
COMMENT DATATYPE:NUMBER(4,0)
COMMENT MANDATORY:N
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains the number which uniquely describes the merchandiser for that department.
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:5
COMMENT COLUMN_NAME:PROFIT_CALC_TYPE
COMMENT DATATYPE:NUMBER(1,0)
COMMENT MANDATORY:Y
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains a number which indicates whether profit will be calculated by Direct Cost or by Retail Inventory.  Valid values are: 1 =
Direct Cost, 2 = Retail Inventory
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:6
COMMENT COLUMN_NAME:PURCHASE_TYPE
COMMENT DATATYPE:NUMBER(1,0)
COMMENT MANDATORY:Y
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains a code which indicates whether items in this department are normal merchandise, consignment stock or concession items.
Valid values are: 0 = Normal Merchandise,  1 = Consignment Stock, 2 = Concession Items
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:7
COMMENT COLUMN_NAME:GROUP_NO
COMMENT DATATYPE:NUMBER(4,0)
COMMENT MANDATORY:Y
COMMENT DEFAULT VALUE:
COMMENT COLUMN_COMMENTS:Contains the number of the group in which the department exists.
COMMENT ###############################################################################
COMMENT COLUMN_NUMBER:8
COMMENT COLUMN_NAME:BUD_INT
COMMENT DATATYPE:NUMBER(12,4)
COMMENT MANDATORY:Y
```

Key Data File Assumptions

When generating your data files based on the template format, there are few key things to keep in mind.

- The data for each table will need to be loaded separately as its own .dat file - at least one per table.
- The file names should match exactly the pattern specified in the template. The file names are case sensitive and should match the case in the prescribed pattern.
- The delimiter to be used to separate the data for each column must be a comma.
- The data file should contain all the columns as expected by the template. If one or more columns are to be left null, then they should still be separated with commas signifying no value; see example below. If the value of null is loaded as 'null', then it will be considered as a string with value 'null'.
- If a comma is a part of the data, then it should be escaped using '\\'
  - For example, if an address field value is No. 21,2nd Street, it should be provided as: No.21\\,2nd Street
- There should be no empty lines in between rows in your files, as it might consider it as the end of the file.

- Newline (\n) and backslash (\) are not supported in the input file. If these characters are required in the data, substitute them with an alternate character during data conversion. After data conversion is completed, as desired, replace the alternate character with newline or backslash characters using the APEX Data Viewer.
- The values in the fields exposed on the template will not be defaulted from Merchandising sequences. Similarly, the fields corresponding to IDs on the templates will not be based on the Merchandising sequences. The associated Merchandising sequences are automatically ramped up at the end of data conversion.

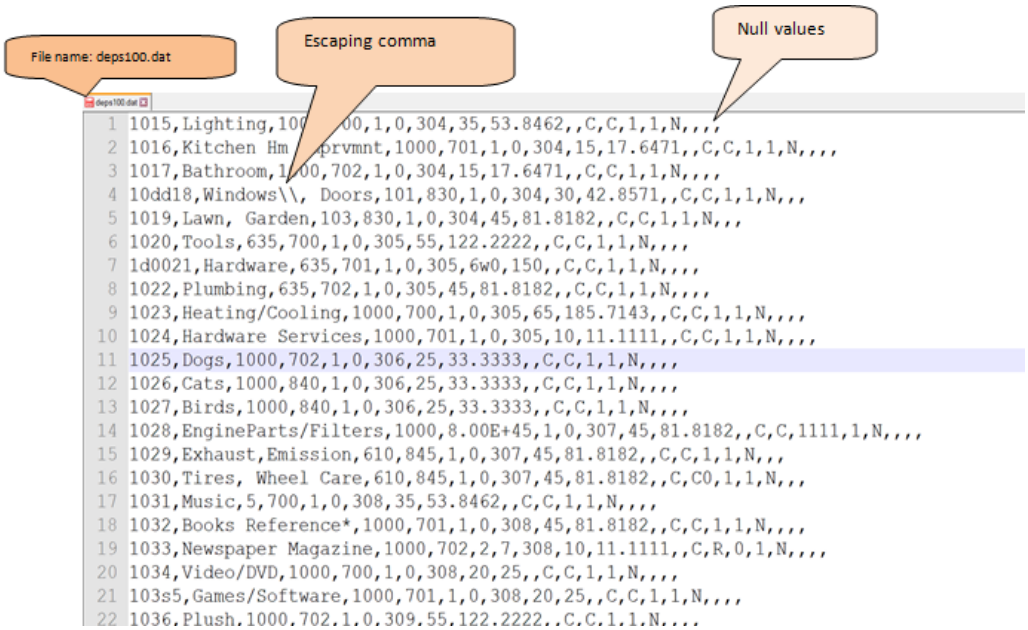
The tool mandates all of the data files to be zipped. The file name of the zip file as well as the individual files within should be as per the recommended pattern.

When zipping your files, ensure that there is only one file per table.

> **Note:**
>
> The files should be directly zipped and not zipped as a folder (For example, the unzip process of the tool, expects to extract the files directly without encountering any folders).

Department Example

Below is an example of a file used for loading department data, named deps100.dat. In the example, you can also see examples of how to format null columns and use the escaping comma.



Integration Triggers

During the data conversion run, it is highly recommended that you temporarily disable MFQUEUE publishing triggers, which capture data for integration to downstream systems. As noted in the "Converting Non-Merchandising Solutions" section, there are other methods that should be used for seeding downstream systems, if needed, after Merchandising conversion. To disable MFQUEUE triggers leverage the DISABLE_PUBLISHING_TRIGGERS task in the System Administration screen. This task will disable all MFQUEUE publishing triggers. Details on how to do this can be found in the "Task Execution Engine" section of this document.

Once the data conversion is completed, you will need to re-enable the triggers and also ensure that the converted data is properly readied for future publication, so that it is sent as an update (or delete) rather than create. Based on your Merchandising data integration to downstream systems, enable the required triggers by executing ENABLE_TRIGGER task in the System Administration screen. See the "MFQUEUE Triggers" section in the appendix for the list of triggers.

Finally, to prepare your data for future publication, records must be loaded into PUB_INFO tables for these entities with the published indicator set to Y. In the cases where the converted entities do not have a separate publishing table and published indicator resides in the merchandising table itself, the indicator should be marked as Y (published). To initialize Merchandising publishing tables, execute the INIT_PUBLISHING task in the System Administration screen. It will populate PUB_INFO tables as necessary and sets the published indicator to Y for the converted entities.

Golden Gate Replication

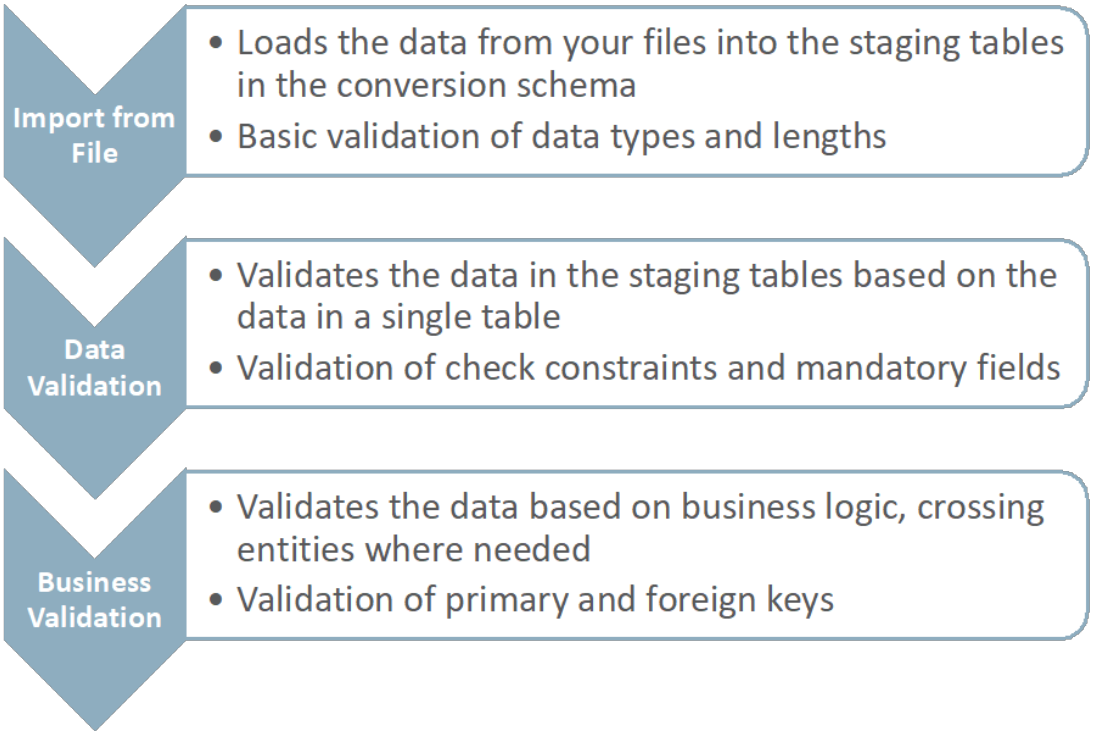
It is strongly recommended you choose an environment to use for conversion that does not need replication by Golden Gate, as disabling Golden Gate before data conversion and re-starting the services after import is completed is an extensive process involving efforts from both, you and Oracle Cloud Operations team. However, in case your data conversion environment does have Golden Gate replication enabled, it should be **disabled for data conversion processes**, including all mock runs, to progress at maximum throughput. For this, you are required to log an SR with Oracle Cloud Operations team. After completion of data conversion, Golden Gate replication can be turned back on.

5

Data Conversion Application

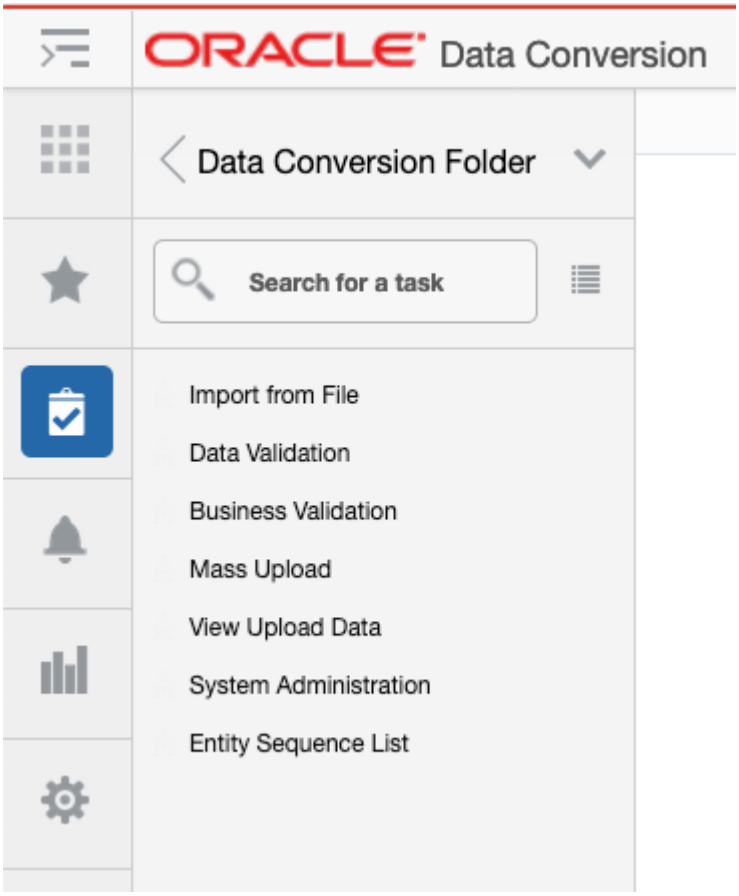
Once your files have been created with the expected file name patterns and format, they can be moved to the folder in the app server for uploading. You will have received instructions from the Oracle Cloud Operations team for moving files using the Merchandising File Transfer Service that should be followed here.

Once this is complete, you will be ready to use the Data Conversion Application to load data from these files into the conversion schema. Data can be loaded one table at a time or in bulk. For your first runs, you may want to run tables individually to determine if there are any major formatting issues that need correcting in your transformation programs.



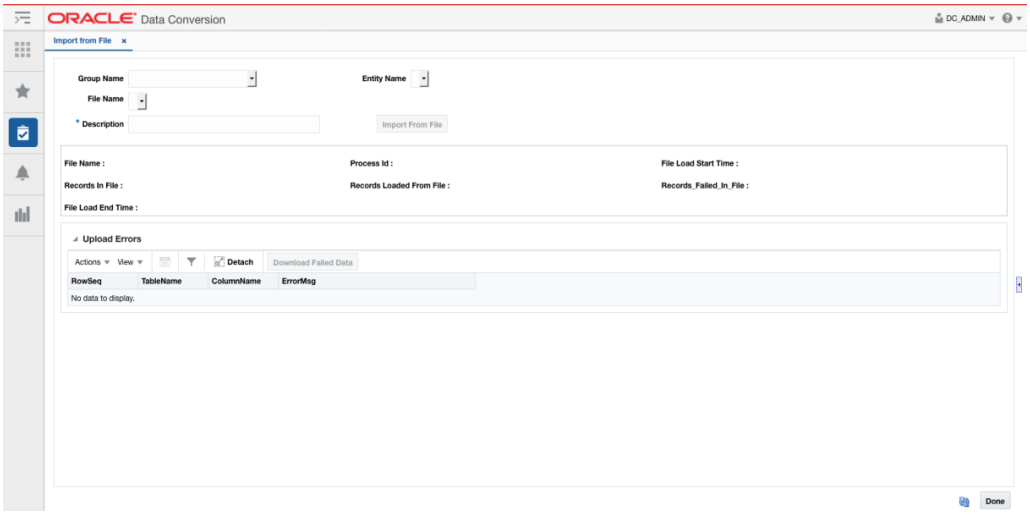
Logging In

Log in to the Data Conversion application with your configured username and password. Once logged in, your task list should look like the picture below, depending on the privileges granted to you.

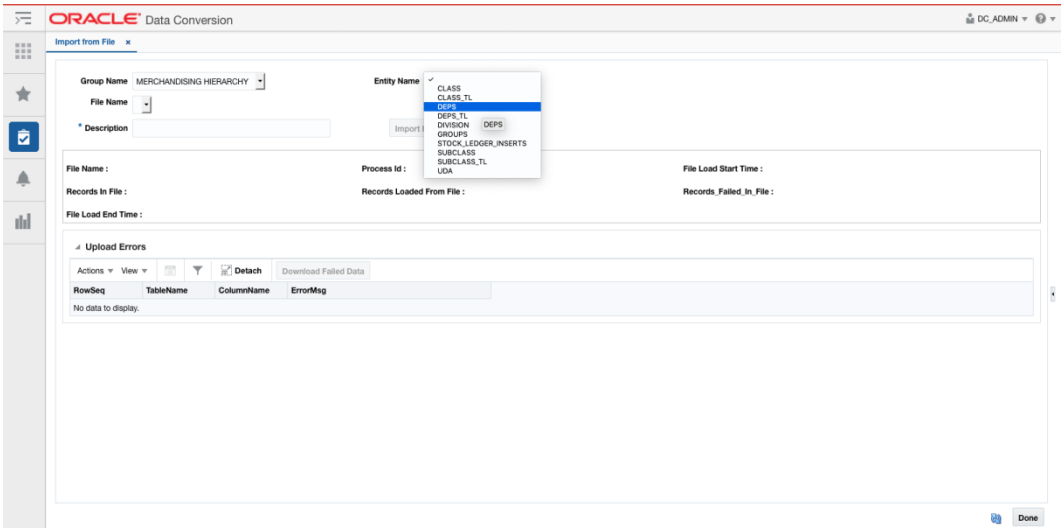


Import from File

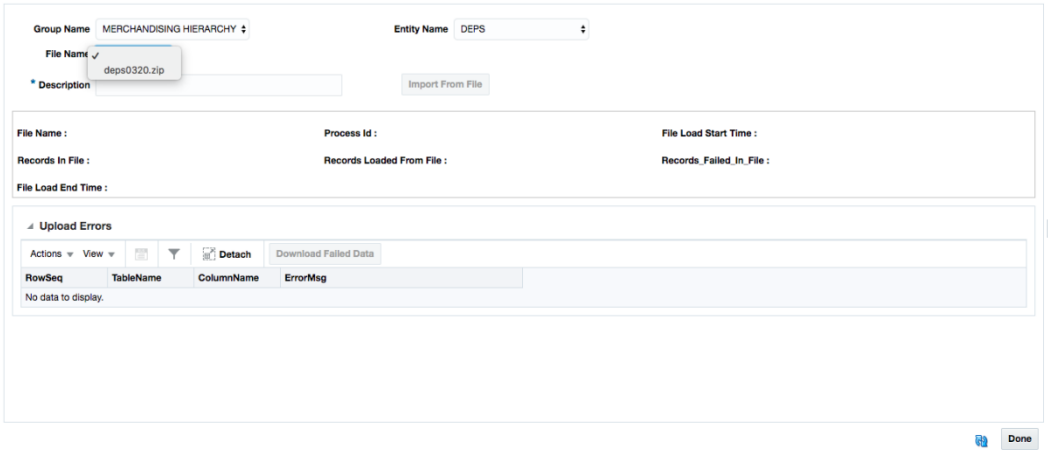
Start with the Import from File option. This screen will aid you in loading data from the files you uploaded to the staging tables present in the data conversion schema. See the "Appendix: Entity Sequence List" section in the appendix for details on the entity groups, entities, naming conventions, and file types supported for this step in the process.



Select the entity group whose data you want to load. Then select the entity name for the table whose files you want to load. In the example below, DEPS is selected to load department data.



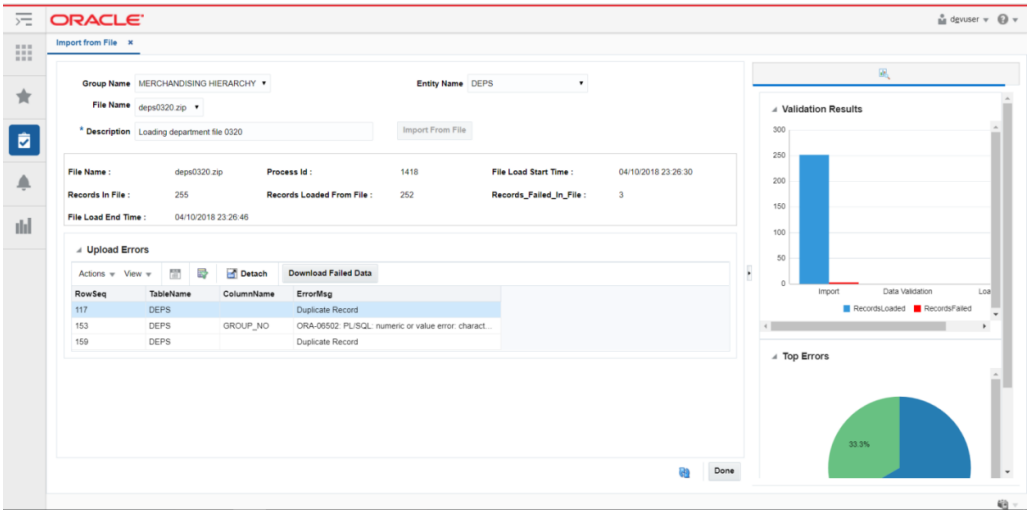
Next, select a file name to load. The dropdown will display the list that is available for loading based on the group and entity combination selected. In this example, a .zip file has been selected that contains two .dat files - one for DEPS and one for VAT_DEPS.



Then, provide a description to identify the process. This will enable the **Import From File** button. On clicking the button, the data is imported from the file to staging tables. During import, the zip file is extracted for the individual dat files, when applicable.

