Oracle® Retail Assortment Planning Cloud Service

Implementation Guide

Release 24.1.101.0
F90345-01
January 2024

ORACLE

Oracle Retail Assortment Planning Cloud Service Implementation Guide, Release 24.1.101.0

F90345-01

Primary Author: Bernadette Goodman

Contributing Authors: Venkatachalam Thiyagarajan

Contents

Send Us Your Comments

Oracle Retail Assortment Planning Cloud Service Implementation Guide, Release 24.1.101.0

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 Help Center (docs.oracle.com) 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 document provides critical information about the processing and operating details of Oracle Retail Assortment Planning Cloud Service.

Audience

This document is for:

- Systems administration and operations personnel
- Systems analysts
- Integrators and implementers
- Business analysts who need information about Oracle Retail Assortment Planning Cloud Service processes and interfaces

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For more information, see the following documents in the Oracle Retail Assortment Planning Cloud Service documentation set:

- *Oracle Retail Assortment Planning Cloud Service Administration Guide*
- *Oracle Retail Assortment Planning Cloud Service Release Readiness Guide*
- *Oracle Retail Assortment Planning Cloud Service Starter Kit Guide*
- *Oracle Retail Assortment Planning Cloud Service User Guide*

Also, see the Oracle Retail Predictive Application Server Cloud Edition documentation set.

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 Help Center (docs.oracle.com) Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

Oracle Retail documentation is available on the Oracle Help Center (docs.oracle.com) at the following URL:

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

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

If a more recent version of the 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 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

Oracle Retail Assortment Planning Cloud Service provides bottom-up item planning and performance over time which is an extension of assortment planning. These functions support proactive, in-season item management resulting in fewer lost sales, excessive markdowns, and profitable exit strategies. For a more detailed overview of the functionality within Assortment Planning Cloud Service, see the *Oracle Retail Assortment Planning Cloud Service User Guide*.

Contents of the Guide

This implementation guide addresses the following topics:

- Introduction: Overview of the Assortment Planning Cloud Service business workflow and skills needed for implementation.
- Implementation Considerations: Explanation of the factors to take into consideration before performing the implementation.
- RAP Integration: Description of RAP integration with Oracle Retail Merchandising Foundation Cloud Service (RMFCS), Retail Insights (RI), and AI Foundation (AIF).
- Appendix: Integration with MFP Cloud Service: List of measures for integration with MFP Cloud Service.
- Appendix: RMFCS Integration: Information needed for integrating with Oracle Retail Merchandising Foundation Cloud Service.
- Appendix: Standard Exports: List of measures for the available exports.
- Appendix: Application Specific Custom Functions, Procedures, and Expressions: Information on custom functions available for Assortment Planning Cloud Service.
- Appendix: APCS Scheduling in JOS/POM: Provides information about the list of jobs available for APCS.

Application Overview

Assortment Planning:

- Assortment Planning module which supports the creation of number of options, and the buy quantity calculation

Item Planning:

- Pre-season buy planning to translate assortment sales targets into actionable receipt plans
- Exception-driven in-season planning to manage key item performance
- Reconciliation to merchandise financial plans

Skills Needed for Implementation

The implementer must have an understanding of the following applications and technical concepts.

Applications

The implementer must understand interface requirements of the integrated applications and data sources for the master data, demand, and inventory history. For full implementation, the implementer requires this knowledge for the following applications:

- Oracle Retail Predictive Application Server Cloud Edition (RPASCE)
- Oracle Retail Merchandise Financial Planning Cloud Service (MFPCS)
- Oracle Retail Inventory Planning Optimization Cloud Service-Demand Forecasting (IPOCS-Demand Forecasting) (optional)
- Oracle Retail AI Foundation Cloud Services (AIF) (optional)
- Oracle Retail Merchandising Foundation Cloud Service (RMFCS)

Technical Concepts

The implementer must understand the following technical concepts:

- UNIX system administration, shell scripts, and job scheduling
- Performance constraints based on the retailer’s infrastructure
- Retailer’s hierarchical (sku/store/day) data
- Understanding of how RPASCE rule language works
- Understanding of measures and dimension constructs
- Understanding of how the RPAS Cloud Edition user interface works

2
Implementation Considerations

The following information must be considered before configuring Assortment Planning Cloud Service:

- Configuration Considerations
- Data
- Integration
- User Roles and Security
- Internationalization
- Batch Process and Scheduling

Configuration Considerations

Assortment Planning Cloud Service (APCS) contains the solutions APFA (Assortment Planning) and IPCS (Item Planning). During implementation, the user has option to extend the application configuration using Extensibility guidelines. For more details about the extensibility of the configuration, see the *Oracle Retail Analytics and Planning Implementation Guide*.

Data

APCS needs the following sets of data from retailers, which are broadly classified as hierarchy files and data files. The data is described in the following sections. Based on solutions implemented in Assortment Planning Cloud Service, only hierarchy files and data files specific for those solutions are needed and those are specified in the subsequent sections:

- Hierarchy Files
- Data Files

Hierarchy Files

This is the foundation data to build any RPASCE solution. Assortment Planning Cloud Service requires the base foundation hierarchy files, such as Calendar, Product, and Location; also, additional sets of hierarchy files specific to different solutions used in APCS. By default, APCS can get the base foundation hierarchy details as part of RAP integration. The customer only needs to upload hierarchy files which are not part of RAP integration. To load the hierarchy files during the batch process, the customer can upload their hierarchy files as individual files into Object Storage under the input directory or zip them up as hiers.zip and upload the file to the same input directory in Object Storage. All hierarchy files should have at least one valid entry, otherwise the customer will face issues in the application if the hierarchy is used in the workbook templates and if it is empty.

> **Note:**
>
> In order to implement Planning cloud services on Retail Analytics and Planning (RAP), the customer should ensure their foundation data, that is, Product and Organization hierarchies align with Oracle Retail Merchandising Foundation Cloud Service (RMFCS) so that the foundation and transactional data can be used by all services in RAP. They can have more alternate dimensions than available in RMFCS if needed for their Planning Cloud Services.
>
> Customers can use the flex fields available in RAP Foundation files to interface this data. Also, if multiple Planning cloud services such as MFPCS, APCS, and IPOCS-Demand Forecasting are residing in the same PDS, then hierarchies which are common across them should have the same dimension names so they can share the same data interfaced from RAP. However, additional non-shared dimensions can be present in each service, but shared dimensions should have the same name.

> **Note:**
>
> Hierarchy files should always contain header information and columns in any order but the file name must be in the format <hier>.hdr.csv.dat.

For information on the base hierarchy files that can be readily interfaced in RAP integration, see the following sections:

- Calendar Hierarchy File
- Product Hierarchy File
- Location Hierarchy File
- Cluster Hierarchy File
- Product Attributes Hierarchy File
- Size Hierarchy File

Calendar Hierarchy File

File name: clnd.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Name	Label	Hierarchy Type	Parent
DAY	Day	Main	None
WEEK	Week	Main	DAY
MNTH	Month	Main	WEEK
QRTR	Quarter	Main	MNTH

Name	Label	Hierarchy Type	Parent
HALF	Half	Main	QRTR
YEAR	Year	Main	HALF
HLDY	Holiday	UDA	WEEK
EVNT	Event	UDA	WEEK
WOYR	Week of Year	Alternate	WEEK
STDB	STD/BTA	UDA	WEEK
BYPD	Assortment Period	UDA	WEEK

Example:

```
day,day_label,week,week_label,mnth,mnth_label,qrtr,qrtr_label,half,half_label,year,year
_label,hldy,hldy_label,evnt,evnt_label,woyr,woyr_label,stdb,stdb_label,bypd,bypd_label
20170129,1/29/2017,w01_2017,2/4/2017,m01_2017,Feb FY2017,q01_2017,Quarter1
FY2017,h1_2017,Half1 FY2017,a2017,FY2017,0,None,0,None,1,Week 01,1,STD,1,AP1
20170130,1/30/2017,w01_2017,2/4/2017,m01_2017,Feb FY2017,q01_2017,Quarter1
FY2017,h1_2017,Half1 FY2017,a2017,FY2017,0,None,0,None,1,Week 01,1,STD,1,AP1
20170131,1/31/2017,w01_2017,2/4/2017,m01_2017,Feb FY2017,q01_2017,Quarter1
FY2017,h1_2017,Half1 FY2017,a2017,FY2017,0,None,0,None,1,Week 01,1,STD,1,AP1
20170201,2/1/2017,w01_2017,2/4/2017,m01_2017,Feb FY2017,q01_2017,Quarter1
FY2017,h1_2017,Half1 FY2017,a2017,FY2017,0,None,0,None,1,Week 01,1,STD,1,AP1
```

Note:

Though RPASCE supports a string for position IDs, for calendar position week, it is preferred to use the date format YYYYMMDD. If the customer uses RAP integration to get the data, the day and week position IDs at which the data needs to be stored are in the YYYYMMDD format.