At this stage, all files are considered individually, so no errors will be raised for missing files for tables that have a dependency on the entity selected, like in this example where DEPS and VAT_DEPS have dependencies.

This allows you to load the dependent tables separately. If not loaded prior to reaching the business valdation step, the missing dependent records would be caught during that stage.



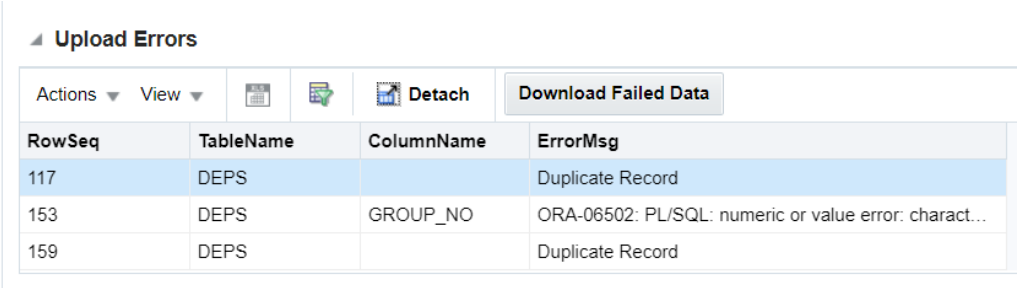
Each file upload is identified using a Process ID. This becomes the identifier while progressing with the validation and load actions on the successive screens. After the import completes, a summary of the import results can be seen in the middle of the screen.

- File Name: this is the original file you imported.
- Process ID: unique identifier for the upload generated by the application.
- Records in File: total records detected in the original file.
- Records Loaded from File: the total records that were successfully loaded into the staging tables from the selected file.
- Records Failed in File: the total records rejected from the file.
- Start and end timings: date and time stamps for these data points to help you gauge the time it takes to load your data and better plan your conversion timeframes.

For all the above statistics, the sum of all the records in all the dat files contained in the zip file is displayed in this section.

View Upload Errors

Records with errors are displayed in the table at the bottom of the screen. This table shows which row in the original file had the error, which table it was trying to load, the column in error, and details on what error occurred.



For the above error examples, the file below is an example of a file that may have generated the second two errors.

```
3408,Test KI Dept,1000,702,1,0,220,10,11.1111,,R,C,1,1,Y,
3409,Building Materials,1000,700,1,0,304,20,25,,C,C,1,1,Y,
3410, Paint*,1000,701,1,0,304,35,53.8462,,C,C,1,1,Y,
3411,Lighting,1000,700,1,0,304,35,53.8462,,C,C,1,1,Y,
3412,Kitchen Hm Imprvmnt,1000,701,1,0,304,15,17.6471,,C,C,1,1,Y,
3413,Bathroom,1000,702,1,0,304,15,17.6471,,C,C,1,1,Y,
3414,Tools,635,700,1,0,305,55,122.2222,,C,C,1,1,Y,
3415,Plumbing,635,702,1,0,305,45,81.8182,,C,C,1,1,Y,
3416,Heating/Cooling,1000,700,1,0,305,65,185.7143,,C,C,1,1,B,
3417,Hardware Services,1000,701,1,0,305,10,11.1111,,C,C,1,1,Y,
3418,Dogs,1000,702,1,0,306,25,33.3333,,O,O,1,1,K,
3419,Cats,1000,840,1,0,306,25,33.3333,,C,C,1,1,Y,
3420,Birds,1000,840,1,0,306,25,33.3333,,C,C,1,1,Y,
3421,Books Reference*,1000,701,1,0,308,45,81.8182,,C,C,1,1,Y,
3422,Video/DVD,1000,700,1,0,308,20,25,,C,C,1,1,Y,
3423,Plush,1000,702,1,0,wqeqwe,55,122.2222,,C,C,1,1,Y,
3424,Dolls,1000,855,1,0,309,55,122.2222,,C,C,1,1,Y,
3425,Frames,1000,860,1,0,310,70,233.3333,,C,C,1,1,Y,
3426,Lenses,1000,860,1,0,310,60,150,,C,C,1,1,Y,
3427,Personal Care*,1000,701,1,0,311,30,42.8571,,C,C,10,10,Y,
3428,Health Products,1000,702,1,0,311,35,53.8462,,C,C,1,1,Y,
3410, Paint*,1000,701,1,0,304,35,53.8462,,C,C,1,1,Y,
```

At this stage, records can be rejected due to the following reasons:

- Data type mismatches in any of the columns
- Size mismatches in any of the columns
- Duplicate records within the same file based on the primary key of the table

If there a large number of errors, you can export the errors to a spreadsheet by selecting the **Export to Excel** button on the toolbar.

All files that were successfully process will be moved to a process folder and can proceed to the next step. You can also choose to proceed to the next step for the successfully loaded data before correcting the import errors. If you choose to proceed, then the records that had failed can be loaded separately in a new file after making corrections.

Correct Import Errors

To help with correcting errors, you can use the Download Failed Data option in the screen. This will download the file (maximum size 2MB) to your local machine with only the erroneous records. You can then manually correct them and reload them to trigger a fresh file upload against that data.

After analyzing the errors displayed, you can make corrections in the original file and re-load all data or you may need to correct the legacy data or transformation code and regenerate your files. Regardless of the method, you will need to transfer the corrected file through Merchandising File Transfer Service and repeat the load process. A new process ID is generated against this second load.

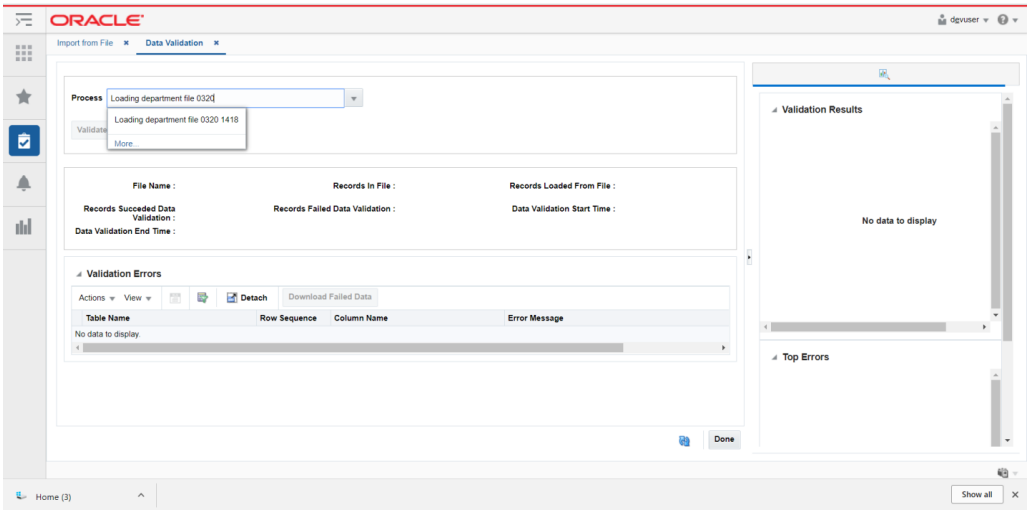
To load a different file for the same or a different entity, click the **refresh** icon ( ) at the bottom right of the screen, to reset the search section.

To close the screen, click the **Done** button at the bottom of the screen.

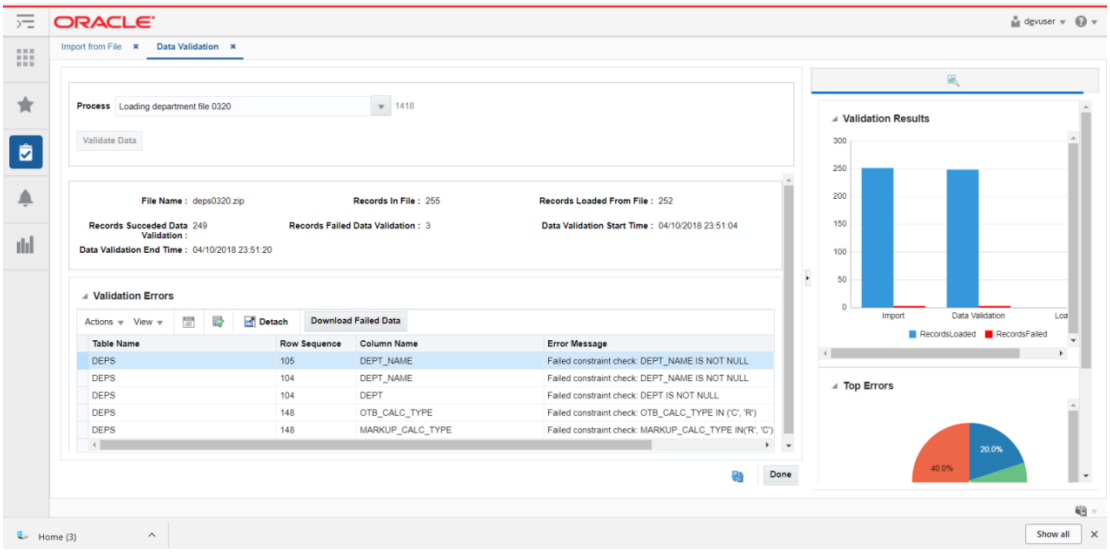
Data Validation

The next step in the process will take the data that was successfully loaded data from the imported files and validate it for data correctness. The validations covered in this step are:

- Mandatory fields/not null checks
- Check constraints on columns
- Complex check constraints on the table rows
- To access this function, click on the Data Validation link from the task list, to open the screen. Then, from the Process drop down, select the process against which the file was loaded. The drop-down allows you to search based on the process description that was provided during file import.

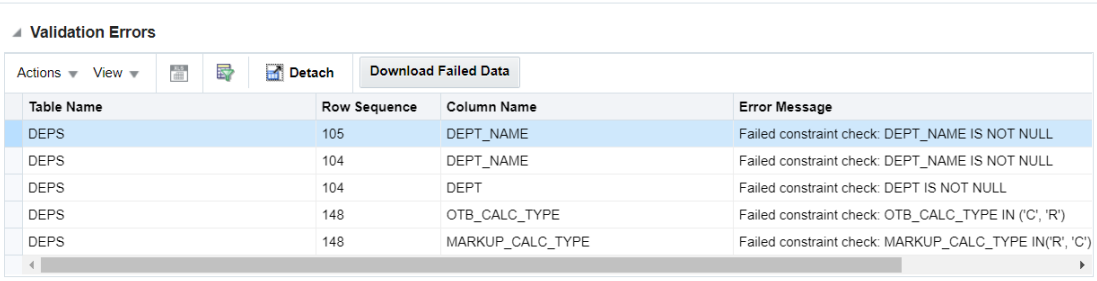


Click the **Validate** button to begin data validation on your set of records. Once complete, you will see the statistics for this processing similar to the Import from File screen.



View Data Validation Errors

The validation errors will be displayed in the table at the bottom of the screen after this process runs. As with the Import from File process, you can see the row number that had the issue, the table with the issue, and the error message indicating the issue. For the Data Validation screen, the column with the issue is also displayed.



In the example above, you can see several example errors that might occur during this stage. In this case, three records (row numbers: 105, 104, and 148) have failed data validations. In some cases, there were multiple issues for a particular row of data, but as you can see failures for every column are captured.

After analyzing the errors displayed, you can either make corrections on the original file or continue processing the records that were successfully validated. If you choose to continue to the business validations, then you can use the same process ID to continue processing just the successfully validated records.

Correct Data Validation Errors

To correct the errors, you can make corrections in the original file and re-load all data or you may need to correct the legacy data or transformation code and regenerate your files. To help with this process, use the **Download Failed Data** button to download the erroneous records into a file, bounded by size configurations on the deployed application. Then, you will need to

re-import the file and then re-run the data validation process to validate that all errors have been corrected. A new process ID is generated against this second load.

To validate another process, click the refresh icon ( ) at the bottom right of the screen. This will reset the search section.

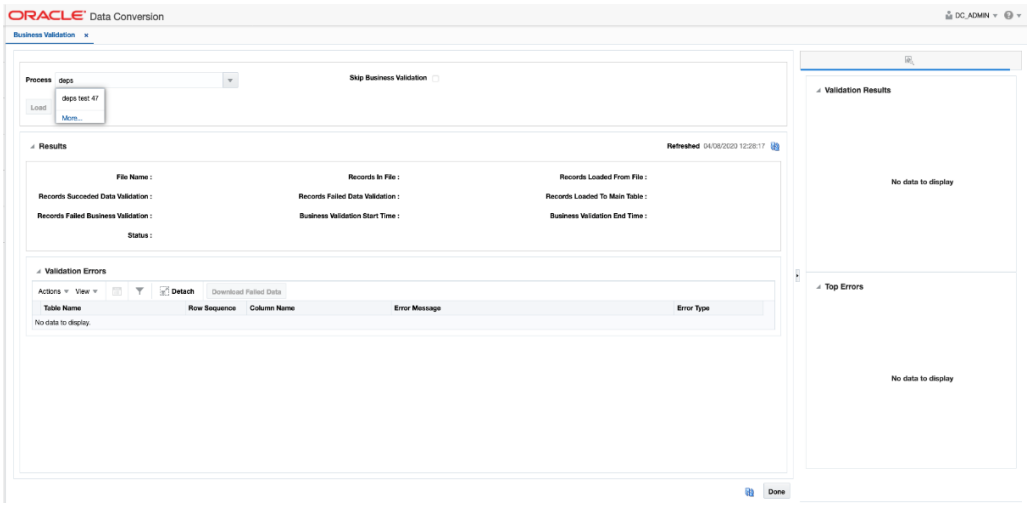
Close the screen by clicking on the **Done** button at the bottom of the screen.

Business Validation

Records that have been successfully validated will next be validated for business rules and, if successful, will be loaded to the main Merchandising tables. The validations covered in this step are:

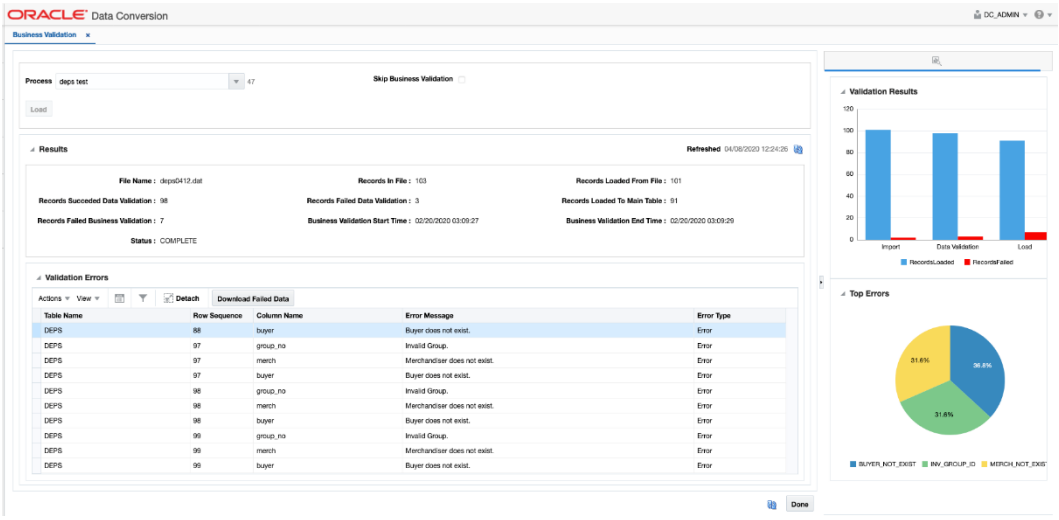
- Primary keys
- Foreign keys
- Unique keys
- Merchandising and Pricing business validations

To access this screen, click Business Validation link in the task list. From the Process drop down, select the process against which the file was loaded. The drop-down allows you to search based on the process description that was provided during file import.



At this point, you will also be given an option to skip business validations and directly load all records that have passed the first two steps of the process (Import from File and Data Validation). If you do this, then records will be loaded from staging to the main Merchandising tables without doing any validation against database constraints. This can speed up the process of loading data, but can result in loading incorrect data that may be harder to correct later. It is not recommended to use this process for initial conversion runs. Additionally, this option will get overridden for the entities that do not insert the data directly as it is in the input files. For example, if there are calculations for some columns that need to be made prior to loading the data into the tables.

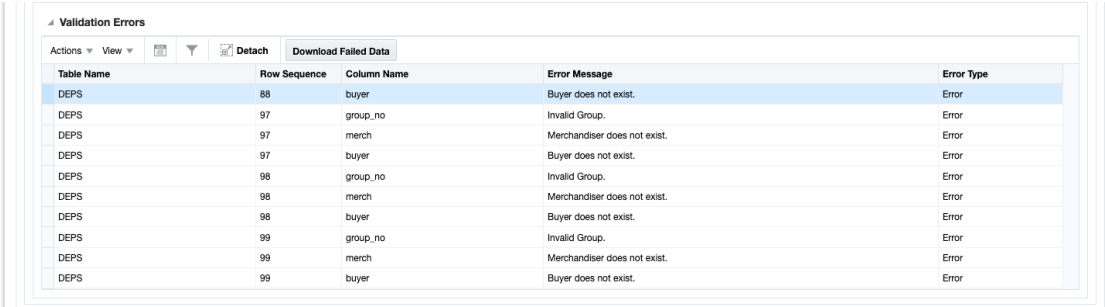
Click the **Load** button to run the business validations and load data to the main tables.



Once the process completes, the loading statistics are displayed in the screen, giving details of total records processed and how many were successful or had errors.

View Business Validation Errors

The validation errors will be displayed in the table at the bottom of the screen after this process runs. As with the previous steps, you can see the row number that had the issue, the table and column with the issue, and the error message indicating the issue.



Correct Business Validation Errors

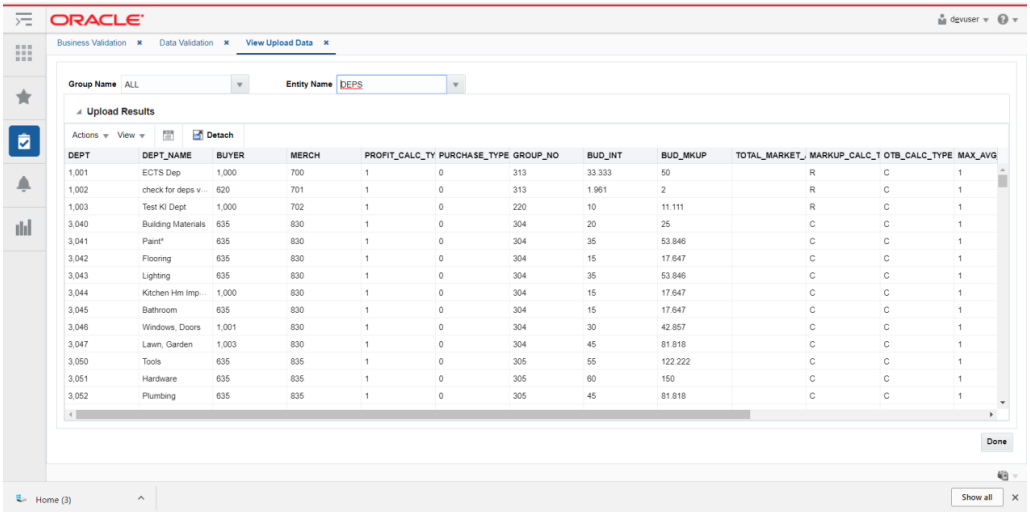
To correct the errors, you can make corrections in the original file and re-load all data or you may need to correct the legacy data or transformation code and regenerate your files. To help with this process, use the **Download Failed Data** button to download the erroneous records into a file, bounded by size configurations on the deployed application. Then, you will need to re-import the file and then re-run the data and business validation process to validate that all errors have been corrected. A new process ID is generated against this second load.