Note:

For non-template customers, it is also recommended to align with the GA Calendar Hierarchy structure for the dimension names for day, week, mnth, qrtr, half, and year so that they can upgrade/implement other GA solutions in a multi-app environment. They can use different labels. The lowest dimension should be day and it is mandatory. The rest of the dimensions are optional and their equivalents should be named accordingly.

Product Hierarchy File

File name: prod.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Name	Label	Hierarchy Type	Parent
SKU	Item	Main	None
SKUP	Style/Color	Main	SKU
SKUG	Style	Main	SKUP
SCLS	Sub-Category	Main	SKUG
CLSS	Category	Main	SCLS

Name	Label	Hierarchy Type	Parent
DEPT	Department	Main	CLSS
PGRP	Group	Main	DEPT
DVSN	Division	Main	PGRP
CMPP	Company	Main	DVSN
STA1	Style UDA 1	UDA	SKUG
BRND	Brand	Alternate	SKU
VNDR	Vendor	Alternate	SKU

Example:

```
sku,sku_label,skup,skup_label,skug,skug_label,scls,scls_label,clss,clss_label,dept,dept_label,pgrp,pgrp_label,dvsn,dvsn_label,cmpp,cmpp_label,brnd,brnd_label,vndr,vndr_label
1000001,Lasagna,1000001,Lasagna,1000001,Lasagna,1000001,Lasagna,70000,Pasta,4000,Dry Goods,100,Shelf Stable Grocery,10,Center Store,1,Spaces Grocery,Brand,Placeholder Brand,Vendor,Placeholder Vendor
1000002,Spagetti,1000002,Spagetti,1000002,Spagetti,1000002,Spagetti,70000,Pasta,4000,Dry Goods,100,Shelf Stable Grocery,10,Center Store,1,Spaces Grocery,Brand,Placeholder Brand,Vendor,Placeholder Vendor
1000003,Rigatoni,1000003,Rigatoni,1000003,Rigatoni,1000003,Rigatoni,70000,Pasta,4000,Dry Goods,100,Shelf Stable Grocery,10,Center Store,1,Spaces Grocery,Brand,Placeholder Brand,Vendor,Placeholder Vendor
1234582,1234582 - Folgers Breakfast Roast Non-Flavored De-Caffeinated 12 oz Can,22222222,Ground De-Caffeinated Can,121212,Ground De-Caffeinated,100000,Ground,10000,Coffee,1000,Shelf Stable Beverages,100,Shelf Stable Grocery,10,Center Store,1,Spaces Grocery,Brand,Placeholder Brand,Vendor,Placeholder Vendor
1234600,1234600 - Maxwell House 100% Columbian Non-Flavored De-Caffeinated 12 oz Can,22222222,Ground De-Caffeinated Can,121212,Ground De-Caffeinated,100000,Ground,10000,Coffee,1000,Shelf Stable Beverages,100,Shelf Stable Grocery,10,Center Store,1,Spaces Grocery,Brand,Placeholder Brand,Vendor,Placeholder Vendor
```

Note:

For non-template customers, it is also recommended to align with the GA Product Hierarchy structure for the dimension names for sku, skup, skug, scls, clss, dept, pgrp, dvsn, and cmpp so that they can upgrade/implement other GA solutions in a multi-app environment. They can use different labels. The lowest dimension should be sku and it is mandatory. The rest of the dimensions are optional and their equivalents should be named accordingly.

Location Hierarchy File

File name: loc.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Name	Label	Hierarchy Type	Parent
STOR	Store	Main	None

Name	Label	Hierarchy Type	Parent
DSTR	District	Main	STOR
REGN	Region	Main	DSTR
CHNL	Area	Main	REGN
CHAN	Chain	Main	CHNL
COMP	Company	Main	CHAN
STRC	Store Cluster	Alternate	STOR
CHNC	Channel	Alternate	STRC
CCTY	Country	Alternate	CHNC
LOCT	Location Type	Alternate	STOR
PHWH	Physical Warehouse	Alternate	STOR
FFLT	Fulfillment Type	Alternate	STOR

Example:

```
STOR,STOR_LABEL,DSTR,DSTR_LABEL,REGN,REGN_LABEL,CHNL,CHNL_LABEL,CHAN,CHAN_LABEL,COMP,CO
MP_LABEL,LOCT,LOCT_LABEL,PHWH,PHWH_LABEL,FFLT,FFLT_LABEL,STRC,STRC_LABEL,CHNC,CHNC_LABE
L,CCTY,CCTY_LABEL
1000,1000 Charlotte,1070,North Carolina,170,Mid-Atlantic,1,Brick &
Mortar,1,US,1,Retailer Ltd,1,Store,WH-1,Warehouse - US,1,Brick & Mortar,1000,1000
Charlotte,1,Brick & Mortar,1,USA
1001,1001 Atlanta,1023,Georgia,400,South Atlantic,1,Brick & Mortar,1,US,1,Retailer
Ltd,2,Kiosk,WH-1,Warehouse - US,1,Brick & Mortar,1001,1001 Atlanta,1,Brick &
Mortar,1,USA
1002,1002 Dallas,1104,Texas,230,Gulf States,1,Brick & Mortar,1,US,1,Retailer
Ltd,1,Store,WH-1,Warehouse - US,1,Brick & Mortar,1002,1002 Dallas,1,Brick &
Mortar,1,USA
1003,1003 Boston,1051,Massachusetts,200,New England,1,Brick & Mortar,1,US,1,Retailer
Ltd,1,Store,WH-1,Warehouse - US,1,Brick & Mortar,1003,1003 Boston,1,Brick &
Mortar,1,USA
1004,1004 New York,1066,New York,200,New England,1,Brick & Mortar,1,US,1,Retailer
Ltd,1,Store,WH-1,Warehouse - US,1,Brick & Mortar,1004,1004 New York,1,Brick &
Mortar,1,USA004,1004 New York,1,Brick & Mortar,1,USA
```

Note:

The Store Cluster dimension (STRC) is dynamically set within the workbook. However, while loading the hierarchy file, the strc position should be loaded with the same position ID as stor and with the label as '.'. The Location clustering solution needs unique identifiers for creating store clusters and will use the unique store identifier loaded at these positions as internal identifiers for creating new clusters within the solution.

Note:

The Planning Location Hierarchy is aligned with the Merchandising Organization Hierarchy for RAP integration, so Region aggregates to Area as in the Merchandising Hierarchy. Channel is an attribute in RMFCS and is not part of the Organization Hierarchy. RMFCS integration to RAP will send the Planning Channel and Planning Country and that will be mapped to the Channel (CHNC) and Country (CCTY) dimension. Store Clusters defined within APCS or interfaced should be below this Channel level.

Note:

If the customer has warehouses holding inventory and receipts data, the Virtual Warehouse locations for each Channel can be loaded as locations with the Location type as ‘W’. The batch process allows splitting of Warehouse Inventory to locations to include in the Location Inventory for Location Planning.

Note:

For the non-template customer, it is also recommended to align with the GA Location Hierarchy structure for the dimension names for stor, dstr, regn, chnl, chan, comp, strc, chnc, and ccty so that they can upgrade/implement other GA solutions in a multi-app environment. They can use different labels. The lowest dimension should be stor and it is mandatory. The rest of the dimensions are optional and their equivalents should be named accordingly.

Cluster Hierarchy File

The cluster hierarchy is an internal application-specific hierarchy used to provide unique cluster IDs to be used during Location Clustering. It needs to be populated with unique cluster IDs (which need to be same as Store Identifiers) used in the Location hierarchy file. There is an OAT process available to synchronize this hierarchy whenever the location hierarchy file is loaded. It can also be scheduled to run on-demand, so retailers do not have to maintain this hierarchy.

Name	Label	Hierarchy Type	Aggs
CLUS	Cluster	Main	None
CHN1	Cluster Channel	Main	CLUS

File name: clrh.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Cluster	This is the unique Cluster identifiers from the Location hierarchy but with the label as '.'. The label is created dynamically and mapped to the unique ID present in this file through the Store Clustering process. The number of positions here represents the maximum pool of cluster positions available.
Cluster Channel	Channel below which the Clusters will be created. This Channel should have the same CHNC (Channel) values in the Location Hierarchy.