To validate another process, click on the refresh icon ( ) at the bottom right of the screen to reset the search section. Or close the screen by clicking the **Done** button at the bottom of the screen.

View Uploaded Data

After performing the three steps outlined above, it is highly recommended that you verify the records have made it into the base tables as expected. To assist with this the View Upload Data screen can be used.

To access this screen, select the View Upload Data option from the task list and select the group name and entity you wish to view.



No changes can be made to the data in this screen; however, you can use the Export to Excel option to export the data if you see errors that need correcting.

You can also view the data in the Merchandising screens or directly in the database using the APEX Data Viewer (via the link in Merchandising).

Mass Upload

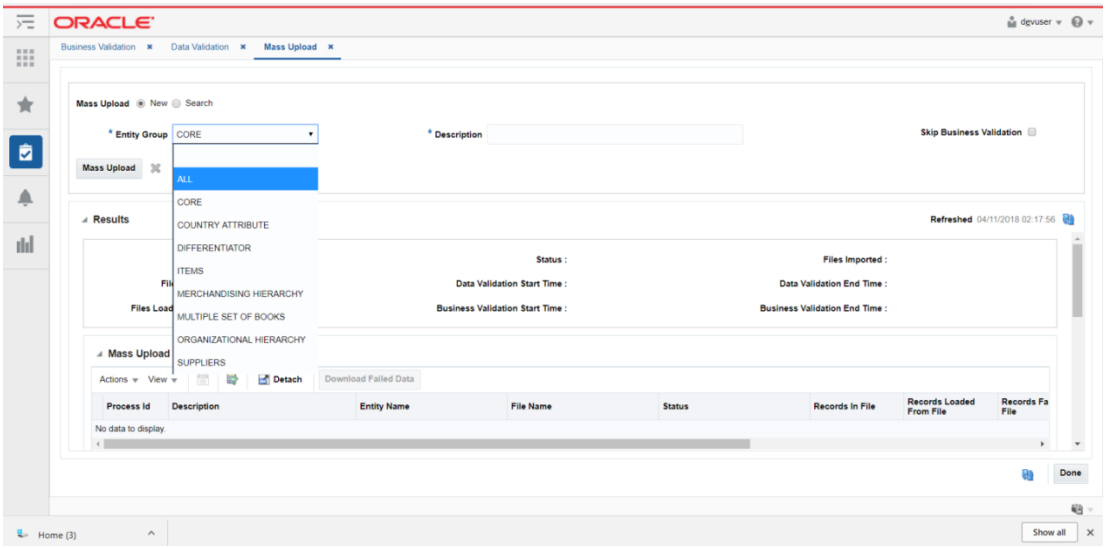
This feature is provided in the application to trigger an upload of multiple entities sequentially through the same three stages described above. It is not recommended to use this during the initial conversion cycles, but rather is intended to be used during the final runs, where there is a higher level of confidence that the data files are clean.

You can use this to trigger an upload for each of the entity groups, one after the other or everything together. If you choose the first option, then you should ensure that the groups are triggered sequentially, one after the other, as per the sequence defined in appendix. Not following this could lead to errors stating mismatch between dependent records.

Only one mass upload process can run at a time. Trying to create a new mass upload process will throw an error if the first hasn't completed. Additionally, when a mass upload is being run, the remaining screens for individual upload (import, validate, load) should not be used. As a Prerequisite, the required files should be available on the configured upload folder, similar to when using the Import from File screen. The mass upload process will process the files that match the prescribed pattern.

To access the screen to initiate the mass upload, click on the Mass Upload option in the task list. Next, select the **New** radio button and then select the entity group for the mass upload. If you want to trigger it for all entities, then select the All option[1].

Then, you will next need to provide a description for the process.

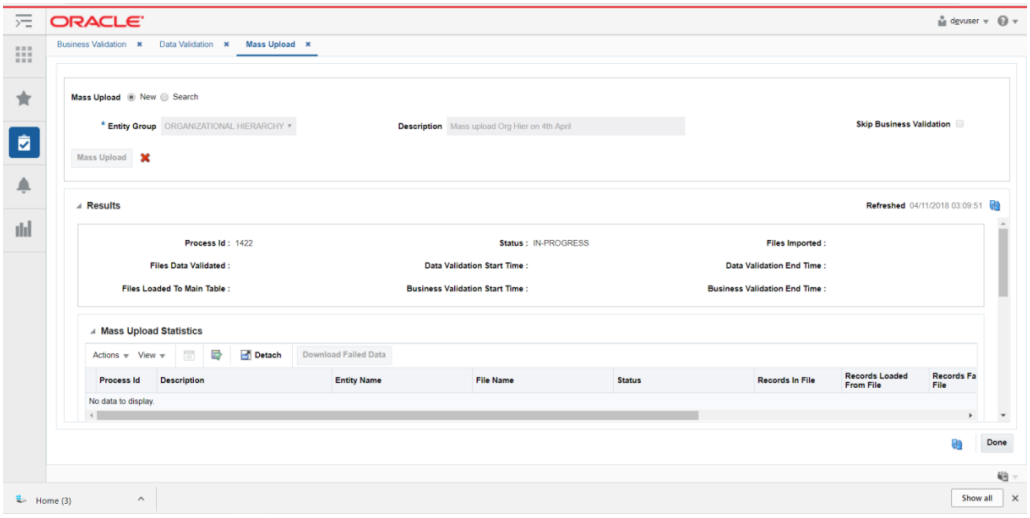


Similar to the Business Validation process, you are given the option to skip business validations at this point by checking the Skip Business Validation checkbox. This will directly load all successfully validated records from the import and data validation stages into the Merchandising tables. If you choose to do this, then records will be loaded from staging to main Merchandising tables without doing any validation against database constraints like primary key, foreign key, or unique key violations. This can speed up the process of loading data, but can result in loading incorrect data that may be harder to correct later. Additionally, this option will get overridden for the entities that do not insert the data directly as it is in the input files. For example, if there are calculations for some columns that need to be made prior to loading the data into the tables.

Click the **Mass Upload** button to trigger the mass upload, which is an asynchronous process.

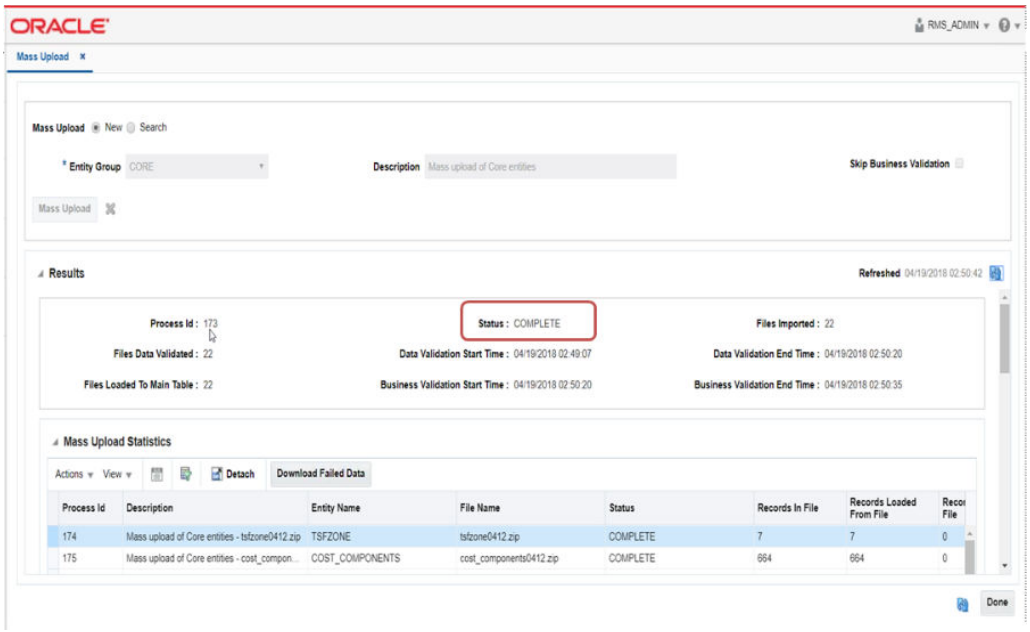
You will need to click the **refresh** icon button ( ) to see the status of the process.

1 Best practice for using the Mass Upload is to use the entity group level for your initial runs to make it easier to find and correct format and data errors. Then, once there is a level of confidence in the converted data, the All option can be selected.

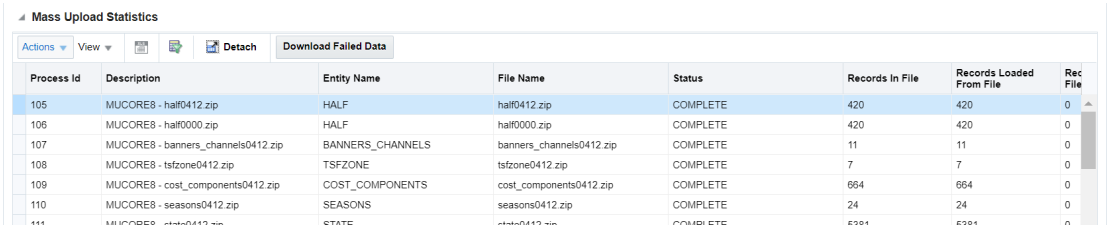


Once you have kicked off a process, a stop icon button ( ) will be enabled. This can be used to halt the execution for in cases where you wish to make some alterations and restart. Restarting will be a fresh mass upload process. This option will be available only in the initial stage where the records are not yet moved to main table. Once the mass upload process has started moving the records to the main table, then the option will not be available in order to prevent any corruption of data.

Once the upload is completed the status on the screen for this process will be marked as COMPLETE. The Results panel provides the statistics for the overall mass upload process.



Mass Upload Results

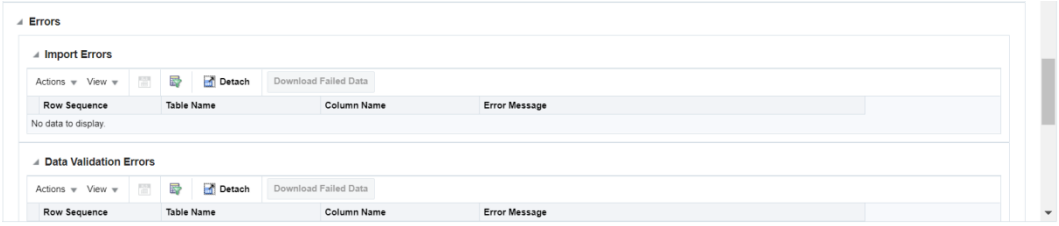


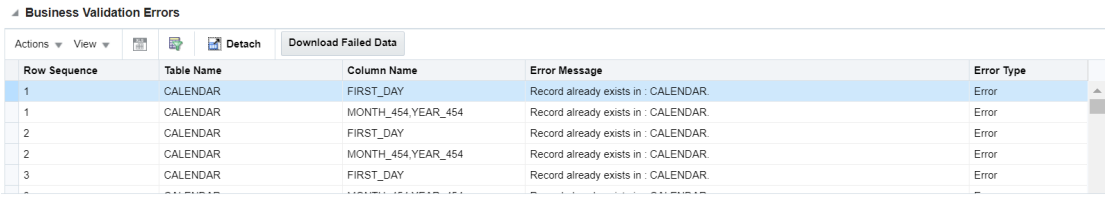
At the bottom of the page, a table provides the details of the individual child processes. Each file will be loaded in a separate child process and is listed in the table with details like file name, record count, processing status, and so on.

Like in the other screens, a **Download Failed Data** button is provided to download the records that have failed for the selected child process into a file with details on all three validation steps.

View Errors

To view the errors for each child process, select the row in the Mass Upload Statistics table. This will display the data in three separate tables below - one for each stage of the process - in the Errors section. Each of these tables also have a **Download Failed Data** button so that the errors for each stage can be exported and viewed separately.





Download Errors

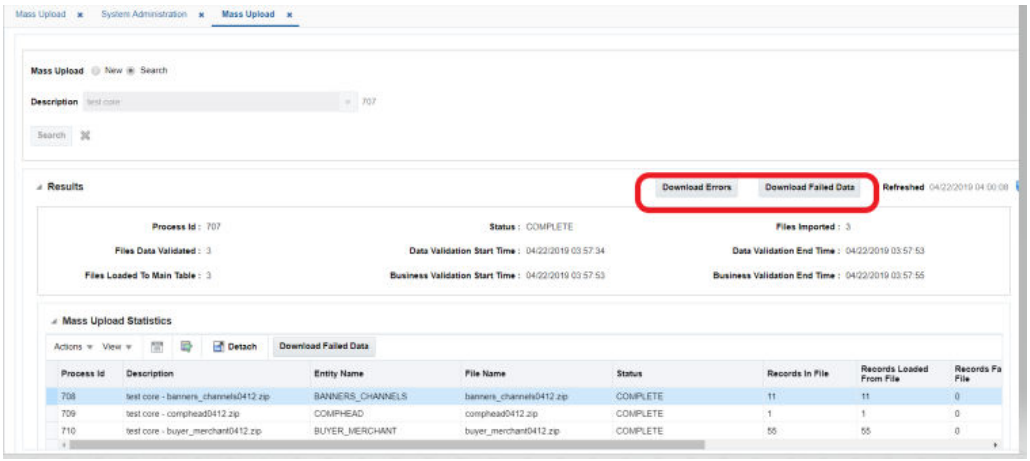
For completed processes, the consolidated errors for the entire upload process can be downloaded using one of two options:

Download Errors

For this option, a .csv file with the errors for all the files processed in the selected mass upload will be generated. If the file is less than 2MB in size it will be generated to your machine. If not, it will be moved to object storage from where you can download using the Merchandising File Transfer Service.

Download Failed Data

For this option, a .zip file with the data from failed rows will be generated. This could then be corrected and reprocessed separately. If the file is less than 2MB in size it will be generated to your machine and can be downloaded from the UI. If not, it will be moved to object storage from where you can download using the Merchandising File Transfer Service. While downloading the files, use `dataconversion/outgoing` as the object storage (OS) prefix. Details on Merchandising File Transfer service can be found in the *Oracle Retail Merchandising System Operations Guide, Volume 2*.



> **Note:**
>
> The file containing failed records will not include any escape characters. Before uploading the file for reprocess, it should be verified and, as needed, escape characters should be added.

Search for a Mass Upload Process

You can also search for a previously executed mass upload by accessing the screen and selecting the **Search** radio button. Next, select the process you wish to search for and click Search. The status is provided in the Results panel.

6
Load to Production Environment

After completing your conversion of data and validating all the resultant data, the last step is to promote the data into your production environment, also known as "lift and shift". This process will need to be performed in coordination with the Oracle Cloud Operations team. Log an SR to schedule with that team.

The extract of data from the non-production environment to the production environment is a full export of the schema and a full overwrite of the production schema. Therefore, you should ensure all data you do not want promoted from the schema has been removed, such as any data in the integration queue tables or any tran data records generated as part of PO or customer order conversion.

Integration queue tables, used for RIB publication, may have data if triggers were not disabled prior to running the conversion processes and if so will need to be cleaned up. As well, details on the publication tables also needs to be set up correctly for future integration. It is assumed that this would normally occur on the final day of cutover after completing multiple trial, or mock, conversions. In addition to clearing the integration queue tables, whether you disabled the triggers or not, the published indicator must be marked as Y (published) in the integration publishing tables to ensure future integration flows as expected.

To do both of these tasks this, once data conversion is complete and before the lift and shift of the converted data into your production environment, execute the INIT_PUBLISHING task in the System Administration screen. This task will clean up MFQUEUE tables if triggers were not disabled and initialize Merchandising publishing tables for the converted data. Details on how to do this can be found in the "Task Execution Engine" section of this document. See the "Publication Tables" for a list of tables to validate that records exist and that the flags are properly set.

For the actual lift and shift there is a standard set of tables that are excluded, such as those related to data filtering and security (roles, privileges, and duties), materialized views, temporary tables, spreadsheet template configuration, and Sales Audit rules and totals. If necessary, you can coordinate with the Oracle Cloud Operations team to add other tables to this list as well, such as if you are performing a phased conversion by functional area.

Prior to executing the lift and shift, the production schema will be backed up and RIB, Golden Gate, and Merchandising solutions will be brought down. After the import is completed, the Oracle Cloud Operations team will execute a series of standard clean-up activities in your production environment. These include

- Creating an export file for re-syncing your target DAS environment via Golden Gate
- Clearing any integration queues and re-enabling triggers and constraints
- Resetting the sequences in production
- Validating the correct business date is set
- Bringing back up the RIB, Merchandising solutions, and Golden Gate

You will then be able to start your validation of the production environment.

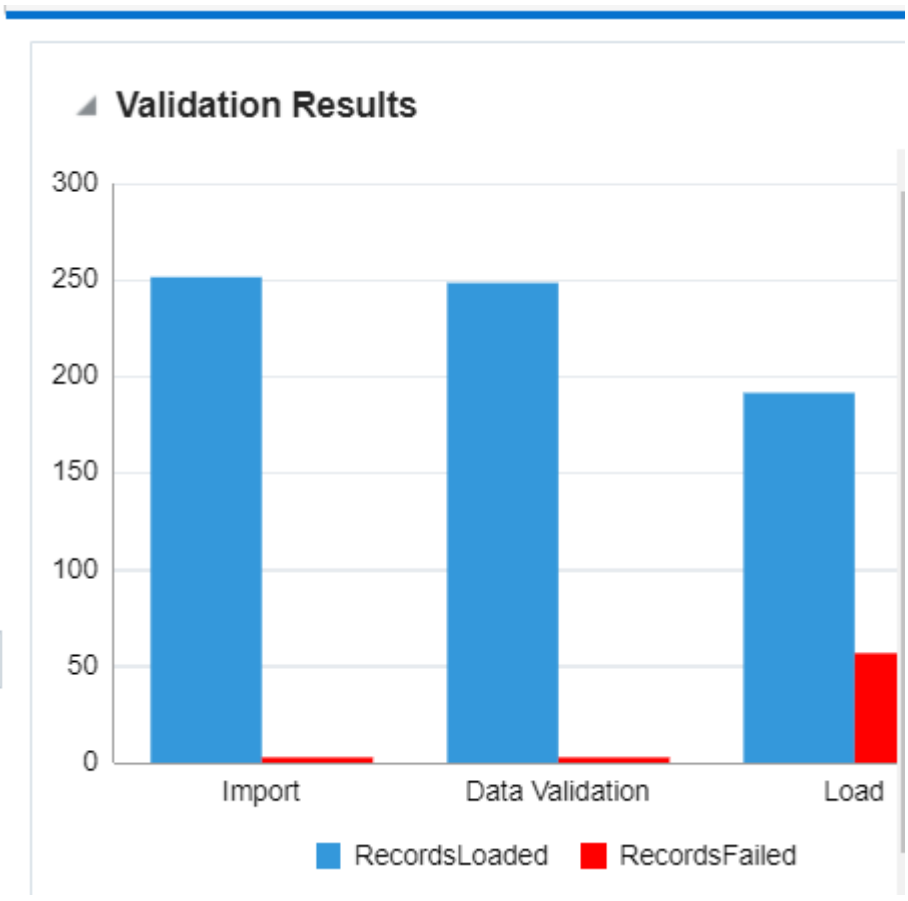
Monitoring and Troubleshooting

As you are loading and validating data through this tool, the progress can be monitored using the reports visible in the user interface. Additionally, the outcome of the validation processes is available in the UI, including the status of the process, the start and end times, and the records processed successfully and in error. For further troubleshooting read-only access has also been provided to the Data Conversion schema through the Merchandising APEX Data Viewer link.