Example:

```
clus,clus_label,chn1,chn1_label
1000,.,1,Brick & Mortar
1001,.,1,Brick & Mortar
1002,.,1,Brick & Mortar
1003,.,1,Brick & Mortar
1004,.,1,Brick & Mortar
```

Product Attributes Hierarchy File

The product attributes hierarchy represents attributes associated with products. These attributes are used to group products within categories. This grouping is what

consumer decision trees are built on and are used when showing dynamic rollups at the item level.

This hierarchy is intended to capture all product attributes for all product types. The attributes are then assigned to individual products. This assignment is used when processing the dynamic rollups.

This hierarchy is intended to be customized for the individual retailer's needs.

Name	Label	Hierarchy Type	Aggs
PATV	Prod Attribute Value	Main	None
PATT	Prod Attribute	Main	PATV

File name: patr.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Prod Attribute Value	The various values that an attribute might have. For example, the package type attribute might take the values bag, box, or convenience.
Prod Attribute	The name of a product attribute, such as brand, family type, flavor, grain, package type, size, or temperature.

Example:

```
patv,patv_label,patt,patt_label
prodtype~basic,Basic,prodtype,Product Type
prodtype~seasonal,Seasonal,prodtype,Product Type
brand~dylanrose,Dylan Rose,brand,Brand
brand~forevercali,Forever Cali,brand,Brand
brand~legaci,Legaci,brand,Brand
```

> **Note:**
>
> PATR is used as the Attribute Hierarchy to support the 2-dimensional Product attribute measure. For detailed information on how this configuration is set up, see the *Oracle Retail Predictive Application Server Cloud Edition Configuration Tools User Guide*.

> **Note:**
>
> APCS has separate workbook flows defined for Items classified as Basic or Seasonal based on the product attribute Product Type. It is recommended to use the Product Type attribute with Basic and Seasonal attribute values for all the Items. The Basic type defines items whose selling pattern is the same across all assortment periods where the Seasonal items selling pattern varies by Season. The customer can assign any UDA to identify the basic items in RMFCS and later can assign that attribute and attribute value to define the basic items in the Planning Admin -> Batch Setup view for the Product Attribute for Basic Items and Product Attribute Value for Basic Items measures.

> **Note:**
>
> APCS uses the Nested Dynamic Rollup of Hierarchies option to review products based on the combination of various product attributes. If non-template customers want to use the same features, customization of their configuration is needed.
>
> For more details about customizing the configuration to use Nested Dynamic Rollup, see the *Oracle Retail Predictive Application Server Cloud Edition Configuration of Nested Dynamic Hierarchies Reference Paper*. It is available on My Oracle Support in the Oracle Retail Predictive Application Server (RPAS) Cloud for Planning and Optimization / Supply Chain Cloud Services Documentation Library Doc ID: 2492295.1.

Size Hierarchy File

The Size hierarchy represents different sizes associated with products. Also, different sizes are grouped by size range. Different product types by Class/Sub-class can be allowed to use different size ranges within the solution.

AP uses this size hierarchy to further plan buy quantity and receipts by different sizes for newly planned Style/Colors based on the Size Profiles either pre-defined by an Administrator or loaded from the Size Profile Optimization module.

This hierarchy is intended to be customized for the individual retailer's needs.

Name	Label	Hierarchy Type	Aggs
SIZD	Size	Main	None
SRNG	Size Range	Main	SIZD

File name: sizh.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Size	Different unique sizes such ase S, XS, M, L, XL
Size Range	Different grouping of Size Ranges such as Men's Shoes, Men's Shirt

Example:

```
sizd,sizd_label,srng,srng_label
04_0t15,4 Master,0t15master,Master 0t15
06_0t15,6 Master,0t15master,Master 0t15
08_0t15,8 Master,0t15master,Master 0t15
10_0t15,10 Master,0t15master,Master 0t15
10_5_mensshoes,10.5 Master,mensshoesmaster,Master Men's Shoes
10_5_womensshoes,10.5 Master,womensshoesmaster,Master Women's Shoes
10_mensshoes,10 Master,mensshoesmaster,Master Men's Shoes
```

Notes:

In RAP Integration with AIF, AP can get the Size hierarchy and Size Profiles from AIF or if the customer is not planning to use the SPO, they can also load the Size Hierarchy and load and use the Admin level Size Profiles.

Additional Specific Hierarchy Files

The following additional hierarchy files are also needed. They are not part of RAP integration, so the customer needs to explicitly provide the input files:

- Assortment Hierarchy File
- Cluster Source Hierarchy File
- Cluster Version Hierarchy File
- Clustering Strategy Hierarchy File
- Curve Points Hierarchy File
- Custom Messages Hierarchy File
- Location Space Hierarchy File
- Performance Group Hierarchy File
- Level Hierarchy File
- RHS Product Hierarchy File

Assortment Hierarchy File

The assortment hierarchy represents the grouping of assortments for a time period. It can be a group of weeks, months, or quarters for which an assortment is planned. This hierarchy is DPM enabled, so users can create new assortments as needed in the Assortment Maintenance workbook and assign the product/calendar association for that assortment period in that workbook.

This hierarchy is intended to be customized for the individual retailer's needs.

Name	Label	Hierarchy Type	Aggs
FLOW	Assortment	Main	None
BPER	Assortment Group	Main	FLOW
BPLB	Assortment Label	UDA	BPER
BCLS	Assortment Detail	UDA	BPER

File name: asrt.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Assortment	This defines different flow in an assortment group such as flow1, flow2, and so on.
Assortment Group	This uniquely groups the assortments for a sub-category/time frame.
Assortment Label	Assortment Label as user defined attribute given to group similar assortments using Label.
Assortment Detail	Assortment Detail as user defined attribute given to group similar assortments.

Example:

```
flow,flow_label,bper,bper_label
ap01f1,Flow 1,ap01,Assort Period 01
ap02f1,Flow 1,ap02,Assort Period 02
ap03f1,Flow 1,ap03,Assort Period 03
```

Cluster Source Hierarchy File

The cluster source hierarchy is an internal application-specific hierarchy. It should be the same as in the GA configuration and should not be changed. This hierarchy is used during wizard selection for Location Clustering to specify the source for clustering.

File name: csls.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Cluster Source	This is the unique Cluster Source identifier which can be Forecast, Plan or, Actual.

Example:

```
csor,csor_label
fcst,Forecast
plan,Plan
ty,Actual
```

Cluster Version Hierarchy File

The cluster version hierarchy is an internal application-specific hierarchy. It should be the same as in the GA configuration which contains 20 versions with 00 to 09 reserved for customer-created cluster versions within the applications and versions 10 to 20 for Loaded Clusters of different date ranges from external systems or from Advanced Clusters from AI Foundation. This hierarchy is used in Location Clustering to approve different versions of location clusters.

File name: cver.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Cluster Version	This is the unique Cluster Version identifier used during approval of a cluster.

Example:

```
vers,vers_label,vlbl,vlbl_label
01,Version 01,01,Version 01
02,Version 02,02,Version 02
03,Version 03,03,Version 03
04,Version 04,04,Version 04
05,Version 05,05,Version 05
06,Version 06,06,Version 06
07,Version 07,07,Version 07
08,Version 08,08,Version 08
09,Version 09,09,Version 09
10,Version 10,10,Version 10
11,Version 11,11,Version 11
12,Version 12,12,Version 12
13,Version 13,13,Version 13
14,Version 14,14,Version 14
15,Version 15,15,Version 15
16,Version 16,16,Version 16
17,Version 17,17,Version 17
18,Version 18,18,Version 18
19,Version 19,19,Version 19
20,Version 20,20,Version 20
```

Clustering Strategy Hierarchy File

The clustering strategy hierarchy is an internal application-specific hierarchy. The retailer can customize this hierarchy during implementation and can use the GA dataset hierarchy as a reference. This hierarchy is used to define different clustering strategies to provide different weights for metrics used during location clustering such as, Sales R and Sales U. This hierarchy is DPM enabled, so users can add more strategies dynamically while assigning strategy weights in the Planning Administration workbook.

File name: pos2.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Clustering Strategy	This is the unique Clustering Strategy to use with different combinations of metric weights to create clusters.

Example:

```
spl2,spl2_label
01,Sales R
02,Sales U
03,Sales AUR
04,GM R
05,GM R %
```

Curve Points Hierarchy File

The curve points hierarchy file is used to define unique curve libraries that can be used to define different sales curve patterns to be used during seeding in Item Planning. The retailer can customize this hierarchy during implementation and can use the GA dataset hierarchy as a reference. This hierarchy is DPM enabled, so users can add more Curve Points dynamically in Curve Setup.