Contextual Reports

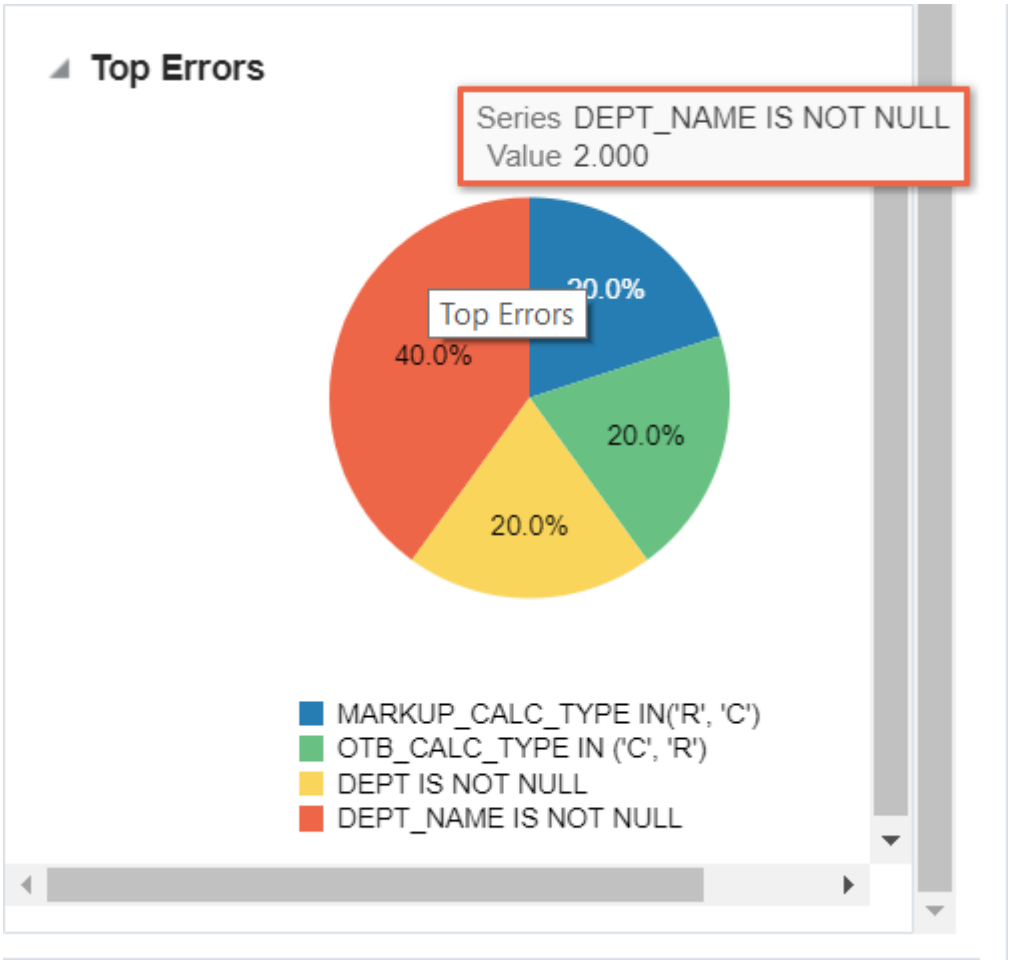
Additionally, to better analyze the errors, two contextual reports are available in each of the screens described above.

Figure 6-1 Validation Results



This summarizes validation results for the current process ID across all three steps in the loading process.

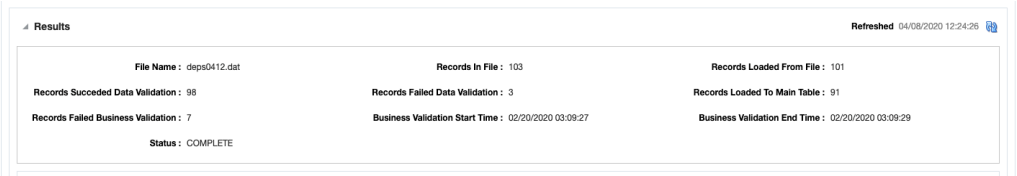
Figure 6-2 Top Errors



This shows the top errors encountered for the process ID being viewed. For example, if you are in the Import from File screen, it will show you the top errors from the file import; whereas in the Data Validation screen, it will show you the top errors from the data validation against the process ID. This may help highlight commonly occurring errors that require corrections on the legacy side or in your transformation programs.

User Interface Monitoring

At each stage along the import and validation process, the Data Conversion tool screens provide statistics for the processing of the upload, including start and end times, number of errors, and number of records successfully processed. Also, for Business Validation and Mass Upload, if you have configured a certain data entity to run using threads or chunks, you will be able to monitor the completion by chunk during business validation by clicking the **Refresh** icon button ( ) in the results section.



Database Monitoring

For larger processes that may take longer to complete due to the volume of data being processes, read-only access has been provided to the data conversion schema to support:

- Monitoring completion by chunk
- Debugging validation errors

To monitor database sessions query V_DC_SESSION_INFO view through the APEX Data Viewer by prefixing the data conversion schema.

The key tables that are available in the Data Conversion schema for troubleshooting and debugging are described in the table below[1].

Table Name	Description
DC_MASTER_PROCESS_TRACKER	Used in data conversion and contains the processes associated to a mass upload process. It shows the relationship between each of the individual processes spawned in the mass upload.
DC_PROCESS_TRACKER	Contains information on the conversion process against a process ID. The information in this table is similar to what you see in the various screens, including start and end times for each phase of the process, and the count of records processed successfully and in error by phase.
DC_IMPORT_STATUS	Used during the Import from File process to calculate the file size and monitor progress.
DC_FILE_ERRORS	Used to hold the errors encountered during Import from File. In addition to the error messages, it also indicates the Merchandising table and column names for which the error applies. This is the same information displayed in the table in the Import from File screen.
DC_FILE_ERROR_DATA	Used to hold the rejected data during Import from File. Note: the data may be split across multiple columns for a single row in this table due to size.

1 To query the tables in the Merchandising schema, you'll need to use the RDC01 qualifier on the table, as synonyms were not created for these objects (for example, RDC01_DC.DC_MOCKS).

Table Name	Description
DC_VALIDATION_ERROR	Holds the errors encountered in the Data Validation stage. In addition to the error messages, it also indicates the Merchandising table and column names for which the error applies. This is the same information displayed in the table in the Data Validation screen.
SVC_ADMIN_UPLD_ER	Contains errors encountered while uploading data to Merchandising tables from Data Conversion tool during the Business Validation stage. In addition to the error messages, it also indicates the Merchandising table and column names for which the error applies. This is the same information displayed in the errors table in the Business Validation screen.
DC_MOCKS	Used to hold data on a mock cleanup cycle, including the start and end date, and status (STARTING, STARTED and COMPLETED).
DC_MOCK_EXCEPTION	If any errors are encountered when performing the clean-up cycle, they will be shown in this table.

Other tables in the data model should be considered more metadata for the solution and, although visible in the database, are likely not useful for troubleshooting. Some details on these tables below.

Table Name	Description
DC_ENTITY	This table is used in data conversion for holding information on the entities supported in the Data Conversion tool
DC_ENTITY_DETAIL	This table provides the details of the staging tables into which data from the input files would be loaded
DC_ENTITY_GROUP	This table is used in data conversion for holding the entity groupings
DC_ENTITY_GROUP_LIST	This table is used in data conversion for holding the entity group to entity mappings
DC_MERCH_SEED_TABLES	This table holds information on the tables seeded during installation.
DC_SEQ_MAPPING	Contains the sequences used by the base Merchandising tables. These sequences would be automatically ramped up to keep in line with the data loaded in the Data Conversion tool.
DC_SEQUENCE_BKP	Used to back up the sequence values at the end of each mock cycle.
DC_SYSTEM_OPTIONS	Holds the system level parameters for the Data Conversion tool. Configuration changes to this by end users is not supported.

Table Name	Description
RTK_ERRORS	Contains one row for each error message used by the Data Conversion tool in your primary language. This table is populated during installation of the tool and cannot be modified.
RTK_ERRORS_TL	Contains translations of the error messages used by the Data Conversion tool into other languages applicable for your users. This table is populated during installation of the tool and cannot be modified.
CORESVC_ITEM_ERR	This is a temporary table used when loading item data during business validation. The errors are then moved to the SVC_ADMIN_UPLD_ER table.
CORESVC_COSTCHG_CHUNKS	This is a temporary table that contains information about chunks that are used to process cost change data.
CORESVC_COSTCHG_ERR	This is a temporary table used when loading cost change data during business validation. The errors are then moved to the SVC_ADMIN_UPLD_ER table.
CORESVC_PO_CHUNKS	This is a temporary table that holds the chunking information for the PO tables.
CORESVC_PO_ERR	This is a temporary table used when loading purchase order data during business validation. The errors are then moved to the SVC_ADMIN_UPLD_ER table.

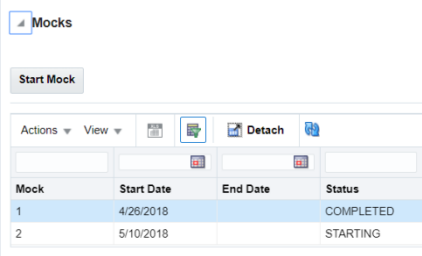
For more information on the columns for each of these tables as well as the staging tables used during the import and validation processes, please see the data model which can be viewed on My Oracle Support under ID 2619103.1.

7

Mock Conversions

During the data conversion process, it is assumed that you will go through multiple iterations of conversion while you fine tune the extracts and transformations from your legacy solutions. To support this, a feature exists in the Data Conversion application for you to perform the mock cycles and then revert the data back to the initial environment, with just your configured Merchandising installation seeded data. This is done by baselining the data in the database prior to starting any of the conversions.

To access this function, click on the System Administration link in the task list. In the System Administration screen, click Start Mock and click Yes on the confirmation message in order to kickoff the refresh process. The refresh will run as an asynchronous process.



The status of the current refresh, as well as previously requested refresh processes are shown in the Mocks table. To see the latest status, click the refresh icon button ( ).

Once the status is STARTED, it means that the new refresh has been created and users can continue their conversion activities. Please wait until it moves to STARTED status. If the process appears to be stuck in STARTING status, contact the Oracle Cloud Operations team for assistance. It is important to refrain from triggering the process again until the previous request is complete.

The mock cycle runs through the following statuses:

- STARTING - Background database processes are running to cleanup conversion data for a new mock cycle.
- STARTED - The instance has been cleaned of conversion data and is ready for the new mock cycle.
- COMPLETED - The current cycle is closed.

Important Notes

The screen should be used with caution by users with higher privileges. Clicking the **Start Mock** button will cleanup up the data and it cannot be reverted.

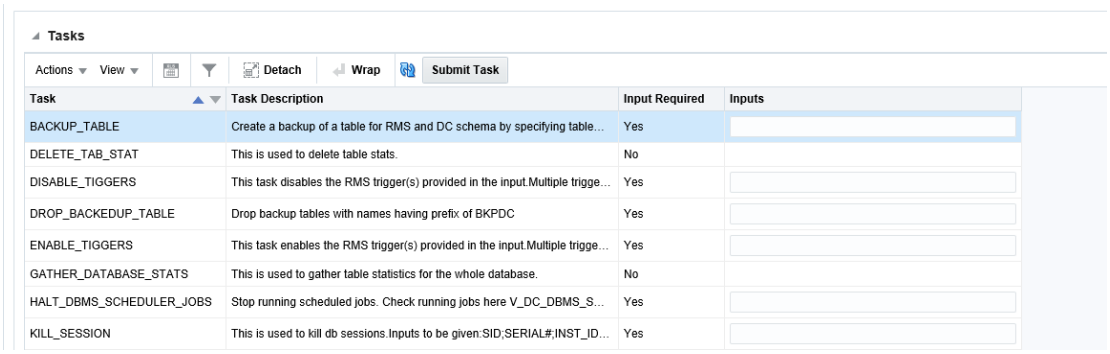
It is recommended that users are logged out from all Merchandising applications, as well as the Data conversion applications before triggering this.

When there is a need to patch Merchandising during conversion, then the conversion data will **always** be cleaned up and the environment will be brought back to the initial seeding. Hence, you will need to accordingly and align your conversion cycles with the patching plans.

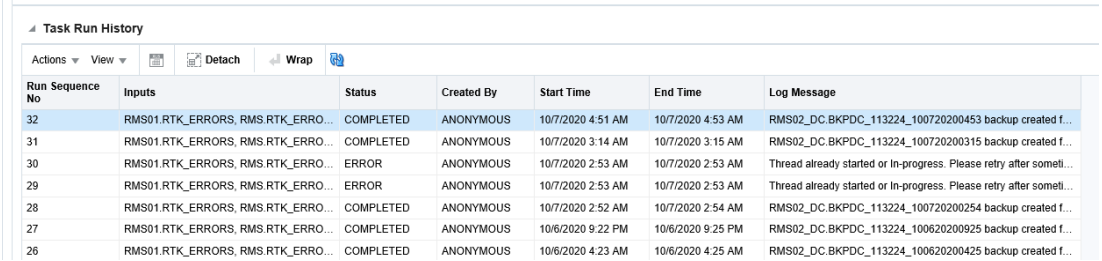
8
Task Execution Engine

Task execution engine enables you to invoke common database operations during data conversion. The tasks include gathering database stats, killing a session, taking table backups, and table purging. These operation are performed by an admin user and should be carefully invoked such that there execution does not overlap with data migration run and vice versa.

These tasks are accessed from the System Administration screen in the Data Conversion tool. From the Tasks section, select an operation and provide input parameters as required. If you need to clear the Inputs field click the refresh icon button ( ). To start the selected task, click on Submit Task button in the table toolbar. Once the task is triggered, it runs as an asynchronous process.



To check the status, click on Refresh icon( ) in the Task Run History section below. The Task Run History section shows the current and past execution history including provided inputs, excution timings and status for a task.



The following database operations are supported by this feature:

Task	Task Description
GATHER_DATABASE_STATS	This task gathers the DB statistics. This should not be executed when a conversion is under process as it may impact the performance of an ongoing data conversion run and vice versa. This can be used as an alternate to running the GATHER_STATS job from POM. Input Format: Not required
KILL_SESSION	This task kills a data conversion session in the conversion schema. It will not impact uses in the Merchandising schema. You can get the session information by querying the V_DC_SESSION_INFO view using the APEX Data Viewer by prefixing the data conversion schema name. Input Format: <SID>;<SERIAL#>;<INST_ID> All three values are to be provided separated by semicolon, for example: 1;1;1
PURGE_TABLE	This task truncates data conversion or Merchandising tables which are not seed tables. You should ensure tables are truncated as per their dependencies (that is, the child tables are truncated before the parent). Input Format: <SCHEMA>.<TABLE_NAME> Input is the table name prefixed with the schema name. For example, RMS01.ITEM_LOC can be given as input to truncate ITEM_LOC table of Merchandising schema RMS01.
BACKUP _TABLE	This can be used to backup any Merchandising or data conversion table if you wish to save a copy for lookup before you truncate/purge. The table is saved in the conversion schema with a generated name prefixed with "BKPDC". The generated table name would be available in the Log Message in the Task Run History section of the System Administration screen. You can query the generated table's data using the APEX Data Viewer by prefixing the data conversion schema name. Input format: <SCHEMA_NAME>.TABLE_NAME Multiple tables can be given as input with comma as separator, for example: RMS01.ITEM_MASTER, RMS01_DC. SVC_ITEM_MASTER
ENABLE_TRIGGER	This can be used to enable one or more triggers. You can also choose to enable all triggers by passing 'ALL' as the input parameter. The trigger names are to be provided as input. Input Format: <TRIGGER_NAME1>,<TRIGGER_NAME2> Multiple triggers can be given as input with comma as separator. For example, EC_TABLE_UIF_AIUDR,EC_TABLE_UIL_AIDR,EC_TABLE_VI_AIUDR,EC_TABLE_ISU_AIUDR

Task	Task Description
DISABLE_TRIGGER	This can be used to disable one or more triggers. The trigger names are to be provided as input. Input Format: <TRIGGER_NAME1>,<TRIGGER_NAME2> Multiple triggers can be given as input with comma as separator. For example, EC_TABLE_UIF_AIUDR, EC_TABLE_UIL_AIDR, EC_TABLE_VI_AIUDR, EC_TABLE_ISU_AIUDR
HALT_DBMS_SCHEDULER_JOBS	This task will stop scheduled jobs that are currently running for data conversion. To view details of a scheduled job whether it is running or already completed, query the view V_DC_DBMS_SCHD_JOB_RUN_DETAILS using the APEX Data Viewer by prefixing the data conversion schema name. Input Format:<JOB_NAME> Multiple jobs can be given as input with comma as separator. For example, DC_ITEM_LOC_11, DC_ITEM_LOC_12, DC_ITEM_LOC_13
DROP_BACKEDUP_TABLE	This task will drop backup tables previously backed up into the data conversion schema through the BACKUP_TABLE task. Input Format: <TABLE_NAME> Multiple tables can be given as input with comma as separator. For example, BKPDC_113224_091120200259, BKPDC_113224_091120200319
DELETE TABLE STATS	This works similar to Merchandising DELETE_TAB_STATS.KSH batch. This truncates temporary tables and locks stats for performance.Input Format: Not required
DISABLE_PUBLISHING_TRIGGERS	This task disables all MFQUEUE publishing triggers. This should be run prior to the start of data conversion. To enable required publishing triggers post data conversion, execute ENABLE_TRIGGER task. Input Format: Not required
INIT_PUBLISHING	This task initializes the merchandising publishing tables. It clears MFQUEUE tables (if any data), populates PUB_INFO tables as needed and sets the published indicator to Y for the converted data. This task should be run after completing conversion of data, prior to lift and shift. Input Format: Not required

Task	Task Description
STOCKLEDGER_PROCESSING	This task initializes the starting inventory totals in the stock ledger tables based on the stock on hand conversion. Verify and align data in the following tables before you execute the task: • Period • System Variable • Week Data, Month Data and Half Data • Half Data Budget Input Format: <NO_OF_THREADS>;<RECALC_OPTION> It accepts two input parameters, • **No of threads** - This indicates the count of threads for parallel execution. • **Recalc option** - When set to 'N', it will process only records with non-zero stock on hand. This should be 'N' when the task is executed for the first time. It should be set to 'Y', if you would like to recalculate the starting inventory totals after updating the stock on-hand to zero.
UPDATE_DC_SYSTEM_OPTIONS	This task updates values in the DC_SYSTEM_OPTIONS table. The following fields can be updated using this task: • MASTER_PROCESS_ID • USE_RPM_DURING_MERCH_CONV • AUTO_GATHER_STATS • RECALCULATE_EXPENSES All these fields except MASTER_PROCESS_ID can be set 'Y' or 'N'. For the MASTER_PROCESS_ID column, the allowed value is NULL and should be used when a mass upload process is terminated using a kill session option. In this case, DC_SYSTEM_OPTIONS.MASTER_PROCESS_ID would still have the old process id and this needs to be cleaned up before starting a new mass upload process. Input Format: < FIELD_NAME>;< FIELD_VALUE > Input is field name and value separated by semicolon, for example, `MASTER_PROCESS_ID;NULL`
DC_RPM_SEED_FUTURE_RETAIL	This task seeds the future retail values in Pricing for all active item/locations in the system at the time the process is run. Verify and align item/locations before you execute the task, Input Format: <NO_OF_THREAD> It requires total number of threads for parallel execution as the input parameter.