File name: curv.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Curve Library	This represents different curves that can be used to define different sales patterns.

Example:

```
cnum,cnum_label
C01,Curve 01
C02,Curve 02
C03,Curve 03
C04,Curve 04
C05,Curve 05
C06,Curve 06
C07,Curve 07
C08,Curve 08
C09,Curve 09
C10,Curve 10
C11,Curve 11
C12,Curve 12
C13,Curve 13
C14,Curve 14
C15,Curve 15
C16,Curve 16
C17,Curve 17
C18,Curve 18
C19,Curve 19
C20,Curve 20
```

Custom Messages Hierarchy File

AP Cloud Service also has an additional internal hierarchy for custom messages used in the application called Custom Messages Hierarchy (CMSH). Custom messages used in the application are pre-configured in that hierarchy file and, unless a retailer needs different custom messages, that file does not need to be changed.

All custom messages are loaded as hierarchy positions to enable the translation of custom messages to different languages. It is a single dimensional hierarchy with only one dimension, CMSD. By default, all positions are loaded in English during the hierarchy load. Custom message position names are hard coded in the application, so users should not change the position names. However, during implementation, custom messages can be changed if more descriptive messages are needed.

If a user wants to change the language of custom messages, the user needs to load the provided r_cmsdlabel.csv.ovr using the standard loadmeasure utility after removing languages not needed from that file.

File name: cmsh.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Name	Label	Hierarchy Type	Parent
CMSD	Messages	Main	None

Example:

```
cmsd,cmsd_label
"ACSA01","Seed Assortment completed successfully."
"ACSA02","Warning: Select Seed Source for Assortment from WP Seed Assortment."
"ACSS01","Seed Sales completed successfully."
"ACSS02","Warning: Select WP Seed Sales to execute the Seeding!"
"ACCM01","Seed IPI Weights completed successfully."
```

Location Space Hierarchy File

The location space hierarchy is an internal application-specific hierarchy to define different location space metrics available based on which location can be clustered. The retailer can customize this hierarchy during implementation and use the GA dataset hierarchy as a reference.

File name: sspc.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Space by Location	This is the unique location metrics that can be used to define a location such as Square Meter, Avg # of Fixtures, Fixture Capacity, and so on.

Example:

```
sloc,sloc_label
sqmetr,Square Meter
sqfeet,Square Feet
avgfix,Avg # of Fixtures
avgfacings,Fixture Capacity
```

Performance Group Hierarchy File

The performance group hierarchy is an internal application-specific hierarchy to define different performance grouping (grading) to use during Location Clustering. The retailer can customize this hierarchy during implementation and use the GA dataset hierarchy as a reference. This hierarchy is DPM enabled, so users can add more performance groups if needed during location clustering.

File name: pos1.hdr.csv.dat

File format: comma-separated values file

The following table describes the fields in this file.

Field	Description
Performance Group	This is the unique performance grouping to use during clustering such as grades A, B, C, and so on.

Example:

```
clst,clst_label
01,A
02,B
03,C
04,D
05,E
```

Level Hierarchy File

The Level hierarchy is an internal application-specific hierarchy to define different levels of the Dynamic Hierarchy Rollup for Product and Location using its attributes in various workbook templates. It is hard coded to have three levels in the APCS solution.

File name: lvlh.hdr.csv.dat

File format: comma-separated values file

The following table describes the field in this file.

Name	Description
Level	Attribute Rollup Level.

Example:

```
lvld,lvld_label
lvl1,Level 1
lvl2,Level 2
lvl3,Level 3
```

RHS Product Hierarchy File

The RHS Product Hierarchy is a duplicate copy of the Product Hierarchy. It is defined as a Virtual Hierarchy using Platform features. Each dimension in the RHS Product Hierarchy is mapped to a corresponding dimension from the Product Hierarchy. It is used within AP to review Similarity Data and Demand Transference data across products in the Build Wedge process. The customer does not have to load any data for this hierarchy. Internally, the platform will create virtual positions for each position loaded into the Product Hierarchy.

Data Files

A broad and detailed data set is required to use the capabilities of APCS to its fullest.

The following tables describe the data files (measures) needed, load intersection, data type, file name, required/optional, and expected data source details. In the Data Source column, RI means any Data Warehouse or equivalent/RMS and those data are readily available from RAP integration, RSP means data from AI Foundation which is also available as part of RAP integration, Internal means any retailer internal system or the data using data files, and Admin means either data can be directly set up by an administration user or can be loaded as files.

Load Data Set

All data loads in batch after the initial domain build are done by scheduling batch tasks in Online Administration Tools. This information specifies which Load Set the user needs to use to load that particular data file while scheduling the Online Administration Tool Tasks. For more details, see the *Oracle Retail Assortment Planning Cloud Service Administration Guide*.

Table 2-1 Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
drtyeop1c	Ty EOP Reg+Promo C	real	week_sku_stor	eopx.csv.ovr	pet	Required	RI
drtyeop1r	Ty EOP Reg+Promo R	real	week_sku_stor	eopx.csv.ovr	pet	Required	RI
drtyeop1u	Ty EOP Reg+Promo U	real	week_sku_stor	eopx.csv.ovr	pet	Required	RI
drtyeop2c	Ty EOP Clr C	real	week_sku_stor	eopx.csv.ovr	pet	Required	RI
drtyeop2r	Ty EOP Clr R	real	week_sku_stor	eopx.csv.ovr	pet	Required	RI
drtyeop2u	Ty EOP Clr U	real	week_sku_stor	eopx.csv.ovr	pet	Required	RI
drtynslsclrc	Ty Net Sales Clear C	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsclrr	Ty Net Sales Clear R	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsclru	Ty Net Sales Clear U	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsproc	Ty Net Sales Promo C	real	week_sku_stor	nsls.csv.ovr	total	Required	RI

Table 2-1 (Cont.) Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
drtynslspror	Ty Net Sales Promo R	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsprou	Ty Net Sales Promo U	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsregc	Ty Net Sales Reg C	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsregr	Ty Net Sales Reg R	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtynslsregu	Ty Net Sales Reg U	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnclrc	Ty Return Clear C	real	week_sku_stor	rtn.csv.ovr	total	Required	RI
drtyrtnclrr	Ty Return Clear R	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnclru	Ty Return Clear U	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnproc	Ty Return Promo C	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnpror	Ty Return Promo R	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnprou	Ty Return Promo U	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnregc	Ty Return Reg C	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnregr	Ty Return Reg R	real	week_sku_stor	nsls.csv.ovr	total	Required	RI
drtyrtnregu	Ty Return Reg U	real	week_sku_stor	rtn.csv.ovr	total	Required	RI
drtyooc	Ty On Order C	real	week_sku_stor	oo.csv.ovr	total	Required	RI
drtyoor	Ty On Order R	real	week_sku_stor	oo.csv.ovr	total	Required	RI
drtyoou	Ty On Order U	real	week_sku_stor	oo.csv.ovr	total	Required	RI
drtyporcptc	Ty PO Receipt C	real	week_sku_stor	rcpt.csv.ovr	total	Required	RI
drtyporcptr	Ty PO Receipt R	real	week_sku_stor	rcpt.csv.ovr	total	Required	RI
drtyporcptu	Ty PO Receipt U	real	week_sku_stor	rcpt.csv.ovr	total	Required	RI
drtytraninbc	Ty Transfers In Book C	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytraninbr	Ty Transfers In Book R	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytraninbu	Ty Transfers In Book U	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytraninic	Ty Transfers In ICT C	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytraninir	Ty Transfers In ICT R	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI

Table 2-1 (Cont.) Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
drtytraniniu	Ty Transfers In ICT U	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytraninr	Ty Transfers In R	real	week_sku_stor	tranx.csv.ovr	total	Required	RI
drtytraninc	Ty Transfers In C	real	week_sku_stor	tranx.csv.ovr	total	Required	RI
drtytraninu	Ty Transfers In U	real	week_sku_stor	tranx.csv.ovr	total	Required	RI
drtytranoutbc	Ty Transfers Out Book C	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytranoutbr	Ty Transfers Out Book R	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytranoutbu	Ty Transfers Out Book U	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytranoutic	Ty Transfers Out ICT C	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytranoutir	Ty Transfers Out ICT R	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytranoutiu	Ty Transfers Out ICT U	real	week_sku_stor	tranx.csv.ovr	total	Optional	RI
drtytranoutr	Ty Transfers Out R	real	week_sku_stor	tranx.csv.ovr	total	Required	RI
drtytranoutu	Ty Transfers Out U	real	week_sku_stor	tranx.csv.ovr	total	Required	RI
drtytranoutc	Ty Transfers Out C	real	week_sku_stor	tranx.csv.ovr	total	Required	RI
drtyicmkdr	TY Inter-Company Markdown R	real	week_sku_stor	ic_mkd.csv.ovr	total	Optional	RI
drtyicmkur	TY Inter-Company Markup R	real	week_sku_stor	ic_mkd.csv.ovr	total	Optional	RI
drtywfslsr	TY W/F Sales R	real	week_sku_stor	wfms.csv.ovr	total	Optional	RI
drtywfslsu	TY W/F Sales U	real	week_sku_stor	wfms.csv.ovr	total	Optional	RI
drtywfslsc	TY W/F Sales C	real	week_sku_stor	wfms.csv.ovr	total	Optional	RI
drtywfrtnr	TY W/F Returns R	real	week_sku_stor	wfms.csv.ovr	total	Optional	RI
drtywfrtnu	TY W/F Returns U	real	week_sku_stor	wfms.csv.ovr	total	Optional	RI
drtywfrtnc	TY W/F Returns C	real	week_sku_stor	wfms.csv.ovr	total	Optional	RI
drdvprdattt	Product Attribute - Item Level	string	sku_patt	drdvprdattt.csv.ovr	mode_pop	Required	RI
drdvppatvt	RMS Product Attribute Value	string	patv	drdvppatvt.csv.ovr	mode_pop	Required	RI

Table 2-1 (Cont.) Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
drtyudab	TY RMS UDA	Boolean	patt	drtyudab.csv.ovr	or	Required	RI
addvlocopnd	Location Open Date	date	stor	stor_a.csv.ovr	ambig_pop	Required	RI
addvlocendd	Location Close Date	date	stor	stor_a.csv.ovr	ambig_pop	Required	RI
addvlocrefd	Location Refurbish Date	date	stor	stor_a.csv.ovr	ambig_pop	Required	RI
addvloctypet	Location Type	string	stor	stor_a.csv.ovr	ambig_pop	Required	RI
drtypclsst	TY RMS Class Display Id	string	sku	prod_a.csv.ovr	ambig_pop	Required	RI
drtypsclst	TY RMS Sub-Class Id	string	sku	prod_a.csv.ovr	ambig_pop	Required	RI
addvlocattt	Location Attribute	string	stor_satt	addvlocattt.csv.ovr	mode_pop	Required	Admin
adlylagwt	LY Week Map	string	week	adlylagwt.csv.ovr	mode_pop	Optional	Admin
addvprdattb	Class - Product Attribute Eligibility	Boolean	clss_patt	addvprdattb.csv.ovr	or	Optional	Admin
addvslscrvv	Sales Curve %	real	woyr_scls_chnc_cnum	addvslscrvv.csv.ovr	total	Optional	Admin
addvslsprcc	Override Cost	real	skup_stor	addvslsprc.csv.ovr	max_pop	Optional	Admin
addvslsprcr	Override Retail Price	real	skup_stor	addvslsprc.csv.ovr	max_pop	Optional	Admin
addvslswgtu	Sales Weight U	real	chnc_spl2	addvstrcwgt.csv.ovr	average_pop	Optional	Admin
addvslswgtr	Sales Weight R	real	chnc_spl2	addvstrcwgt.csv.ovr	average_pop	Optional	Admin
addvslswgtar	Sales Weight AUR	real	chnc_spl2	addvstrcwgt.csv.ovr	average_pop	Optional	Admin
addvgmwgtr	Gross Margin Weight R	real	chnc_spl2	addvstrcwgt.csv.ovr	average_pop	Optional	Admin
addvgmwgtrp	Gross Margin Weight R %	real	chnc_spl2	addvstrcwgt.csv.ovr	mode_pop	Optional	Admin
drdvstrclust	Loaded Location Cluster	string	week_dept_stor	drdvstrclus.csv.ovr	mode_pop	Optional	AI Foundation
drdvstrclusl	Loaded Location Cluster Label	string	week_dept_stor	drdvstrclus.csv.ovr	mode_pop	Optional	AI Foundation

Table 2-1 (Cont.) Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
drdvsrtd	Start Date	date	week_dept_stor	drdvstrclus.csv.ovr	ambig_pop	Optional	AI Foundation
drdvendd	End Date	date	week_dept_stor	drdvstrclus.csv.ovr	ambig_pop	Optional	AI Foundation
drtyassrtelasv	TY Assortment Elasticity	real	scls_chnc_patt_csgd	drtyassrtelasv.csv.ovr	average_pop	Required	AI Foundation
drtyattrwgtv	TY Attribute Weight %	real	scls_chnc_patt_csgd	drtyattrwgtv.csv.ovr	average_pop	Required	AI Foundation
drtyfuncfitb	TY Functional Fit	Boolean	scls_chnc_patt_csgd	drtyattrwgtv.csv.ovr	or	Required	AI Foundation
fcdvsls1u	Fcst Sales Reg+Promo U	real	week_scls_stor	fcst_scls.csv.ovr	total	Optional	AI Foundation
fcdvsls1r	Fcst Sales Reg+Promo R	real	week_scls_stor	fcst_scls.csv.ovr	total	Optional	AI Foundation
fctyfcpmu	Fcst Pre-Season Sales U	real	week_sku_stor	fcst.csv.ovr	total	Required	AI Foundation
fctyfcimu	Fcst In-Season Sales U	real	week_sku_stor	fcst.csv.ovr	total	Required	AI Foundation
fctyfcpmr	Fcst Pre-Season Sales R	real	week_sku_stor	fcst.csv.ovr	total	Required	AI Foundation
fctyfcimr	Fcst In-Season Sales R	real	week_sku_stor	fcst.csv.ovr	total	Required	AI Foundation
mlcpeopc	MFP Loaded CP EOP C	real	week_scls_stor	mfp_mpcp.csv.ovr	pet	Required	MFP
mlcpeopr	MFP Loaded CP EOP R	real	week_scls_stor	mfp_mpcp.csv.ovr	pet	Required	MFP
mlcpeopu	MFP Loaded CP EOP U	real	week_scls_stor	mfp_mpcp.csv.ovr	pet	Required	MFP
mlcprcptc	MFP Loaded CP Receipts C	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcprcptr	MFP Loaded CP Receipts R	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcprcptu	MFP Loaded CP Receipts U	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP

Table 2-1 (Cont.) Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
mlcprtn1r	MFP Loaded CP Returns Reg+Promo R	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcprtn1u	MFP Loaded CP Returns Reg+Promo U	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcprtn2r	MFP Loaded CP Returns Clear R	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcprtn2u	MFP Loaded CP Returns Clear U	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcpsls1r	MFP Loaded CP Sales Reg+Promo R	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcpsls1u	MFP Loaded CP Sales Reg+Promo U	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcpsls2r	MFP Loaded CP Sales Clr R	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcpsls2u	MFP Loaded CP Sales Clr U	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlcpslsc	MFP Loaded CP Sales Reg+Promo C	real	week_scls_stor	mfp_mpcp.csv.ovr	total	Required	MFP
mlwpooadjc	MFP Loaded WP On Order Adj C	real	week_scls_stor	mfp_otb.csv.rpl	total	Required	MFP
mlwpooadjr	MFP Loaded WP On Order Adj R	real	week_scls_stor	mfp_otb.csv.rpl	total	Required	MFP
mlwpooadju	MFP Loaded WP On Order Adj U	real	week_scls_stor	mfp_otb.csv.rpl	total	Required	MFP
mlwpotbc	MFP Loaded WP OTB C	real	week_scls_stor	mfp_otb.csv.rpl	total	Required	MFP
mlwpotbr	MFP Loaded WP OTB R	real	week_scls_stor	mfp_otb.csv.rpl	total	Required	MFP
mlwpotbu	MFP Loaded WP OTB U	real	week_scls_stor	mfp_otb.csv.rpl	total	Required	MFP
lplaeopc	LP AP EOP C	real	week_dept_stor	mfp_lpap.csv.ovr	pet	Optional	MFP
lplaeopr	LP AP EOP R	real	week_dept_stor	mfp_lpap.csv.ovr	pet	Optional	MFP
lplaeopu	LP AP EOP U	real	week_dept_stor	mfp_lpap.csv.ovr	pet	Optional	MFP
lplarcptc	LP AP Receipts C	real	week_dept_stor	mfp_lpap.csv.ovr	total	Optional	MFP