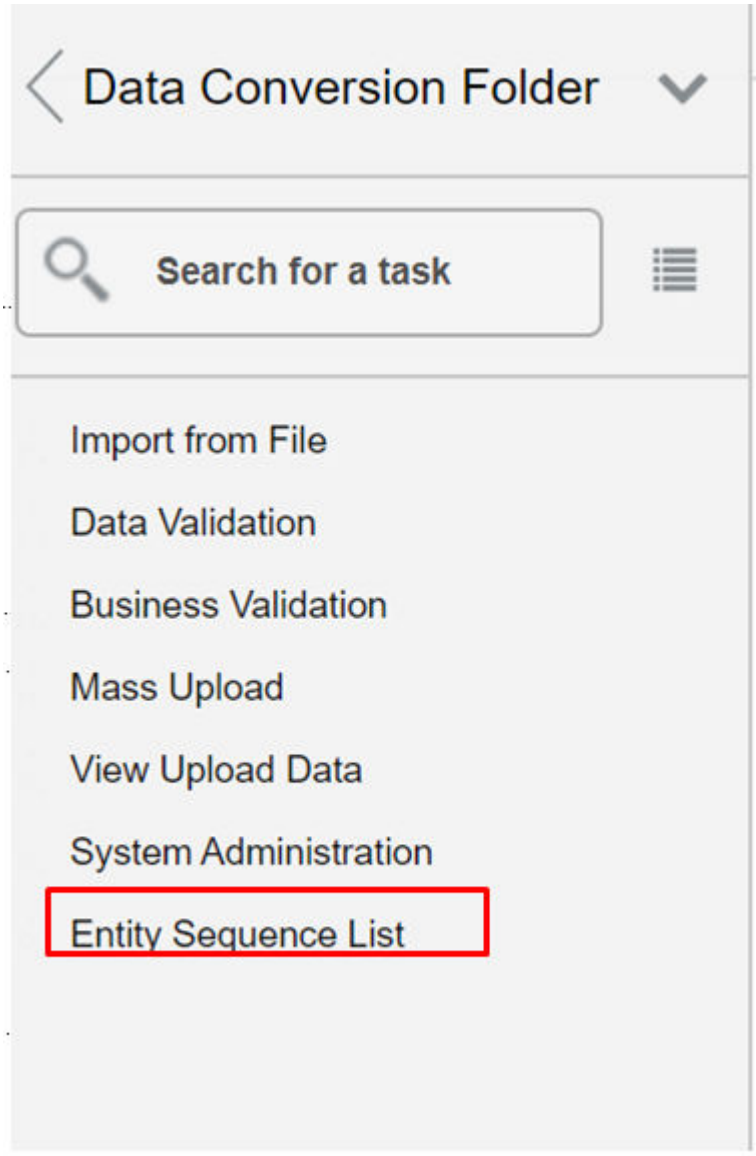
Important Notes

- The operations covered here should be invoked with caution by users with higher privileges. Some of these tasks might impact an on-going conversion run and should not be triggered when a data conversion is in progress. Similarly, conversions should not be run while these tasks are under execution.
- Avoid any spaces in the input string to these tasks.
- Maximum data length for the input string can up to 1000 characters. If required, break the activity into multiple calls to required tasks.

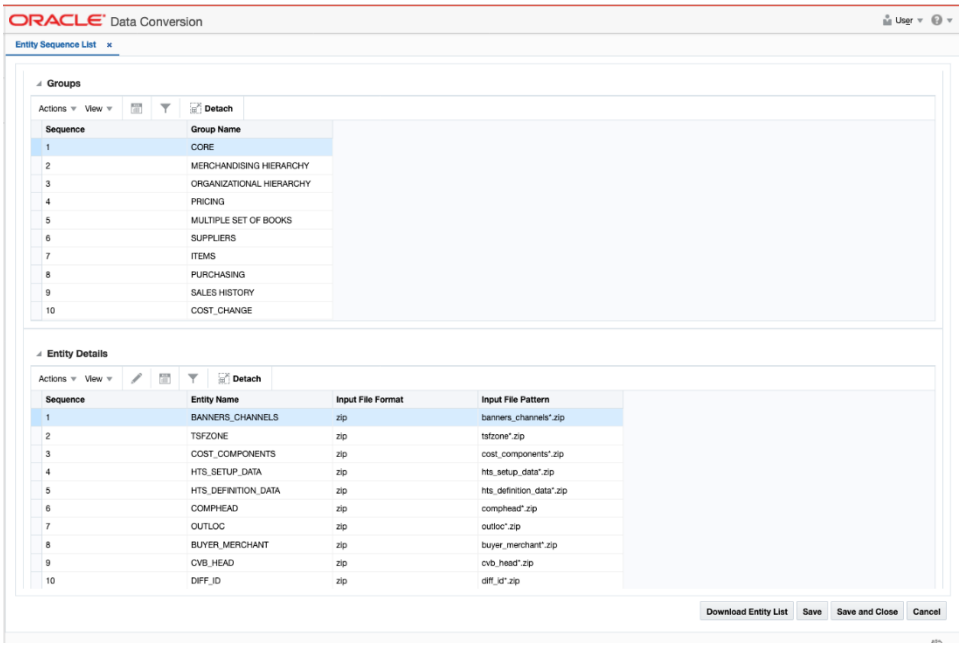
A

Appendix: Entity Sequence List

The Entity Sequence screen in the Data Conversion tool will provide you with the list of entities that can be loaded through the Data Conversion tool. The screen also lists the entity file formats, entity grouping and the load sequences. To access this screen, select Entity Sequence List from the task list.



The screen contains three tables - entity groups, entity details, and entities.

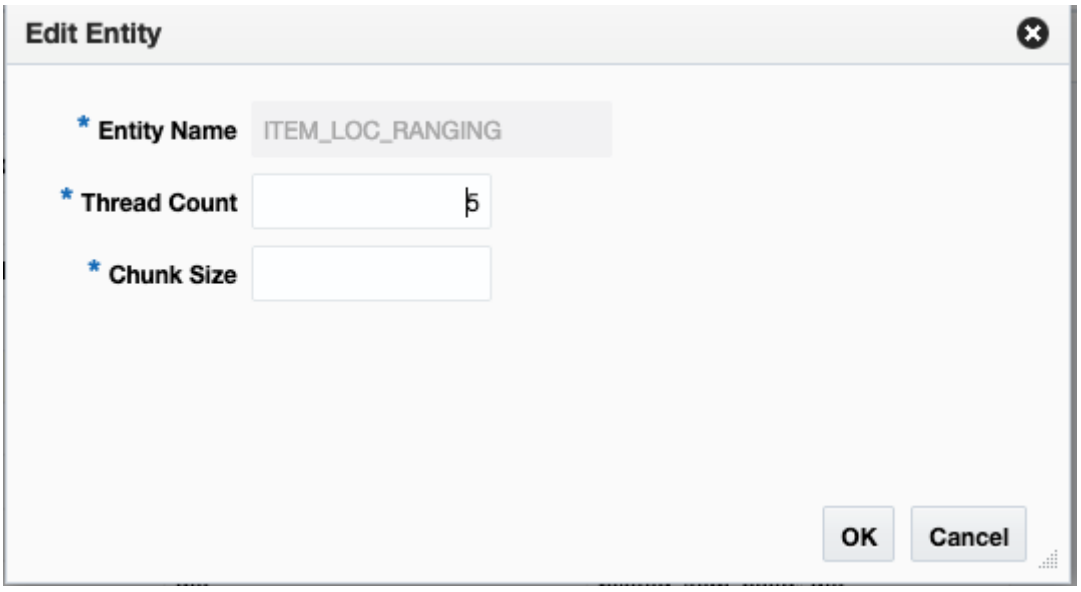


Entity Groups

The entity group table shows the groups that are used for organizing sets of tables for conversion. You will select the group in the screens when importing and validating data, as described above. Clicking on the different rows in this table will show the entity details contained in each group.

Entity Details

In the entity details table, you will be able to see the sequence within each group that the entities will be processed. Additionally, it shows the expected file naming convention for your files and the type (*.zip). Additionally, clicking the edit icon button or the Action menu item will allow you to manage the thread count and chunk sizes when processing. This may be especially important for larger data sets being loaded. It is only enabled for entities that are expected to have larger data volumes such as, item/location ranging, as shown below, and is used during the Business Validation phase. The thread count refers to the maximum number of parallel processes spawned during business validation processing for each file of that entity, and the chunk size refers to maximum number of header records that can be processed by a single thread. Clicking on the different rows in the Entity Details table will provide details about the entities represented in each file.

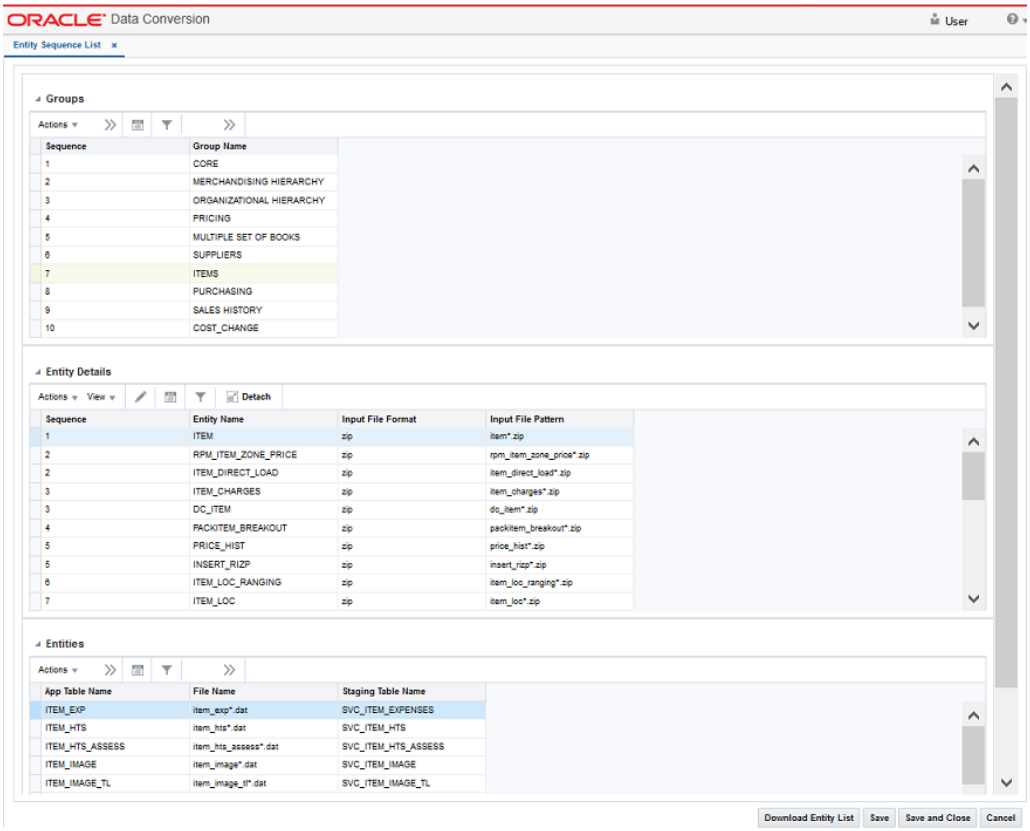


Below are a few examples on how thread count and chunk size is used when processing large datasets.

Entity	Thread Size	Chunk Size	Data Volume	Processing
CREATE_PURCHASE_ORDER	20	100	2000 Order Header records, 5000 Order Detail records	Based on this thread count and chunk size configuration, the 2000 header records are split into chunks of 100. Each such chunk of header records along with its details runs in one thread. Therefore, at one time, chunks 1 to 20 run in parallel on separate threads.
ITEM_LOC_SOH	20	10000	500K Item/Location records	Based on this thread count and chunk size configuration, 500K records are split into chunks of 10000 and, at one time, a maximum of 20 threads with each having 10000 records is processed.

Entities

This table lists each of the tables that are in the highlighted row in the Entity Details table. Only *.zip format is supported for upload. For the zip file with multiple dat files, there will be multiple tables listed that are loaded as a group. Each row in this table will show the naming convention for that specific table and the name of the staging table that is used for that entity.



The **Download Entity List** button on this page can be used to download a csv file containing all details for entity groups, entities relationship, file formats and sequencing. This may be a helpful reference to download while designing, building, and testing your conversion load programs.

B

Appendix: Offline File Validator

In order to help with the preparation of the data files, apart from the above detailed templates, an additional java-based utility is available to validate the correctness of your generated files before they are loaded into the Data Conversion Application. The jar can be downloaded onto a local machine and does not require any database to run it.

This application, along with the templates, serve as useful tools to help you in generating the right files even before the Data Conversion Application and Merchandising environments have been provisioned.

Prerequisites

- The system where this validator is loaded should have JDK (Java Development Kit) installed supporting Java 8 or higher.
- The DataConversionFileValidator.jar should be copied in the local folder of the system.
- The data should be in .dat file format (values separated by commas) in the format dictated by the templates.
- The folder that contains the input .dat files should have read and write permission.

> **Note:**
>
> This tool will identify errors with respect to data type mismatch, size mismatch and check constraint failures that are within a single file. It will not cover: primary, foreign, or unique key checks, or any other business validation. It will also not validate the file name pattern.

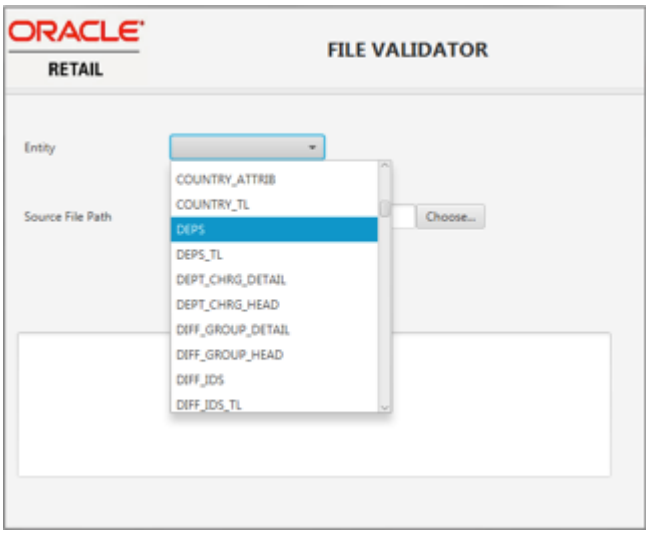
Download the Off-line File Validator

This tool is available on My Oracle Support under document ID 2538102.1. In order to download the tool, you will need a password. This password can be obtained by logging an SR and requesting the information for the relevant version of Merchandising.

Use the Off-line File Validator

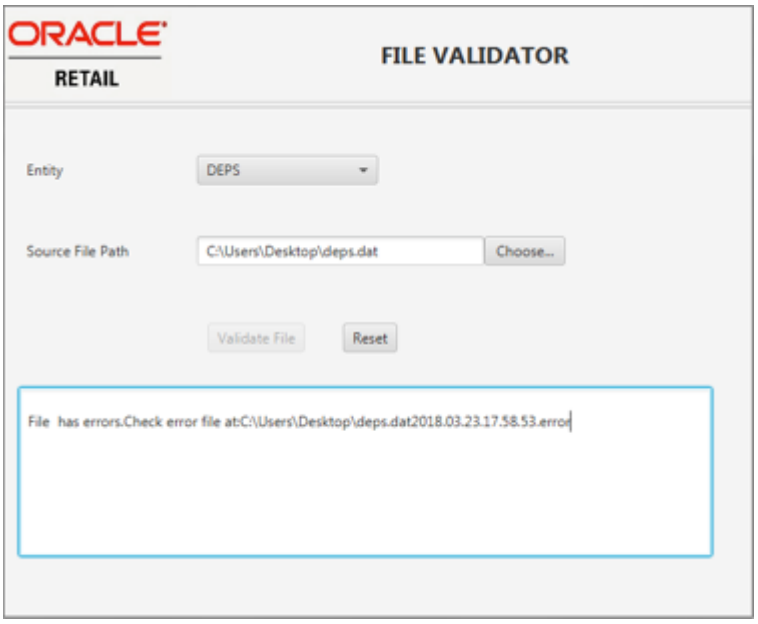
To use the Off-line File Validator tool, double click the DataConversionFileValidator.jar to run the application.

Select the entity (table) whose file you want to validate.



Choose the source file path.

Click the **Validate File** button. If the data has errors, then an error file is generated and the file path is displayed in the text area.



Below is an example of what an error file might look like.

```
1
2  Error: line No-8 Column-DEPT ErrorMessage:Field is Mandatory
3
4  Error: line No-11 Column-DEPT ErrorMessage:Invalid Number
5
6  Error: line No-14 Column-DEPT_NAME ErrorMessage:Field is Mandatory
7
8  Error: line No-18 Column-MARKUP_CALC_TYPE ErrorMessage:Field is Mandatory
9  Error: line No-18 Column-PROFIT_CALC_TYPE ErrorMessage:Size exceeded
10 Error: line No-18 Column-DEPT ErrorMessage:Invalid Number
11 Error: line No-18 Column-BUYER ErrorMessage:Invalid Number
12 Error: line No-18 Column-MAX_AVG_COUNTER ErrorMessage:Invalid Number
13
14 Error: line No-19 Column-MARKUP_CALC_TYPE ErrorMessage:Field is Mandatory
15 Error: line No-19 Column-PROFIT_CALC_TYPE ErrorMessage:Size exceeded
16 Error: line No-19 Column-BUYER ErrorMessage:Invalid Number
17 Error: line No-19 Column-MAX_AVG_COUNTER ErrorMessage:Invalid Number
18
19 Error: line No-21 Column-BUD_INT ErrorMessage:Invalid Number
20 Error: line No-21 Column-DEPT ErrorMessage:Invalid Number
21
22 Error: line No-28 Column-MERCH ErrorMessage:Size exceeded
23 Error: line No-28 Column-MAX_AVG_COUNTER ErrorMessage:Size exceeded
24
```

Bulk Offline Validation

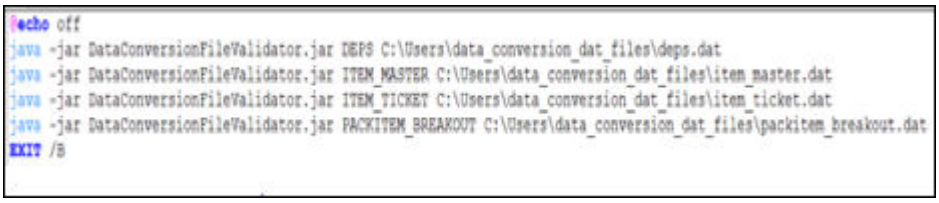
The tool also supports triggering the validation of multiple files together. This is done without using the UI, via the command prompt. To use this approach, run the jar file with arguments as:

Java -jar DataConversionFileValidator.jar <TABLE_NAME> <File_name>

This will validate the file specified.

To trigger validation of multiple files together, create a batch file containing the arguments for table name and source file path. Then in a command prompt, go to the directory where the batch file is located and run the batch file. This will execute the command in sequence without launching the application UI.

Here is an example of a batch file.



C

Appendix: Loading Replenishment Data

Currently, replenishment conversion is not support by Data Conversion tool. This section outlines different ways for loading replenishment data by utilizing core replenishment processing in RMFCS.

Manual Spreadsheet Upload

This functionality supports manual upload of replenishment information from a spreadsheet into REPL_ATTR_UPDATE* tables. An upload from a spreadsheet is initiated by user request through Merchandising screens. Post processing of data into replenishment tables including MASTER_REPL_ATTR table will be done through replenishment batch process. Records are picked by batch processing based on activation date in REPL_ATTR_UPDATE_HEAD table. The size for upload file is controlled by a system parameter, REPL_INDUCT_CONFIG.MAX_FILE_SIZE_FOR_UPLD. By default, it allows maximum file size of 500000 bytes, however this can be increased to support loads up to 2MB.

Bulk Upload Batch

Bulk integration capability is provided through the upload replenishment induction data batch process (replindbatch.ksh), which bulk uploads xml file data to REPL_ATTR_UPDATE* tables. Based on Activate Date, item/location becomes active for replenishment through replenishment batch process. Bulk upload is preferred over manual spreadsheet upload when file size is greater than 2MB. The XML format accepted by the batch will be the same as the deconstructed XML format used by manual spreadsheet upload.