Table 2-1 (Cont.) Assortment Planning Cloud Service Measure List - Details 1

Measure Name	Measure Label	Data Type	Load Intersection	File Name	Agg Type	Required or Optional?	Data Source
lplarcptr	LP AP Receipts R	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
lplarcptu	LP AP Receipts U	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
lplartnr	LP AP Returns R	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
lplartnu	LP AP Returns U	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
lplaslsu	LP AP Sales U	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
lplaslsr	LP AP Sales R	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
lplaslsc	LP AP Sales C	real	week_dept_stor	mfp_ lpap.csv.ovr	total	Optional	MFP
addvpskugt	Rename Style Id	string	skug	addvpskugt.csv.ovr	mode_pop	Optional	Admin
addvpskupt	Rename Style/ Color Id	string	skup	addvpskugt.csv.ovr	mode_pop	Optional	Admin
addvpskut	Rename Item Id	string	sku	addvpskugt.csv.ovr	mode_pop	Optional	Admin
adwpsizeprfp	Admin Size Profile %	real	scls_stor_sizd	adwpsizeprfp.csv.ovr	max	Optional	Admin
drdvsizeprfp	SPO Size Profile %	real	scls_stor_sizd	drdvsizeprfp.csv.ovr	max	Optional	AI Foundation
addvskuimgt	Item Image	string	sku	addvskuimgt.csv.ovr	mode_pop	Optional	Admin
addvskupimgt	Style-Color Image	string	skup	addvskupimgt.csv.ovr	mode_pop	Optional	Admin
addvskugimgt	Style Image	string	skug	addvskugimgt.csv.ovr	mode_pop	Optional	Admin
addvsclsimgt	Sub-Class Image	string	scls	addvsclsimgt.csv.ovr	mode_pop	Optional	Admin
addvpatvimgt	Product Attribute Value Image	string	patv	addvpatvimgt.csv.ovr	mode_pop	Optional	Admin
addvpattimgt	Product Attribute Image	string	patt	addvpattimgt.csv.ovr	mode_pop	Optional	Admin
drdvskuimgt	Item Image Name	string	sku	drdvskuimg.csv.ovr	mode_pop	Optional	RI
drdvskuimgl	Item Image Address	string	sku	drdvskuimg.csv.ovr	mode_pop	Optional	RI

All measure files that need to be loaded as data files need to be grouped based on the File Name. The files should contain the header for the measures to be loaded and it should be

in .csv format. Measures within a file can be grouped in any order as long as the header column is specified correctly. If a measure is optional in a file, the customer can ignore that measure and group the remaining measures which are available for the customer.

Example:

In following example, the customer is using RAP integration and only grouping the data that is not coming in RAP (or RI) in a file for which customer has the data.

File Name: tranx.csv.ovr

Base Intersection: week/sku/stor

Data Type: real

```
week,sku,stor,drtyroyalr,drtymiscadju,drtymiscadjr,drtycogsr
w01_2021,100000,1000,30.96,31.52,0,0
w02_2021,100000,1000,169.13,112.61,1,37.85
w03_2021,100000,1000,233.54,50.26,1,35.09
```

Historical Data

It is recommended that you have at least one full year of historical data to create in Assortment Planning Cloud Service. Less data can be used, but the more data that is available, the more statistical significance can be given to the plan data.

By default, RAP integration is set up to interface two years of history into Planning.

Loading and Extracting Data

Data is loaded into Assortment Planning Cloud Service using the Online Administration Tools, which in turn use standard RPAS utilities. For more information on loading and extracting data using Online Administration Tools, see the *Oracle Retail Assortment Planning Cloud Service Administration Guide*.

Loading Image Based Data

Assortment Planning Cloud Service is pre-configured to provide the item level image view in the templates. The measure set up as the Style-Color level image attribute is addvskupimgt with the base intersection of Style/Color and product attributes images to addvpatvimgt.

The Content Server exposes the client's image files placed into a particular directory as HTTP URLs. The images available in http://{content server url}/imgfetch/image-library/{sub directory if defined}/<image-file-name> must be defined in the load file in xml format.

Sample file for addvskupimgt.csv.ovr:

```
1234582,"<image id=""main"" label=""Front View""><url size=""thumb"">http://
<server>:<port>/<image_path>/sku_10000019_main_thumb.jpg</url></image>"
1234600,"<image id=""main"" label=""Front View""><url size=""thumb"">http://
<server>:<port>/<image_path>/sku_10000053_main_thumb.jpg</url></image>"
```

The first field represents the Style Color ID followed by the required image location. At a minimum, a "thumb" size image file must be loaded to show in the pivot table. However, both the "thumb" and "full" size images can be loaded. For example:

```
10000010,"<image id=""main"" label=""Front View""><url size=""thumb"">http://
<server>:<port>/<image_path>/sku_10000010_main_thumb.jpg</url><url
size=""full"">http://<server>:<port>/<image_path>/sku_10000010_main_full.jpg</url></
image>
```

The customer can also use the same format to upload image URLs for the Item Image, Style Image, and Sub-Class Images. The same image URLs can also be directly managed in Planning Admin -> Define Product Image. They can also set the Item Image Name and Item Image Address and enable the Aggregate Boolean to create the Image URLs for different levels from the same base Item Image.

In order to view the images, the Valid Image URL Hosts property should include the Image URL server name. That needs to be set in RPASCE UI Settings -> System Configuration -> Config Properties -> Images.

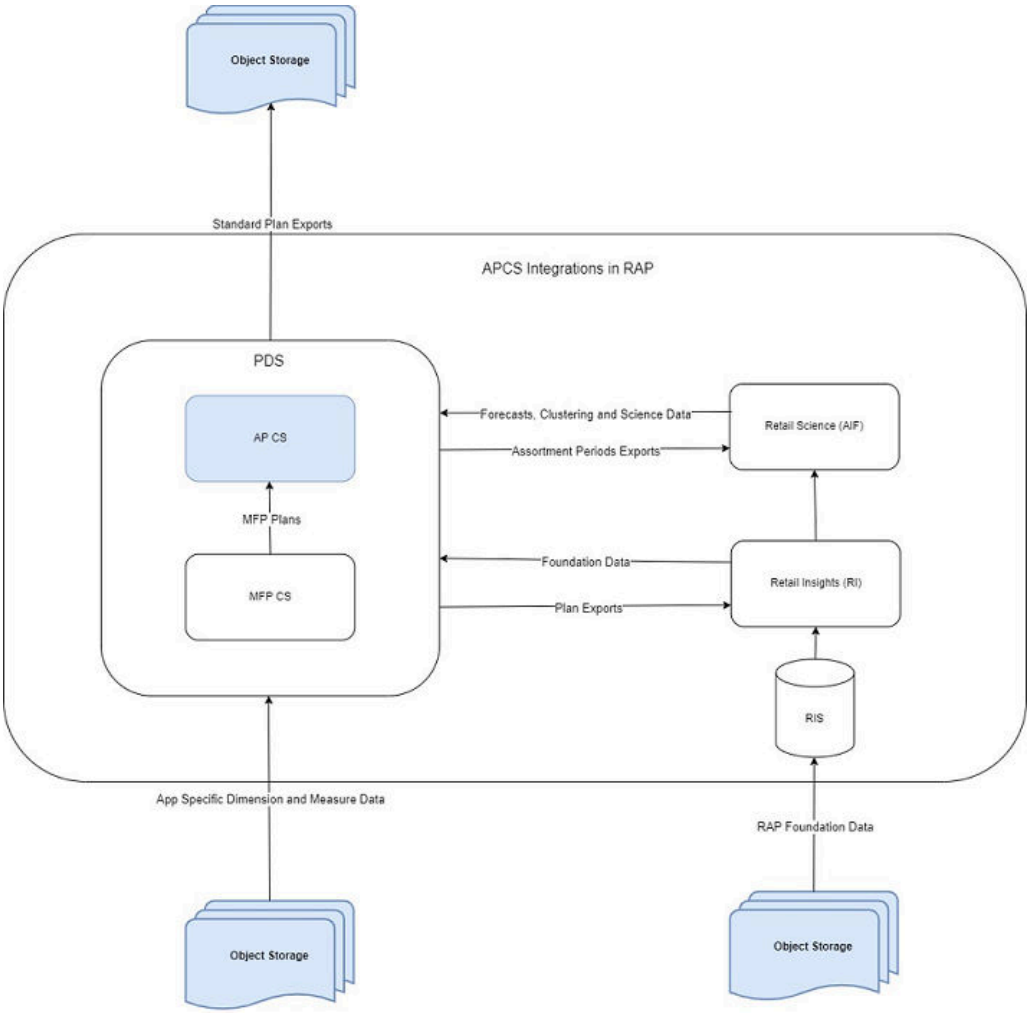
Integration

Assortment Planning Cloud Service uses RAP integration to interface with RI to get foundation data from RMFCS or other similar source systems and to get forecast and clustering data from AI Foundation (AIF). For more details about RAP integration, see RAP Integration.

Assortment Planning Cloud Service integrates with MFP Cloud Service for Merchandise Financial Plan Data to use as the Financial Target while creating assortments. For more details, see Appendix: Integration with MFP Cloud Service

If the customer is using an RMFCS version that does not use RAP integration, it can still interface using the file-based approach to interface the foundation data. For more details, see Appendix: RMFCS Integration.

Figure 2-1 Assortment Planning Cloud Service Integration



Assortment Planning Cloud Service provides some standard exports that can be used by external systems that need Assortment and Item Plan Data. For details about the standard exports from Assortment Planning Cloud Service, see Appendix: Standard Exports.

Retailers using either the template or non-template version must extract and provide the foundation files needed from other source systems as flat files in the required format as needed by RAP integration and then upload to Object Storage. Any data or hierarchy files that are specific to their Planning Solution that cannot be integrated using RAP integration can be directly uploaded to Object Storage for Planning. In the same way, exported files from the solution if not part of RAP integration are sent back to the Object Storage and retailers can download the extracted files from there. The retailer must integrate it with any other system that requires extracted plan data from APCS, if not part of RAP integration

User Roles and Security

To define workbook template security, the system administrator grants individual users, or user groups, access to specific workbook templates. Granting access to workbook

templates provides users with the ability to create, modify, save, and commit workbooks for the assigned workbook templates. Users are typically assigned to groups based on their user application (or solution) role. Users in the same group can be given access to workbook templates that belong to that group alone. Users can be assigned to more than one group and granted workbook template access without belonging to the user group that typically uses a specific workbook template. Workbook access is either denied, read-only, or full access. Read-only access allows a user to create a workbook for the template, but the user is not able to edit any values or commit the workbook. The read-only workbook can be refreshed.

The following table provides guidance regarding which Assortment Planning Cloud Service users must have access to each of the workbooks.

Table 2-2 User's Access Permission for APCS Workbooks

Workbook	User Roles
Planning Administration	Planning Administrator
Validate Loaded Data	Planning Administrator
Location Clustering	Planner, Planning Administrator
Assortment Period Setup	Planner, Planning Administrator
Curve Maintenance	Planner, Planning Administrator
Dashboard Parameters	Planner, Planning Administrator
Create Assortment	Planner
Item Planning	Planner
Item Planning - Basics	Planner

For more information on security, see the *Oracle Retail Predictive Application Server Cloud Edition Administration Guide*. For more information on data security in a cloud environment, see the Hosting Policy documents for the cloud solution.

Internationalization

Internationalization is the process of creating software that can be translated more easily. Changes to the code are not specific to any particular market.

Oracle Retail applications have been internationalized to support multiple languages.

The RPASCE platform supports associated solution extensions and solution templates:

- A solution extension includes a collection of code and generally available configurations. Typically, solution extensions are implemented by a retailer with minimal configuration.
- A solution template does not include code. A solution template is most typically implemented as a retailer configuration.

Oracle Retail releases the translations of the RPASCE server and client, as well as strings from the solution extensions.

Translations of the solution templates are not released. All templates have the ability to support multi-byte characters.

For more information on internationalization, see the *Oracle Retail Predictive Application Server Cloud Service Administration Guide*.

Translations are available for Assortment Planning Cloud Service for the following languages:

- Chinese (Simplified)
- Chinese (Traditional)
- Croatian
- Dutch
- French
- German
- Greek
- Hungarian
- Italian
- Japanese
- Korean
- Polish
- Portuguese (Brazilian)
- Russian
- Spanish
- Swedish
- Turkish

> **Note:**
>
> For information about adding languages for the first time or for translation information in general, see the *Oracle Retail Predictive Application Server Cloud Edition Administration Guide*.

Batch Process and Scheduling

Batch scripts are lists of commands or jobs executed without human intervention. A batch window is the time frame in which the batch process must run. It is the upper limit on how long the batch can take. Batch scripts are used for loading foundation data received from a merchandising system, importing and exporting data, and generating targets. The retailer must decide the best time for running batch scripts within the available batch window.

How often to upload updated sales and inventory data and how often to recreate targets must be determined.

- You must consider at what interval to load the latest sales and inventory data. A weekly load of transactional type data is supported, since the base intersection is at week. It is recommended that the information transactional system, such as RMS, be loaded daily.

- Product availability and seasonal changes can be reasons for recalculating the targets. This can also be triggered by the addition of new products and availability of substantial new sales and inventory history.

The recommended batch schedule for Assortment Planning Cloud Service is to load historical and actual data on a weekly basis. All hierarchy changes can be loaded on a weekly basis.

In Assortment Planning Cloud Service, batch tasks can be controlled by a system administrator by using the Online Administration Tools. Those tasks, in turn, call the batch scripts with preset parameters to perform the batch tasks. For more information on the Online Administration Tool tasks, see the *Oracle Retail Assortment Planning Cloud Service Administration Guide*.

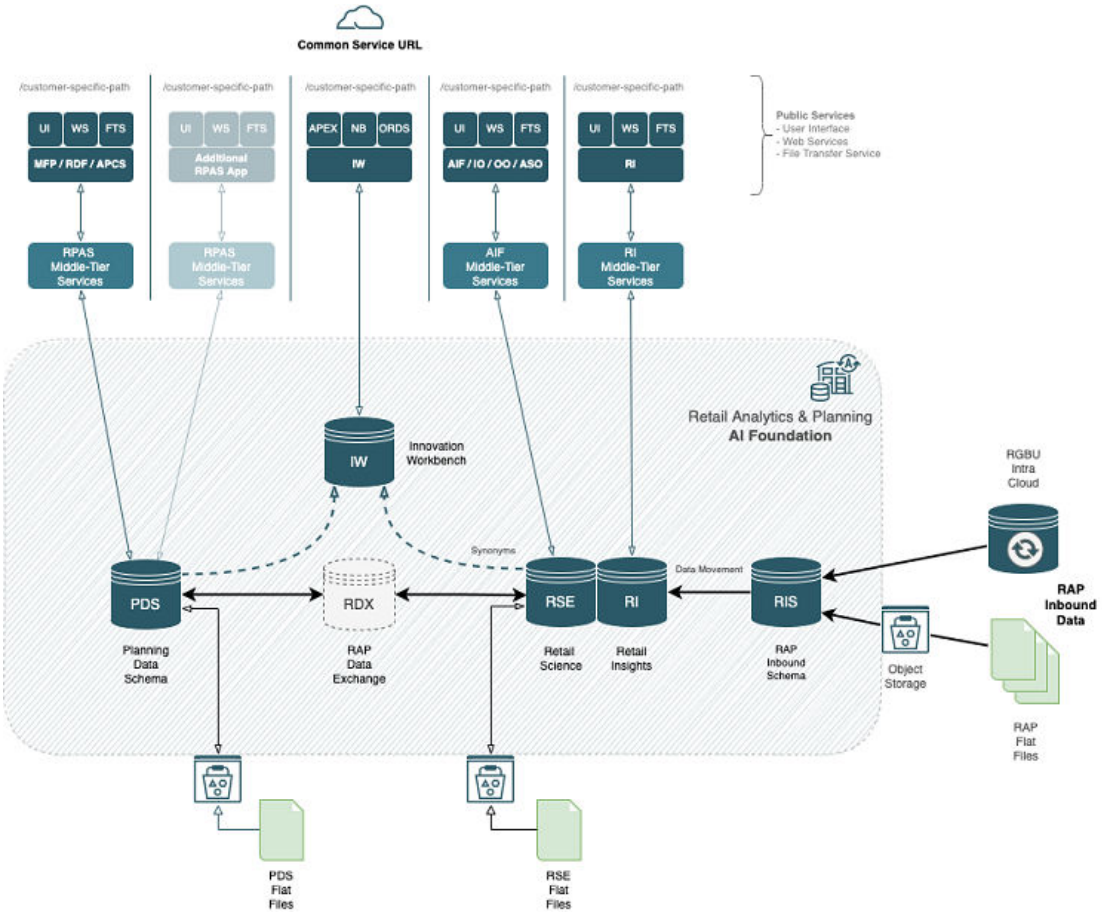
For more details about the list of batch control files, the batch process using them, and details about updating them, see the Enterprise Edition Batch framework in the *Oracle Retail Predictive Application Server Cloud Edition Implementation Guide*.