The batch supports multiple files to be loaded at one time, then a single batch call can be scheduled using the Process Orchestration and Monitoring (POM) solution or batch scheduler to process all the files.

Replenishment Schedule ReST Service

This service creates scheduled replenishments to load input data to the staging tables and then calling the core replenishment package to validate and insert data to the REPL_ATTR_UPDATE* tables. Based on Activate Date, item/location becomes active for replenishment through replenishment batch process. You are required to build a script to call this rest service.

Custom Load Script

You may also choose to load replenishment data by means of custom scripts in the pre-production environment using the APEX Data Viewer. There are two ways to load data using this approach,

1. Load data directly into main merchandising replenishment tables. This requires loading data into MASTER_REPL_ATTR table using the data conversion tool and then running custom scripts to load data into other replenishment tables. Or,

2. Load data into REPL_ATTR_UPDATE* tables using custom scripts. Processing of data into replenishment tables including MASTER_REPL_ATTR table will be done through replenishment batch process.

Key Considerations

- If you wish to just load the data into REPL_ATTR_UPDATE* tables and not make these record active during the data conversion, either populate the ACTIVATE_DATE as a future date, like cut-over or go-live date, in upload files so that it becomes active from that date. Another option is after loading the data into REPL_ATTR_UPDATE* tables, do a mass update of activate and deactivate date to a date prior to the SCHEDULED_ACTIVE_DATE. Ensure DEACTIVATE_DATE should be at least ACTIVATE_DATE +1.
- Regardless of method used, the activate date must be set to the current date or later. Past dates are not supported.

D

Appendix: Best Practices

This section outlines the key recommendations while performing data conversion activities.

Data Conversion Tool

To gain better performance and seamless conversion using the data conversion tool, the following should be considered,

1. Utilize the offline data validator tool to verify input files containing transformed data. The tool helps to identify errors related to data type mismatch, size mismatch and check constraint violations. The process of validation and corrections at the source should be repeated until input files are error free.
2. Mock runs are essential in establishing the data quality and detailed migration process. It is recommended to perform at least one mock run for initial data, two or more mock runs for full data volume and at least one cutover rehearsal in preparation for the cut-over migration.
3. Golden Gate replication should be turned off during data conversion as keeping the replication enabled would severely degrade the import performance.
4. It is also suggested, before the cut-over migration, to carry out full batch cycle runs on the converted data in the system for minimum of a week to ensure that daily batch run along with week-end processing is occurring without any issues.
5. You may consider following for better performance during business validation phase:
    - For large volume entities like items, start with lower volume files during the first mock. There are often a large number of errors in the input files at the initial stages and running a lower record set will ease error identification and correction.
    - In a new instance, the DB optimizer may settle for better execution plan after 2-3 runs of execution, so continue loading the files with smaller volumes for few mock runs even if performance is not as expected.
    - Delete and collect the database stats at regular intervals while performing conversion for large volume data, such as items and item/locations. Use GATHER_DATABASE_STATS and DELETE TABLE STATS tasks in the System Administration Screen.
    - Through initial mock runs, load only new or previously failed records, excluding the records that are already processed. In this way, the system will not consume time validating and logging such records that you know to be successful.
    - Before reloading the files, ensure that you have corrected your transformation logic based on errors that you incurred in earlier mock runs so that you do not continue to encounter the same errors.
    - The migration tool is available in all non-prod environments. However, it is advisable to perform conversion in only one environment and lift-shift the converted data, if the data is required in other environments as well. This will avoid unnecessary steps of configuring additional environments for the higher volumes expected during conversion that may not be needed otherwise for that environment.

- It is highly recommended to truncate staging and error tables for an entity using the PURGE_TABLE task in the System Administration Screen at regular intervals. This will improve the execution timings for the subsequent runs. As needed, take backup of these tables before purging for future reference.

6. While loading and reloading of data, use TRUNCATE instead of DELETE for database tables, as deleting a table with larger volume data deteriorates the performance significantly.

7. Environment configuration to attain optimal performance with the data conversion tool will be reliant on expected data volume for conversion and outspread of data based on system configurations. For instance, based on the business requirements you may need to convert expenses and assessments for open purchase orders.

   The conversion environment size which includes number of CPUs, RAM, temp tablespace etc. plays a key role in determining the overall conversion performance. The thread and chunk sizes for supporting parallel processing of data should be varied to check what runs best for your data and environment. You can configure thread and chunk size through Entity Sequence List screen. For more details see the "Appendix: Entity Sequence List" section above.

   Below are the recommended configurations for thread and chunk sizes on the basis on available machine size for a few big tables converted by the data conversion tool.

Functional Area	Available CPU	Volume per load	Thread Size	Chunk Size	Parallel Files
Item	8	50K -100K	1	5000	-
	16	50K -100K	1	5000	-
	32	100K -300K	1	10000	-
	64	100K -300K	1	10000	-
ITEM_LOC RANGING - if using location level ranging	8	50K	8	50	1
	16	100K	15	50	1
	32	200K	20	5050	1
	64	200K	20		1
ITEM_LOC - for retail updates	8	1M	8	10000	2
	16	1M	15	10000	2
	32	4M	20	10000	2
	64	4M	20	10000	2
ITEM_LOC_SOH - for stock on hand updates	8	1M	8	20000	1
	16	1M	15	20000	1
	32	2M	20	50000	1
	64	2M	20	50000	1
ITEM_RANGING - for location level ranging	8	1-5M	30	1000	-
	16	1-5M	30	1000	-
	32	1-5M	30	2000	-
	64	1-5M	30	2000	-
PRICE_HIST	8	1M	8	10000	2
	16	1M	15	10000	2
	32	4M	20	10000	2
	64	4M	20	10000	2

Functional Area	Available CPU	Volume per load	Thread Size	Chunk Size	Parallel Files
VAT_ITEM	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
ITEM_HTS	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
ITEM_IMAGE	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
ITEM_EXPENSES	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
ITEM_SEASONS	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
ITEM_UDA	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
INSERT_RIZP	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
ITEM_LOC_TRAITS	8	1M	15	10000	>1
	16	1M	15	10000	>1
	32	1M	15	10000	>1
	64	1M	15	10000	>1
CREATE_PURCHASE_ORDER	8	2000	15	50	2
	16	2000	15	50	2
	32	2000	20	100	2
	64	2000	20	100	2
RPM_CLEARANCE	8	50K	8	1000	1
	16	100K	15	1000	1
	32	200K	20	1000	1
	64	200K	20	1000	1

> **Note:**
>
> For the functional area not mentioned in the above table, it is recommended to start with default configurations in the tool and, as needed, update the thread count and chunk size based on the data volume and available CPUs.

Item Conversion

For item conversion, generally higher volume data is converted and one of the key considerations during data conversion is performance. Following needs to be taken into account for item conversion,

- Break the input files at a dept/class/subclass level and not load the whole set.
- Ensure expense profiles are correctly configured to default item expenses during the item conversion. Incorrect configuration of expense profiles can result in performance issues.
- It is highly recommended to disable MFQUEUE publishing triggers during data conversion and leverage other integration options for seeding data to the downstream systems. See the section "MFQUEUE Triggers" in the appendix for the list of MFQUEUE triggers.
- If you have configured Merchandising to run with Global Tax as the default tax type, then it is advisable to complete foundation data setup before loading tax rules, as loading rules earlier may slow down the item conversion.
- If item expense and HTS information are to be loaded, it is recommended to convert only non-zero expenses or assessments in the input files for estimated value (that is, EST_EXP_VALUE and EST_ASSESS_VALUE columns). Although these values can be recalculated during data conversion, it is not advisable, as it will adversely impact the load performance. If the system is required to recalculate these column values, DC_SYSTEM_OPTIONS.RECALCULATE_EXPENSES should be set to 'Y'.
- It is highly advisable to convert items using the DC_ITEM entity which is built with improved performance capabilities. The ITEM and ITEM_DIRECT_LOAD entities which were previously used for item conversion will be discontinued in future release. There is no difference in the file structures for all these entities. The DC_ITEM entity supports certain key item-related entities be converted after items, to enhance performance of the item load by limiting the amount of data being converted in a single load. Below entities support bulk processing of data independent of item data:
  - UDAs
  - Expenses
  - VAT
  - Images
  - HTS
  - Seasons/Phases
  - Zone-level Pricing

- To enhance overall item conversion performance, the DC_ITEM entity does not support cascading of data from parent to child items. Each item level should be loaded separately with the complete set of data applicable for that level. For the same reason, inheritance and defaulting for Price History, Item/Supplier/Country/Dimension, Item/Expenses, and Pack Item Breakout tables are removed from this entity. If data conversion is needed for them, then these should be loaded in separate files.
- When using the ITEM entity, for better performance, consider limiting your file size; for instance, loading 20,000 items per file might be an acceptable file size for a 32-CPU system.
- It is highly recommended to truncate staging and error tables using the PURGE_TABLE task in the System Administration Screen after converted data is verified for each conversion cycle. This will improve the execution timings for the subsequent runs.
- For the RPM_ITEM_ZONE_PRICE conversion, it is advisable to execute the INSERT_RIZP entity, which is built with bulk validations and processing capabilities. The file structure for the INSERT_RIZP entity is the same as the RPM_ITEM_ZONE_PRICE entity. The latter would be deprecated in a future release.

Item Location Conversion

The following should be considered to achieve optimum performance during item location conversion:

- Item location conversion is a threaded process. The thread and chunk size could be varied to check what suits best for your data and environment. See the recommended configurations for thread and chunk sizes above for initial configuration.
- Only one item/location file can be converted at a time. This is to prevent database resource contention among multiple files.
- Ensure that the fields associated with consignment set up including calculation basis, purchase type, and purchase rate, match to that of item/supplier/country unless you want to explicitly range them differently at a location level. This controls the number records going into ITEM_SUPP_COUNTRY_LOC table. Wrong input data will unnecessarily write too many records into ITEM_SUPP_COUNTRY_LOC table deteriorating performance.
- If you haven't already disabled triggers that capture data for integration to downstream systems on ITEM_LOC table, this should be done as part of item/location conversion. See the section "MFQUEUE Triggers" in the appendix for the list of triggers to disable.
- Execute GATHER_DATABASE_STATS and DELETE TABLE STAT tasks in the System Administration Screen which gathers and deletes database statistics respectively. It is recommended to gather statistics multiple times a day in between conversion runs. Also, it is advisable to delete statistics for every 5-10 million item/locations processed through item location ranging.
- For item/location ranging, it is highly recommended to execute the ITEM_RANGING entity which is built with bulk validations and processing capabilities. The file structure for the ITEM_RANGING entity is different from the item ranging entity, ITEM_LOC_RANGING. The latter would be deprecated in a future release. Key features of the ITEM_RANGING entity are:
  - Pricing at item/location can be provided in the input file. If the value is not provided, it will be defaulted based on the zone level pricing.
  - Item can be ranged only at store and warehouse level in the organizational hierarchy.

  - If you are using parent/child item hierarchy, the input data will need to be added for **both** the parent and the child. Parent ranging will not be defaulted to child items using this process.
  - Item/location traits are loaded separately after item ranging is completed. This will limit the data being converted in a single process.
- If you choose to run ITEM_LOC_RANGING entity, the following points should be noted:
  - If an item is to be ranged to all stores, then consider ranging it at a higher hierarchy level (that is, chain or region).
  - Consider adding only transaction-level items to the files, which will auto-range the parent items.

Pricing Conversion

It is highly recommended for the future retail seeding and active clearances conversion you use the RPM_CLEARANCE entity and the DC_RPM_SEED_FUTURE_RETAIL tasks, respectively, in the data conversion tool. The future retail seeding and active clearances conversion can also be done by scheduling the Data Conversion Seed Future Retail batch process[1] and the Clearance Conversion processes[2], respectively, using the Process Orchestration and Monitoring (POM) solution. However, the process of scheduling the pricing conversion from POM would be deprecated in a future release. For the clearance conversion, there is no difference in the input file format for the POM job and the data conversion entity except for the file delimiter. The POM job uses | (pipe) as the field separator whereas the conversion file format uses , (comma) as the delimiter.

Purchase Orders Conversion

The following should be taken into consideration while converting purchase orders using the tool:

- It is highly recommended you close all POs that can be closed in the legacy system prior to conversion. Loading a large volume of old orders may hinder the performance of the data conversion.
- The relevant item/location records required to support the purchase orders conversion should be loaded through the item/location ranging process. If not already ranged, item/location records are created during the purchase order conversion, increasing the processing time.

1 This can be scheduled in the Process Orchestration and Monitoring (POM) solution. The job that correlates with this is RPM_SFR_DATA_CONVERSION_JOB, the process is RPM_SFR_DATA_CONVERSION_PROCESS_ADHOC.

2 This can be scheduled in the Process Orchestration and Monitoring (POM) solution. The job that correlates with this is RPM_CLR_DATA_CONVERSION_JOB, the process is RPM_CLR_DATA_CONVERSION_PROCESS_ADHOC.

E

Appendix: Conversion APIs

The conversion APIs are designed to enable conversion of certain entities into a production environment which, if converted through the Merchandising Data Conversion tool, may take longer to complete due to the volume of data being processed and require longer downtime for production systems during the phased data load. As these APIs are run in a production environment, they eliminate the need for the lift & shift activity during go-live, thus reducing the overall cutover period. When using the conversion APIs for a phased conversion, the key assumption is that the Merchandising Conversion tool was used for the initial phase of conversion. These conversion APIs are compatible with the existing file layout in the Merchandising Data Conversion tool and are designed to be run in the production environment after you have established the data quality and detailed migration process by performing multiple mock cycles and cutover rehearsal in a non-production environment.

The APIs are invoked through the batch jobs scheduled using the Process Orchestration and Monitoring (POM) solution and go through all the phases handled in the Merchandising Data Conversion tool (Import from File, Data Validation, and Business Validation).

Entities Supported

The entities below are supported for conversion into the production environment using the conversion APIs.

Functional Area	Notes
Partner	As in the Merchandising Data Conversion tool, this includes both the partner and address information. If you have configured any custom flex attributes for partner, including CFAS attributes for associated addresses, these can be converted as well. Also, if using the Retail Financials Integration (RFI) for integration with PeopleSoft, Oracle Financials, or Cloud Financials, there is a cross-reference that would also need to be maintained and needs to be considered in conversion.
Item HTS	Only applicable if Import Management functionality is configured on. See Data Entities above for information on conversion of HTS.
Item Expenses	As in the Merchandising Data Conversion tool, you can choose to convert both expenses and assessments associated with item using the conversion API.

Functional Area	Notes
Purchase Order	As in the Merchandising Data Conversion tool, the conversion API supports create, ship, and receive flows of purchase orders. To convert a closed purchase order from your legacy system to Merchandising, you will need to perform the entire flow of create, ship, and receive for purchase orders. Currently, there is no provision to directly load closed purchase orders through the data conversion API. Purchase order receipt conversion updates inventory in the ITEM_LOC_SOH table. It is assumed that the inventory at item-location level will be validated and reset during the inventory conversion to the correct values. Transactional stock ledger (TRAN_DATA) records created out of the shipping and receiving of purchase orders should be cleared by running the TRUNCTBL job from POM to truncate the records from the TRAN_DATA table.
Customer Order	As in the Merchandising Data Conversion tool, the conversion API supports the create, ship, and receive flow of customer orders.
Stock On Hand	The relevant item/location records have already been established at this phase of conversion through the item/location ranging process. If not, then this should be a pre-conversion step for stock-on-hand conversion.
Stock Ledger	Ensure that the inventory (ITEM_LOC_SOH table) conversion is completed before the stock ledger initialization job is run. If you wish to align the cumulative mark-on and shrinkage percentage in advance of the stock ledger conversion, to use that value for calculating the balances, leverage the HALF_DATA and HALF_DATA_BUDGET entities.

Key Considerations

- The conversion APIs are managed through the USE_CONV indicator in the CONV_SYSTEM_OPTIONS table. To use conversion APIs, the USE_CONV parameter should be set to Y. In the non-production environment, you can set this indicator using the APEX Data Viewer. To enable this indicator in the production environment, you are required to log an SR with the Oracle Cloud Operations team.
- You will need to do several mock runs using the Conversion APIs in a non-production environment to reach the optimum number for the thread count, chunk size, and parallel files that is suitable for your data load and machine size. Once benchmarked, to configure these in the production environment, log an SR with the Oracle Cloud Operations team. It should be noted when Golden Gate replication is turned on in the production environment, loading huge volumes of data can cause issues with replication.
- The task execution engine in the Data Conversion tool is not available in the production environment. Some of the features from the task execution engine, like gathering stats on staging tables and so on, are incorporated in the conversion APIs.
- The conversion APIs do not require you to turn off the MFQUEUE publishing triggers in the production environment. These APIs are designed in such a manner that these will not publish the MFQUEUE records for converted data. However,

data captured for integration to downstream systems through other business transactions (for example, through the MFCS UI) will continue to be published. As part of the conversion APIs, the publication tables will be loaded with published flag set to `Y` for setting up the converted data for future publication to downstream systems.

- If other systems need to be seeded with the converted data, then use other data seeding options.
- When testing in non-production environments, enable Flashback once the Mock conversion has started. This is to mimic the behavior of loading in the production environment where the flashback is always turned on.
- In a new instance, the database optimizer may settle into a better execution plan after 2-3 runs of execution, so continue loading the files with smaller volumes for a few mock runs even if performance is not as expected.
- To achieve optimal performance for data conversion using conversion APIs in your production environment, it is recommended that you avoid running heavy SQL APEX queries, conversion of parallel entities, and item induction/purchase order induction in parallel with conversion jobs.

Key Data File Assumptions

When generating your data files based on the template format, there are few key things to keep in mind:

- When zipping your files, ensure that there is only one file per table.
- The same file layouts that are used in the Data Conversion tool are extended for the conversion APIs. See the Download Templates and Key Data File Assumptions section in the Prepare for Conversion section for more details on file layout and assumptions.
- The input files for entities with the file names having the same starting string such as `HALF_DATA` and `HALF_DATA_BUDGET`, are required to be sent in a sequential manner. The files in the object storage are searched with the file name string and may result in upload errors due to an unexpected file. For example: the `HALF_DATA` file should be processed first and then the `HALF_DATA_BUDGET` file.
- Only the file extension defined for an entity in the `CONV_ENTITY` tables is accepted for processing.
- Once your data files upload to the object storage is completed, sending a `.complete` file to indicate end of file upload for an entity is not mandatory.
- The incoming prefix in the object storage for uploading the input files for Conversion APIs is different from the Data Conversion tool. Please reach out to the Oracle Cloud Operations team for instruction on the path.

Use Conversion API

Once your files have been created with the expected file name patterns and format, the Merchandising File Transfer service will be used for uploading the files to the object storage incoming prefix. You will receive instruction from the Oracle Cloud Operations team on file upload through the Merchandising File Transfer service that should be followed here. When this is complete, you will be ready to use the conversion APIs to load data from these files into the production environment. The data conversion through these APIs involves a two step process:

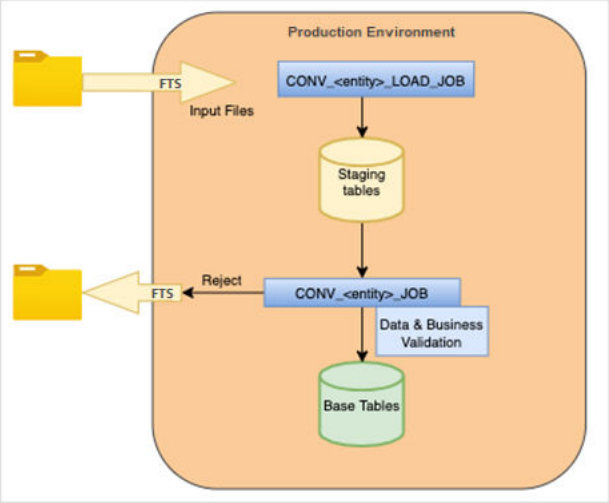
1. Load the data from the input files into the conversion staging tables in the production environment.
2. Invoke data checks and business validations to ensure the correctness of data being migrated.

The records that are successfully validated are moved to the main Merchandising tables.

For both these steps, there are separate conversion jobs for each entity that are supported for conversion in the production environment.

Execute the `CONV_ARC_PURGE_STATS_JOB_ADHOC` job from the Process Orchestration and Monitoring (POM) solution to both archive and purge the conversion staging tables and gather stats on the Merchandising main tables for the entities supported in the conversion APIs. It is highly recommended to truncate staging and error tables at regular intervals for an entity using this job. This will improve the execution timings for the subsequent runs.

The diagram below shows, at a high level, how the conversion schema is set up in relation to your production environment.



Loading to Staging Tables

The conversion load jobs load the data from input files into the conversion staging tables in the production environment. There is a load job created in POM for each entity supported for the data conversion through the production conversion API. There are no input parameters required for running these jobs.

The common tables that are available for monitoring and troubleshooting any errors reported during data loading to staging tables are described below:

Table Name	Description
CONV_PROCESS_TRACKER	Contains information on the data conversion against a process ID. For data loading from input files to the staging table, it will have a status of `END_IMPORT` for success and `FAILED_IMPORT` for failure.
FILE_UPLOAD_STATUS	Holds the status of the files uploaded during import. It will have status as `S` for success and `F` for failure.
FILE_UPLOAD_LOG	When the `sqlloader` import failed, this table will be populated with log/bad data from `sqlloader` logs.

Loading to Merchandising Main Tables

The conversion job invokes data checks & business validations to ensure the correctness of data being migrated. The records that are successfully validated will be moved to the main Merchandising tables. The job picks up the process IDs where the status in the `CONV_PROCESS_TRACKER` table is `END_IMPORT` or `END_VALIDATE`. There are no input parameters required for running this job. The job also invokes the rejection process, which generates the reject files for any errors and moves them to the `OUT` directory, so that you can manually correct them and reload the records to trigger a fresh file upload against that data. There is a conversion job created in POM for each entity supported for the data conversion through the production conversion API.

The common conversion tables that are available for monitoring and troubleshooting any errors reported during data validation and upload to the main tables are described below:

Table Name	Description
CONV_PROCESS_TRACKER	Contains information on the data conversion against a process ID. It will hold an intermediate status of `END_VALIDATE`, which is updated after data validation is completed by a conversion load job. Upon completion of data upload from staging to the main tables, the process status in the `CONV_PROCESS_TRACKER` table will be either `COMPLETE_NO_REJECTS` or `COMPLETE_WITH_REJECTS`.
CONV_VALIDATION_ERROR	Holds the errors encountered during validation and upload to the Merchandising main table. In case of validation or upload errors, the record status in the corresponding conversion staging tables will be either `E` or `PE` - to indicate processed with errors or `S` or `PS` to indicate processed successfully.

Stock Ledger Initialization Job

To initialize the starting inventory totals in the stock ledger tables based on the stock-on-hand conversion in your production environment, run the `CONV_SL_INIT_JOB_ADHOC` job using POM. There is a parameter named as `ACTION_TYPE` in the `CONV_ENTITY` table for entity name `SLOPNINIT`. This can be configured based on the business need. The `ACTION_TYPE` should be set as `New` if you would like to recalculate the starting inventory totals after updating the stock on hand to zero. Any other value for this parameter would process only records with non-zero stock on hand. By default, the `ACTION_TYPE` would be `New` and, as needed, you can reset the parameter value by logging an SR with the Oracle Cloud Operations team. There are no input parameters required to run these jobs.

Backup and Purge Job

To archive and purge the conversion staging tables and gather stats on the Merchandising main tables for the entities supported in the conversion APIs, execute the `CONV_ARC_PURGE_STATS_JOB_ADHOC` job from POM.

There are two parameters for this job: `ENTITY_NAME` and `GATHER_STATS`.

- `GATHER_STATS` is an optional parameter and is set to `Y` to gather tables stats.
- `ENTITY_NAME` is a required parameter and valid values are:
  - CREATE_PURCHASE_ORDERS
  - SHIP_PURCHASE_ORDERS
  - RECEIVE_PURCHASE_ORDERS
  - CLOSE_PURCHASE_ORDERS
  - ITEM_LOC_SOH
  - PARTNER
  - ITEM_HTS
  - ITEM_EXPENSES
  - CREATE_CUSTOMER_ORDERS
  - SHIP_CUSTOMER_ORDERS
  - RECEIVE_CUSTOMER_ORDERS

Conversion Process Flow

At a high level, for each entity you wish to convert using the conversion APIs in the production environment, you need to perform the following steps:

1. Place the conversion zip file into the object storage incoming prefix.
2. Run the `CONV_<entity>_LOAD_JOB` batch process using the Process Orchestration and Monitoring (POM). The batch moves the input files from the object storage to an inbound directory for data processing and loading into the conversion staging tables. There is separate load job for each conversion entity. For example, the

CONV_PARTNER_LOAD_JOB_ADHOC job loads the partner data in the related staging table.

3. The above batch process will insert a record with a unique process ID into the CONV_PROCESS_TRACKER table. You can monitor the load processing by querying this table using the Merchandising APEX Data Viewer link. The status END_IMPORT in the table indicates that load was successful. If there are errors during the data load into the staging tables, then the status is recorded as FAILED_IMPORT.
4. In case of errors during the data loading into the staging tables, to debug and troubleshoot the load errors, query the FILE_UPLOAD_STATUS and FILE_UPLOAD_LOG tables. These tables will provide the details of the files loaded and corresponding load errors.
5. Run the CONV_<entity >_JOB batch process using the Process Orchestration and Monitoring (POM). This job will perform data and business validation on the imported data in the conversion staging tables and upload the successfully validated records to the main Merchandising tables based on the entity being loaded. For example, the CONV_PARTNER_JOB_ADHOC job validates and uploads the partner data to relevant Merchandising tables.
6. After running step 5, the status in the CONV_PROCESS_TRACKER table - COMPLETE_NO_REJECTS indicates a successful load into the Merchandising tables and COMPLETE_WITH_REJECTS indicates the process completed with some rejected records. The status END_VALIDATE in the CONV_PROCESS_TRACKER table is an intermediate status once the job in step 5 completes data validation and is in the process of business validation and loading to Merchandising tables.
7. The successful records are inserted into the Merchandising main tables. Any rejected records are written to the CONV_VALIDATION_ERROR table. The reject file is generated and moved to the outgoing prefix in the object storage. You can download the file to view the errored data, make relevant corrections, and reprocess the rejects.

The entities below are supported for conversion into the production environment using the conversion APIs.

ENTITY	CONVERSION LOAD JOB (TO STAGE TABLES)	CONVERSION JOB (TO MAIN TABLES)
CREATE_PURCHASE_ORDERS	CONV_PO_LOAD_JOB_ADHOC	CONV_PO_CREATE_JOB_ADHOC
SHIP_PURCHASE_ORDERS	CONV_PO_SHIP_LOAD_JOB_ADHOC	CONV_PO_SHIP_JOB_ADHOC
RECEIVE_PURCHASE_ORDERS	CONV_PO_RECEIVE_LOAD_JOB_ADHOC	CONV_PO_RECEIVE_JOB_ADHOC
CLOSE_PURCHASE_ORDERS	CONV_PO_CLOSE_LOAD_JOB_ADHOC	CONV_PO_CLOSE_JOB_ADHOC
ITEM_HTS	CONV_ITEM_HTS_LOAD_JOB_ADHOC	CONV_ITEM_HTS_JOB_ADHOC
ITEM_EXPENSES	CONV_ITEM_EXP_LOAD_JOB_ADHOC	CONV_ITEM_EXP_JOB_ADHOC
ITEM_LOC_SOH	CONV_ITEM_LOC_SOH_LOAD_JOB_ADHOC	CONV_ITEM_LOC_SOH_JOB_ADHOC
PARTNER	CONV_PARTNER_LOAD_JOB_ADHOC	CONV_PARTNER_JOB_ADHOC
CREATE_CUSTOMER_ORDERS	CONV_ORDCUST_LOAD_JOB_ADHOC	CONV_ORDCUST_CREATE_JOB_ADHOC

ENTITY	CONVERSION LOAD JOB (TO STAGE TABLES)	CONVERSION JOB (TO MAIN TABLES)
SHIP_CUSTOMER_ORDERS	CONV_ORDCUST_SHIP_LOAD_JOB_ADHOC	CONV_ORDCUST_SHIP_JOB_ADHOC
RECEIVE_CUSTOMER_ORDERS	CONV_ORDCUST_RECEIVE_LOAD_JOB_ADHOC	CONV_ORDCUST_RECEIVE_JOB_ADHOC
HALF_DATA	CONV_HALF_DATA_LOAD_JOB_ADHOC	CONV_HALF_DATA_JOB_ADHOC
HALF_DATA_BUDGET	CONV_HALF_DATA_BUDGET_LOAD_JOB_ADHOC	CONV_HALF_DATA_BUDGET_JOB_ADHOC
SLOPNINIT	-	CONV_SL_INIT_JOB_ADHOC

F

Appendix: Publication Tables and Triggers

MFQUEUE Triggers

The following table contains the list of MFQUEUE triggers in Merchandising, as well as the queue tables that are written to if the triggers are not disabled. As noted above, you **should** disable these triggers prior to converting Merchandising data and re-enable once conversion is complete. If triggers are not disabled, then the queue tables must be cleared prior to lift and shift.

Merchandising Base Table	Trigger Name	Queue Table
ADDR	EC_TABLE_ADR_AIUDR	SUPPLIER_MFQUEUE
	DE_TABLE_ADDR_AIUDR	WH_MFQUEUE
		STORE_MFQUEUE
		PARTNER_MFQUEUE
ADDR_CFA_EXT	EC_TABLE_ACE_AIUR	SUPPLIER_MFQUEUE
		WH_MFQUEUE
		STORE_MFQUEUE
		PARTNER_MFQUEUE
ALLOC_DETAIL	EC_TABLE_ALD_AIUDR	ALLOC_MFQUEUE
ALLOC_HEADER	EC_TABLE_ALH_AIUDR	ALLOC_MFQUEUE
AREA	EC_TABLE_ARE_AIUDR	ORGHIER_MFQUEUE
	DE_TABLE_AREA_AIUDR	
BANNER	EC_TABLE_BAN_AIUDR	BANNER_MFQUEUE
BRAND	EC_TABLE_BRAND_AIUDR	BRAND_MFQUEUE
CHAIN	EC_TABLE_CHA_AIUDR	ORGHIER_MFQUEUE
	DE_TABLE_CHA_AIUDR	
CHANNELS	EC_TABLE_CHN_AIUDR	BANNER_MFQUEUE
CLASS	EC_TABLE_CLA_AIUDR	MERCHHIER_MFQUEUE
	DE_TABLE_CLS_AIUDR	
CLASS_CFA_EXT	EC_TABLE_CCE_AIUR	MERCHHIER_MFQUEUE
CODE_DETAIL	EC_TABLE_CODEDTL_AIUDR	CODES_MFQUEUE
CODE_HEAD	EC_TABLE_CODEHD_AIUDR	CODES_MFQUEUE
COMPANY_CLOSED	EC_TABLE_CO_CLOSE_AIUDR	COMPANY_CLOSED_MFQUEUE
COMPANY_CLOSED_EXCEP	EC_TABLE_CCEXP_AIUDR	COMPANY_CLOSED_EXCEP_MFQUEUE
COMPHEAD	DE_TABLE_CO_AIUR	
COUNTRY	EC_TABLE_CNT_AIUDR	SEEDOBJ_MFQUEUE
DELIVERY_SLOT	EC_TABLE_DLVY_AIUDR	DELIVERY_SLOT_MFQUEUE

Merchandising Base Table	Trigger Name	Queue Table
DEPS	EC_TABLE_DEP_AIUDR DE_TABLE_DEPT_AIUDR	MERCHHIER_MFQUEUE
DEPS_CFA_EXT	EC_TABLE_DCE_AIUR	MERCHHIER_MFQUEUE
DIFF_GROUP_DETAIL	EC_TABLE_DGD_AIUDR DE_TABLE_DGDTL_AIUDR	DIFFGRP_MFQUEUE
DIFF_GROUP_HEAD	EC_TABLE_DGH_AIUDR DE_TABLE_DGHEAD_AIUDR	DIFFGRP_MFQUEUE
DIFF_IDS	EC_TABLE_DID_AIUDR DE_TABLE_DIFF_AIUDR	DIFFID_MFQUEUE
DIFF_TYPE	EC_TABLE_DIFF_TYPE_AIUDR	DIFFTYPE_MFQUEUE
DISTRICT	EC_TABLE_DIS_AIUDR DE_TABLE_DIS_AIUDR	ORGHIER_MFQUEUE
DIVISION	EC_TABLE_DIV_AIUDR DE_TABLE_DIVI_AIUDR	MERCHHIER_MFQUEUE
GROUPS	EC_TABLE_GRO_AIUDR DE_TABLE_GRP_AIUDR	MERCHHIER_MFQUEUE
ITEM_IMAGE	EC_TABLE_IIM_AIUDR	ITEM_MFQUEUE
ITEM_LOC	EC_TABLE_ITL_AIUDR DE_TABLE_IL_AIUDR	ITEMLOC_MFQUEUE
ITEM_LOC_CFA_EXT	EC_TABLE_ILE_AIUR	ITEMLOC_MFQUEUE
ITEM_LOC_SOH	EC_TABLE_ILS_AUR	ITEM_LOC_SOH_MFQUEUE
ITEM_LOC_TRAITS	EC_TABLE_ILT_AIUDR DE_TABLE_ILT_AIUDR	ITEMLOC_MFQUEUE
ITEM_MASTER	EC_TABLE_IEM_AIUDR DE_TABLE_IM_AIUDR	ITEM_MFQUEUE
ITEM_MASTER_CFA_EXT	EC_TABLE_IME_AIUR	ITEM_MFQUEUE
ITEM_SUPP_COUNTRY	EC_TABLE_ISC_AIUDR	ITEM_MFQUEUE
ITEM_SUPP_COUNTRY_CFA_EXT	EC_TABLE_ICE_AIUR	ITEM_MFQUEUE
ITEM_SUPP_COUNTRY_DIM	EC_TABLE_ISD_AIUDR	ITEM_MFQUEUE
ITEM_SUPP_MANU_COUNTRY	EC_TABLE_ISMC_AIUDR	ITEM_MFQUEUE
ITEM_SUPP_UOM	EC_TABLE_ISU_AIUDR	ITEM_MFQUEUE
ITEM_SUPPLIER	EC_TABLE_ISP_AIUDR	ITEM_MFQUEUE
ITEM_SUPPLIER_CFA_EXT	EC_TABLE_ISE_AIUR	ITEM_MFQUEUE
ITEM_TICKET	EC_TABLE_ITK_AIDR	ITEM_MFQUEUE
LOCATION_CLOSED	EC_TABLE_LCLS_AIUDR	LOCATION_CLOSED_MFQUEUE
ORDCUST	EC_TABLE_ORC_AUR	TSF_MFQUEUE
ORDCUST_PUB_TEMP	EC_TABLE_ORP_AIR	
ORDHEAD	EC_TABLE_OHE_AIUDR	ORDER_MFQUEUE
ORDHEAD_CFA_EXT	EC_TABLE_OCE_AIUR	ORDER_MFQUEUE
ORDLOC	EC_TABLE_OLO_AIUDR	ORDER_MFQUEUE
ORDLOC_CFA_EXT	EC_TABLE_OLE_AIUR	ORDER_MFQUEUE

Merchandising Base Table	Trigger Name	Queue Table
ORDSKU_CFA_EXT	EC_TABLE_ORE_AIUR	ORDER_MFQUEUE
OWNERSHIP_PUB_GTT	EC_TABLE_OPT_AIR	
PACKITEM_BREAKOUT	EC_TABLE_PKS_AIUDR EC_TABLE_PKS_IUDS	ITEM_MFQUEUE
PARTNER	EC_TABLE_PRT_AIUDR	PARTNER_MFQUEUE
PARTNER_CFA_EXT	EC_TABLE_PCE_AIUR	PARTNER_MFQUEUE
PARTNER_ORG_UNIT	EC_TABLE_POU_AIUDR	SUPPLIER_MFQUEUE
PHASES	EC_TABLE_PHS_AIUDR	SEASON_MFQUEUE
REGION	EC_TABLE_REG_AIUDR DE_TABLE_REG_AIUDR	ORGHIER_MFQUEUE
RELATED_ITEM_DETAIL	EC_TABLE_RID_AIUDR DE_TABLE_RID_AIUDR	ITEM_MFQUEUE
RELATED_ITEM_HEAD	EC_TABLE_RIH_AIUDR DE_TABLE_RIH_AIUDR	ITEM_MFQUEUE
REPL_ITEM_LOC	EC_TABLE_RIL_AIUDR	ITEMLOC_MFQUEUE
RTV_DETAIL	EC_TABLE_RDT_AIUDR	RTVREQ_MFQUEUE
RTV_HEAD	EC_TABLE_RHD_AIUDR	RTVREQ_MFQUEUE
RTV_HEAD_CFA_EXT	EC_TABLE_RCE_AIUR	RTVREQ_MFQUEUE
RUA_RIB_INTERFACE	EC_TABLE_RUA_AIR	RUA_MFQUEUE
SEASONS	EC_TABLE_SSN_AIUDR	SEASON_MFQUEUE
SHIPMENT_PUB_TEMP	EC_TABLE_SPT_AIR	
STORE	EC_TABLE_STR_AIUDR DE_TABLE_STORE_AIUDR	STORE_MFQUEUE
STORE_CFA_EXT	EC_TABLE_STE_AIUR	STORE_MFQUEUE
STORE_HOURS	EC_TABLE_STHR_AIUR	STORE_MFQUEUE
SUBCLASS	EC_TABLE_SCL_AIUDR DE_TABLE_SCLS_AIUDR	MERCHHIER_MFQUEUE
SUBCLASS_CFA_EXT	EC_TABLE_SCL_AIUR	MERCHHIER_MFQUEUE
SUPS	EC_TABLE_SUP_AIUDR	SUPPLIER_MFQUEUE
SUPS_CFA_EXT	EC_TABLE_SCE_AIUR	SUPPLIER_MFQUEUE
TSFDETAIL	EC_TABLE_TDT_AIUDR	TSF_MFQUEUE
TSFHEAD	EC_TABLE_THD_AIUDR	TSF_MFQUEUE WOOUT_MFQUEUE
TSFHEAD_CFA_EXT	EC_TABLE_TCE_AIUR	TSF_MFQUEUE
UDA	EC_TABLE_UDA_AIUDR	UDA_MFQUEUE
UDA_ITEM_DATE	EC_TABLE_UIT_AIUDR	ITEM_MFQUEUE
UDA_ITEM_FF	EC_TABLE_UIF_AIUDR	ITEM_MFQUEUE
UDA_ITEM_LOV	EC_TABLE_UIL_AIDR	ITEM_MFQUEUE
UDA_VALUES	EC_TABLE_UDV_AIUDR	UDA_MFQUEUE
VAT_CODE_RATES	DE_TABLE_VCR_AIUDR	

Merchandising Base Table	Trigger Name	Queue Table
VAT_CODES	DE_TABLE_VC_AUDR	
VAT_ITEM	EC_TABLE_VI_AIUDR DE_TABLE_VI_AIUDR	ITEM_MFQUEUE
VAT_REGION	DE_TABLE_VR_AIUDR	
WH	EC_TABLE_WH_AIUDR DE_TABLE_WHOUSE_AIUDR	WH_MFQUEUE
WH_ADD	EC_TABLE_WHA_AIUR	WH_MFQUEUE
WH_CFA_EXT	EC_TABLE_WCE_AIUR	WH_MFQUEUE
WO_DETAIL	EC_TABLE_WDL_AIUDR	WOIN_MFQUEUE

Publication Tables

Below tables contain publish indicators used for setting up the converted data for future publication to downstream systems for any changes after you start using Merchandising in production. For tables that end in *_PUB_INFO or *DETAILS_PUBLISHED, records must be loaded into these tables for the converted data with a published flag set to Y. In other cases, the entity table itself (for example, ADDR) holds the publication flag. If this was not already set to Y as part of your conversion, this should be updated.

Functional Area	Publication Table
Address	ADDR
Allocations	ALLOC_PUB_INFO ALLOC_DETAILS_PUBLISHED
Items	ITEM_PUB_INFO
Partners	PARTNER_PUB_INFO
Purchase Orders	ORDER_PUB_INFO ORDER_DETAILS_PUBLISHED ORDCUST_PUB_INFO
RTVs	RTVREQ_PUB_INFO RTV_DETAIL
Shipments	SHIPMENT_PUB_INFO
Stores	STORE_PUB_INFO
Transfers	TRANSFERS_PUB_INFO TSFDETAIL ORDCUST_PUB_INFO TSF_PACKING TSF_WO_DETAIL TSF_XFORM_DETAIL WOOUT_PUB_INFO
Warehouses	WH_PUB_INFO
Purchase Order Work Orders	WO_DETAIL

> **Note:**
>
> Data Conversion for most inventory transactions is not support by the tool. It is expected transactions such as RTVs, transfers and allocations are closed in legacy before cutover. However, if you migrate inventory transaction using custom conversion scripts, you are required to clear the queue tables and load PUB_INFO* tables with the published flag set to Y.

G

Appendix: Tables List

The following table contains list of the entities and the corresponding tables supported by the Data Conversion Tool.

Entity Name	Table Name
CORE	
BANNERS_CHANNELS	BANNER
	BANNER_TL
	CHANNELS
	CHANNELS_TL
TSFZONE	TSFZONE
	TSFZONE_TL
COST_COMPONENTS	ELC_COMP
	ELC_COMP_TL
	ELC_COMP_CFA_EXT
HTS_SETUP_DATA	HTS_IMPORT_COUNTRY_SETUP
	OGA
	OGA_TL
	TARIFF_TREATMENT
	TARIFF_TREATMENT_TL
	COUNTRY_TARIFF_TREATMENT
	QUOTA_CATEGORY
	QUOTA_CATEGORY_TL
	HTS_CHAPTER
	HTS_CHAPTER_TL
	HTS_CHAPTER_RESTRAINTS
	HTS_CHAPTER_RESTRAINTS_TL
HTS_DEFINITION_DATA	HTS
	HTS_TL
	HTS_TARIFF_TREATMENT
	HTS_TT_EXCLUSIONS
	HTS_TARIFF_TREATMENT_ZONE
	HTS_FEE
	HTS_FEE_ZONE
	HTS_TAX
	HTS_TAX_ZONE
	HTS_AD

Entity Name	Table Name
	HTS_CVD
	HTS_REFERENCE
	HTS_OGA
COMPHEAD	COMPHEAD
	COMPHEAD_TL
OUTLOC	OUTLOC
	OUTLOC_TL
BUYER_MERCHANT	BUYER
	MERCHANT
CVB_HEAD	CVB_HEAD
	CVB_HEAD_TL
	CVB_DETAIL
DIFF_ID	DIFF_TYPE
	DIFF_TYPE_TL
	DIFF_IDS
	DIFF_IDS_TL
	DIFF_TYPE_CFA_EXT
FREIGHT_TERMS	FREIGHT_TERMS
	FREIGHT_TERMS_TL
	TERMS_HEAD
	TERMS_HEAD_TL
	TERMS_DETAIL
FREIGHT_TYPE	FREIGHT_SIZE
	FREIGHT_SIZE_TL
	FREIGHT_TYPE
	FREIGHT_TYPE_TL
VAT	VAT_CODES
	VAT_CODES_TL
	VAT_CODE_RATES
	VAT_REGION
	VAT_REGION_TL
	VAT_CODES_CFA_EXT
TICKET_TYPE	TICKET_TYPE_HEAD
	TICKET_TYPE_DETAIL
	TICKET_TYPE_HEAD_TL
FIF_GL_SETUP	FIF_GL_SETUP
ORG_UNIT	ORG_UNIT
	ORG_UNIT_TL
TSF	TSF_ENTITY

Entity Name	Table Name
	TSF_ENTITY_ORG_UNIT_SOB
	TSF_ENTITY_TL
BRAND	BRAND
	BRAND_TL
DIFF_RATIO	DIFF_RATIO_HEAD
	DIFF_RATIO_DETAIL
DIFF_GROUP	DIFF_GROUP_HEAD
	DIFF_GROUP_DETAIL
DIFF_RANGE	DIFF_RANGE_HEAD
	DIFF_RANGE_DETAIL
COUNTRY_ATTRIB	COUNTRY_ATTRIB
	COUNTRY_TL
CURRENCY_RATES	CURRENCIES
	CURRENCIES_TL
	CURRENCY_RATES
INV_STATUS_ADJ_REASON	INV_STATUS_TYPES
	INV_STATUS_TYPES_TL
	INV_STATUS_CODES
	INV_STATUS_CODES_TL
	INV_ADJ_REASON
FIF_GL_ACCT	FIF_GL_ACCT
MERCHANDISING HIERARCHY	
DIVISION	DIVISION
	DIVISION_TL
GROUPS	GROUPS
	GROUPS_TL
DEPS	DEPS
	VAT_DEPS
	DEPS_CFA_EXT
	DEPS_TL
CLASS	CLASS
	CLASS_CFA_EXT
	CLASS_TL
SUBCLASS	SUBCLASS
	SUBCLASS_CFA_EXT
	SUBCLASS_TL
ORGANIZATIONAL HIERARCHY	
CHAIN	CHAIN
	CHAIN_TL
AREA	AREA

Entity Name	Table Name
	AREA_TL
REGION	REGION
	REGION_TL
DISTRICT	DISTRICT
	DISTRICT_TL
STORE_FORMAT	STORE_FORMAT
	STORE_FORMAT_TL
STORE_ADD	STORE_ADD
	ADDR
	STORE_CFA_EXT
	ADDR_CFA_EXT
WAREHOUSE	WH
	ADDR
	WH_CFA_EXT
	WH_TL
	ADDR_CFA_EXT
STORE_ADD_WITH_DEFAULT_WH	STORE_ADD
	ADDR
	STORE_CFA_EXT
	ADDR_CFA_EXT
WAREHOUSE_WT_PRICING_STORE	WH
	ADDR
	WH_CFA_EXT
	WH_TL
	ADDR_CFA_EXT
TRANSIT_TIMES	TRANSIT_TIMES
WH_DEPT	WH_DEPT
	WH_DEPT_EXPL
PRICING	
RPM_ZONE_GROUP	RPM_ZONE_GROUP
RPM_MERCH_RETAIL_DEF	RPM_MERCH_RETAIL_DEF
RPM_ZONE	RPM_ZONE
RPM_ZONE_LOCATION	RPM_ZONE_LOCATION
MULTIPLE SET OF BOOKS	
PARTNER_ORG_UNIT	PARTNER_ORG_UNIT
FIF_GL_CROSS_REF	FIF_GL_CROSS_REF
SUPPLIERS	
PARTNER	PARTNER
	PARTNER_TL
	ADDR

Entity Name	Table Name
	PARTNER_CFA_EXT
	ADDR_CFA_EXT
SUPS	SUPS_TL
	SUPS
	ADDR
	SUPS_CFA_EXT
	ADDR_CFA_EXT
SUP_IMPORT_ATTR	SUP_IMPORT_ATTR
SUP_INV_MGMT	SUP_INV_MGMT
	SUP_REPL_DAY
SUP_BRACKET_COST	SUP_BRACKET_COST
SOURCE_DLVRY_SCHED	SOURCE_DLVRY_SCHED
	SOURCE_DLVRY_SCHED_DAYS
	SOURCE_DLVRY_SCHED_EXC
ITEMS	
ITEM	ITEM_MASTER
	ITEM_SUPPLIER
	ITEM_SUPP_COUNTRY
	ITEM_SUPP_COUNTRY_DIM
	ITEM_SUPP_MANU_COUNTRY
	ITEM_SUPP_UOM
	PACKITEM
	VAT_ITEM
	UDA_ITEM_LOV
	UDA_ITEM_DATE
	UDA_ITEM_FF
	ITEM_SEASONS
	ITEM_IMAGE
	ITEM_MASTER_TL
	ITEM_IMAGE_TL
	ITEM_MASTER_CFA_EXT
	ITEM_SUPPLIER_CFA_EXT
	ITEM_SUPP_COUNTRY_CFA_EXT
	ITEM_HTS
	ITEM_HTS_ASSESS
	ITEM_EXP
RPM_ITEM_ZONE_PRICE	RPM_ITEM_ZONE_PRICE
ITEM_CHARGES	ITEM_CHRG_HEAD
	ITEM_CHRG_DETAIL
DC_ITEM	ITEM_MASTER

Entity Name	Table Name
	ITEM_SUPPLIER
	ITEM_SUPP_COUNTRY
	ITEM_SUPP_COUNTRY_DIM
	ITEM_SUPP_MANU_COUNTRY
	ITEM_SUPP_UOM
	PACKITEM
	ITEM_MASTER_TL
	ITEM_MASTER_CFA_EXT
	ITEM_SUPPLIER_CFA_EXT
	ITEM_SUPP_COUNTRY_CFA_EXT
	PACKITEM_BREAKOUT
PACKITEM_BREAKOUT	PACKITEM_BREAKOUT
PRICE_HIST	PRICE_HIST
INSERT_RIZP	RPM_ITEM_ZONE_PRICE
ITEM_LOC_RANGING	NIL_DETAIL
	NIL_TRAIT
ITEM_LOC	ITEM_LOC
	ITEM_LOC_CFA_EXT
ITEM_TICKET	ITEM_TICKET
RELATED_ITEM	RELATED_ITEM_HEAD
	RELATED_ITEM_DETAIL
ITEM_UDA	UDA_ITEM_DATE
	UDA_ITEM_FF
	UDA_ITEM_LOV
VAT_ITEM	VAT_ITEM
ITEM_IMAGE	ITEM_IMAGE
	ITEM_IMAGE_TL
PACK_TEMPLATES	PACK_TMPL_HEAD
	PACK_TMPL_DETAIL
ITEM_SEASONS	ITEM_SEASONS
ITEM_EXPENSES	ITEM_EXP
SUB_ITEMS	SUB_ITEMS_HEAD
	SUB_ITEMS_DETAIL
ITEM_FORECAST	ITEM_FORECAST
ITEM_HTS	ITEM_HTS
	ITEM_HTS_ASSESS
MASTER_REPL_ATTR	MASTER_REPL_ATTR
ITEM_RANGING	NIL_DETAIL_R
ITEM_LOC_TRAITS	ITEM_LOC_TRAITS

Entity Name	Table Name
ITEM_SUPP_COUNTRY_LOC_CFA_EXT	ITEM_SUPP_COUNTRY_LOC_CFA_EXT
PURCHASING	
CREATE_PURCHASE_ORDERS	ORDHEAD
	ORDDETAIL
	ORDLC
	ORDLOC_EXP
	ORDSKU_HTS
	ORDSKU_HTS_ASSESS
	ORDHEAD_CFA_EXT
	ORDLOC_CFA_EXT
	ORDSKU_CFA_EXT
SHIP_PURCHASE_ORDERS	ASNIN_HEAD
	ASNIN_PO
	ASNIN_CTN
	ASNIN_ITEM
RECEIVE_PURCHASE_ORDERS	PO_RECEIPT_HEAD
	PO_RECEIPT_DETAIL
CLOSE_PURCHASE_ORDERS	ORDHEAD
CREATE_CUSTOMER_ORDERS	FULFILL_ORD_HEAD
	FULFILL_ORD_CUST_DTL
	FULFILL_ORD_DTL
SHIP_CUSTOMER_ORDERS	ASNOUT_HEAD
	ASNOUT_DISTRO
	ASNOUT_CTN
	ASNOUT_ITEM
RECEIVE_CUSTOMER_ORDERS	TSF_RECEIPT_HEAD
	TSF_RECEIPT_DETAIL
	TSF_RECEIPT_CTN
SALES HISTORY	
ITEM_LOC_HIST	ITEM_LOC_HIST
COST CHANGE	
COST_CHANGE	COST_SUSP_SUP_HEAD
	COST_SUSP_SUP_DETAIL
	COST_SUSP_SUP_DETAIL_LOC
PRICING CLEARANCES	
RPM_CLEARANCE	RPM_CLEARANCE_DETAIL

H

Appendix: Frequently Asked Questions

This section calls out the possible encountered issues while using the tool and suggests solutions for these.

Issue	Potential Solution
Why is my file not listed in the Import from File screen?	Validate that the file is present in the configured folder. Validate that the file pattern matches the prescribed pattern. Ensure the rules described above for data file preparations are followed.
How to achieve optimal performance for data conversion?	To obtain better performance and seamless conversion using the data conversion tool, it is highly recommended to follow the best practices specified in this document. See "Appendix: Best Practices" for more details.
Why was one of the files in the zipped file not processed?	Validate that the file pattern of the .dat file matches the prescribed pattern.
I do not wish to load data for a table that is a part of the zip file. Can it be excluded?	Yes, such files can be excluded. If there is a dependent file that has been missed in the zip file, then while running the business validations, it will be called out as errors.
I'm getting an exception while clicking on the file list in import screen. How do I resolve this issue?	Verify if the upload folder set in DC_SYSTEM_OPTIONS table matches to the configured upload folder in the app server
Why is the halt button on the Mass Upload not enabled?	This will be enabled only for certain conditions to prevent data corruption. For more details see the "Mass Upload" section above.
How do I verify if the records have actually gotten loaded into the staging tables from the screen?	The statistics on the import screen and list of errors would help identify if records were not processed.
How do I verify if records are actually loaded into main Merchandising tables?	The "View Uploaded Data" screen can be used to validate the data loaded. Additionally, the APEX Data viewer can also be used.
How do I revert/modify the records loaded in the staging tables from the UI?	You would need to correct the data in the source files and re-run as a fresh upload with the modified data without processing the earlier loaded data through to the end of the process.
How do I revert/modify the records loaded into the main tables from the UI?	Once data is loaded into the main Merchandising tables in production, the only way to modify it is via Merchandising supported methods - UIs, spreadsheet uploads, and so on. If it is still in the pre-production environment, then the APEX Data Viewer can be used to update the data as well.
If a Merchandising application is running on the same data conversion environment, can I open and work in the application?	There is nothing stopping users from opening the Merchandising applications connected to the same Merchandising schema where conversion is being done. But if any data is modified by users, then there could be instances where the sequences are not in sync with the loaded data and this could lead to errors. It is recommended all users be kept out of the Merchandising screens while running conversion.

Issue	Potential Solution
How can transactions (transfers, PO, and so on) be loaded?	At this time, purchase orders and customer orders are the only transactions that can be converted using this application. It is recommended other transactions be closed in your legacy applications prior to conversion and new transactions be created in Merchandising when needed. See also the "Data Entities" section.
How can flex attributes be converted?	To convert the data, the attributes will first need to be configured in both your stage and production environments. For more information on this functionality, see the *Customization and Extension Guide*.
Does the offline validator verify file name patterns?	No.
What is the maximum file size supported for upload?	Importing data of file size of less than 1GB is supported. However, processing large volumes takes up a lot of memory and reduces performance, especially where complex business validations are executed. Therefore, it is recommended that, in cases of large files, the data are broken up into multiple files. The maximum number of records in file supported will depend on the entity being loaded.
If we have multiple files for an entity, could the files be loaded together?	Yes. For this, place these files alone in the upload folder and use the Mass upload screen to trigger an upload at the entity group level. This will process the files placed in the Upload folder.
Can a zip file have two dat files for the same table?	No, if you have multiple files for the same table and are zipping files to load dependent tables together, you need to have one zip file per table/dat file for the dependent tables.
What will happen if I have two files with the same name?	While uploading files, the second file will overwrite the first.
If the filename does not match the pattern, what would happen?	Such files will not be listed in the import screen and the mass upload will ignore/skip the files.
How long does it take to download the templates?	This usually takes about 60 seconds, but could be longer based on network latency.
Are there performance issues during import?	If there are too many errors, then it may slow down the data processing. It is recommended to truncate the error table, DC_FILE_ERRORS, at regular intervals. Also, the offline validator should be used to verify the input files to identify errors related to data type mismatch, size mismatch and check constraint violations prior to starting the conversion process.

Issue	Potential Solution
Why does the file processing freeze at import?	Files being stuck at the Import stage is usually due to an issue with the file and not related to the performance or the data migration tool. Please ensure below points to avoid this issue, • Avoid zero byte or empty files in the zip file. • The zip file should contain only .dat files with filenames expected for the entity. • The zip file should not contain a folder inside it. • For example, the naming pattern for the DEPS table is defined as `deps*.dat`. The `*` in the name can be replaced by a number (such as `deps1.dat`, `deps2.dat`, and so on) or just left off, depending on your conversion plans • Name the zip file uniquely `ITEM*.zip`, where `*` in the name can be replaced by a number or just left off, depending on your conversion plans. For example, `ITEM12.zip` rather than `ITEM.zip` for all the files. This helps while debugging if one or more files are corrupted. • If the file has a large number of duplicates, then it could lead to locks as the records are processed in parallel. If you still run into this issue, upload the file that has encountered this issue in the SR to enable further investigation.
Why is the First line-field is rejected as INVALID NUMBER?	This would happen if the encoding of the .dat file is UTF-8 BOM instead of UTF-8. The UTF-8 BOM adds a sequence of bytes at the start of a text stream. The offline validator and the data conversion tool would reject such a starting line during import if the first field is expected to be a number. If the first field is to be a varchar2, then it would accept the junk characters and include them in the field's data, which would get caught only if there are business validations on that field which check any dependencies. You should ensure the data file encoding is UTF-8. You can change file encoding either manually through apps like Notepad++ or in the source file generation code.
What is the time benchmark for each entity?	Environment factors including sizing, data setup will impact the timings.
How do I to monitor progress of large volume entities?	You can monitor DB session for the data conversion tool using a view V_DC_SESSION_INFO. For item and purchase order entities, you can also query the following tables from the APEX Data Viewer: • SVC_PROCESS_CHUNKS • CORESVC_PO_CHUNKS
How do I check the time taken for file processing?	The DC_PROCESS_TRACKER table will hold the details for the time taken during each stage in the data conversion. The data conversion data model can be referred to understand the tables that can be looked up for analysis.
What values need to be set for thread count and chunk size on entities?	This would be based on: • Number of threads your environment can manage based on your machine size. See "Appendix: Best Practices" section above for more details. • Data volume in the import files also plays a role to define the optimum number for thread and chunk size. You may need to do a couple of mocks runs to reach the optimum number that suits your data load.

Issue	Potential Solution
How to identify and correct data when the entire chunk is rejected?	There are a few validations that can cause the entire chunk to be rejected due to errors in one or more records. To isolate and correct these records, execute the entity with a chunk size of 1.
Can I perform database operational tasks like enable or disable a trigger, kill a DB session, etc.?	The task execution engine in the data conversion tool invokes supported database operations which otherwise require assistance from the Oracle Cloud Operations team. See "Task Execution Engine" section above for more details.