The customer can use JOS/POM if RAP integration is used and implemented to schedule pre-configured daily and weekly batch tasks in APCS. Those tasks scheduled using JOS/POM in turn call the same Configured batch tasks under the Online Administration Tool tasks. For more details about scheduling of tasks using JOS/POM, see the *Oracle Retail Predictive Application Server Cloud Service Administration Guide*. For more details about the APCS schedule in JOS/POM, see Appendix: APCS Scheduling in JOS/POM.

3
RAP Integration

Planning Solutions built on Planning Data Schema (PDS) can be integrated with AI Foundation (AIF) and Retail Insights (RI) with in Retail Analytics and Planning (RAP) using RAP integration which in turn can get the foundation data from RMFCS or any other systems integrated to RAP. All three solutions (AIF, RI, PDS) within RAP can share data using RAP Data Exchange (RDX) using RAP interfaces.

Planning Solutions, such as APCS, configured and built in PDS using the RPASCE Configuration allows configurable solutions if they are not using template to use their custom hierarchy and fact names. Planning Solutions, such as APCS, allow configuring of interfaces using **interface.cfg** which controls the mapping of dimensions and facts to columns in the RAP interface staging tables. For more details about configuring interfaces using interfaces.cfg in RAP integration, see the *Oracle Retail Predictive Application Server Cloud Edition Implementation Guide*. The APCS template version has a pre-configured interface.cfg which contains the mapping of interfaces.



The above diagram shows the high-level RAP Architecture. The customer can upload their main input files as RAP Inbound Data using Object Storage and from their Planning

Applications deployed in PDS can get the same using RDX. For more details about the RAP Inbound Interfaces, see the *Oracle Retail Analytics and Planning Implementation Guide*. Any supplemental data that is specific to planning can be directly loaded into PDS as PDS Flat Files. This section shows the details about the interfaces used by PDS in RAP using RDX.

Following is the pre-defined grouping of interfaces available in the APCS template version within RAP integration.

Pre-defined imports from RAP integration to APCS:

- Import Foundation and Transactional data from Retail Insights (RI)
- Import Forecasts from AI Foundation (AIF)
- Import DT Parameters from AI Foundation (AIF)
- Import Location Clusters from AI Foundation (AIF)
- Import Size Profiles from AI Foundation (AIF)

Pre-defined exports to RAP integration to APCS:

- Export Assortment Periods for Location Clusters from AI Foundation (AIF)
- Export Active Assortments to AI Foundation (AIF)
- Export Assortment Plans to Retail Insights (RI)

Import Foundation and Transactional Data from Retail Insights

RMFCS can send Foundation and Transactional data to RAP integration using Retail Insights (RI) and other systems within RAP. The systems can share the data, even if RMFCS is not implemented for the customer. The customer can upload the foundation and data files in the file format needed by RI in RAP integration. That way, the same data can be published to all applications within RAP. It can be done by scheduling required job flows in Retail Insights to get the foundation data from RMFCS and loading it into the staging tables present in Retail Data Exchange (RDX) from where the configured interfaces in APCS can pull the required data into Facts in Planning Data Schema (PDS) where APCS is deployed.

The customer can also load foundation data directly into RAP using the file format specified for RAP integration and using the same staging process in RI to write the data into RDX staging tables from where Planning can pull the data using standard configured interfaces. Only mapped columns specific to GA interfaces are detailed in this guide. For more details about interface file formats and the jobs flow details, see the *Oracle Retail Analytics and Planning Implementation Guide*. Also refer to those guides to find more information about the available columns in each interface staging tables in RDX sourced from RI so that customers using extensibility on template or using custom configuration (non-template) can pull the required data from RDX.

The following table shows the list of interfaces in RAP to get the foundation and transactional data:

Interface	Interface and Table Name	Interface Type
Product Hierarchy	W_PDS_PRODUCT_D	Hierarchy Importer

Interface	Interface and Table Name	Interface Type
Location Hierarchy	W_PDS_ORGANIZATION_D	Hierarchy Importer
Calendar Hierarchy	W_PDS_CALENDAR_D (VW_CLND_HIER)	Hierarchy Importer
Cluster Hierarchy	VW_CLRH_HIER (W_PDS_ORGANIZATION_D	Hierarchy Importer
Product Attribute Hierarchy	W_PDS_UDA_D	Hierarchy Importer
Sales Interface	W_PDS_SLS_IT_LC_WK_A	Data Importer
Inventory Interface	W_PDS_INV_IT_LC_WK_A	Data Importer
On Order Interface	W_PDS_PO_ONORD_IT_LC_WK_A	Data Importer
Receipts Interface	W_PDS_INVRC_IT_LC_WK_A	Data Importer
Inventory Transfers	W_PDS_INVTSF_IT_LC_WK_A	Data Importer
Markdowns Interface	W_PDS_MKDN_IT_LC_WK_A	Data Importer
Wholesale/Franchise	W_PDS_SLSWF_IT_LC_WK_A	Data Importer
Product Attributes	W_PDS_PRODUCT_ATTR_D	Data Importer
Location Data	VW_LOC_DATA	Data Importer
Product Data	VW_PROD_DATA	Data Importer
UDA Data	VW_UDA_DATA	Data Importer

The following table shows the mapping of dimensions to columns for Hierarchy Importer interfaces from external interface tables:

Hierarchy	Dimension	External Interface Table	External Mapped Column
prod	sku	W_PDS_PRODUCT_D	ITEM
prod	sku_label	W_PDS_PRODUCT_D	ITEM_DESC
prod	skup	W_PDS_PRODUCT_D	ITEM_PARENT_DIFF
prod	skup_label	W_PDS_PRODUCT_D	ITEM_PARENT_DIFF_DESC
prod	skug	W_PDS_PRODUCT_D	ITEM_PARENT
prod	skug_label	W_PDS_PRODUCT_D	ITEM_PARENT_DESC
prod	scls	W_PDS_PRODUCT_D	CLASS_ID
prod	scls_label	W_PDS_PRODUCT_D	DEPT
prod	clss	W_PDS_PRODUCT_D	GROUP_NO
prod	class_label	W_PDS_PRODUCT_D	DIVISION
prod	dept	W_PDS_PRODUCT_D	COMPANY
prod	dept_label	W_PDS_PRODUCT_D	CO_NAME
prod	pgrp	W_PDS_PRODUCT_D	GROUP_NO
prod	pgrp_label	W_PDS_PRODUCT_D	GROUP_NAME
prod	dvsn	W_PDS_PRODUCT_D	DIVISION
prod	dvsn_label	W_PDS_PRODUCT_D	DIV_NAME
prod	cmpp	W_PDS_PRODUCT_D	COMPANY
prod	cmpp_label	W_PDS_PRODUCT_D	CO_NAME
prod	vndr	W_PDS_PRODUCT_D	SUPPLIER

Hierarchy	Dimension	External Interface Table	External Mapped Column
prod	vndr_label	W_PDS_PRODUCT_D	SUP_NAME
prod	brnd	W_PDS_PRODUCT_D	BRAND_NAME
prod	brnd_label	W_PDS_PRODUCT_D	BRAND_DESCRIPTION
prod	sta1	W_PDS_PRODUCT_D	NA
prod	sta1_label	W_PDS_PRODUCT_D	Unassigned
loc	stor	W_PDS_ORGANIZATION_D	LOCATION
loc	stor_label	W_PDS_ORGANIZATION_D	LOC_NAME
loc	dstr	W_PDS_ORGANIZATION_D	DISTRICT
loc	dstr_label	W_PDS_ORGANIZATION_D	DISTRICT_NAME
loc	regn	W_PDS_ORGANIZATION_D	REGION
loc	regn_label	W_PDS_ORGANIZATION_D	REGION_NAME
loc	chnl	W_PDS_ORGANIZATION_D	AREA
loc	chnl_label	W_PDS_ORGANIZATION_D	AREA_NAME
loc	chan	W_PDS_ORGANIZATION_D	CHAIN
loc	chan_label	W_PDS_ORGANIZATION_D	CHAIN_NAME
loc	comp	W_PDS_ORGANIZATION_D	COMPANY
loc	comp_label	W_PDS_ORGANIZATION_D	CO_NAME
loc	phwh	W_PDS_ORGANIZATION_D	PHYSICAL_WH
loc	phwh_label	W_PDS_ORGANIZATION_D	PHYSICAL_WH_NAME
loc	loct	W_PDS_ORGANIZATION_D	LOC_TYPE
loc	loct_label	W_PDS_ORGANIZATION_D	LOC_TYPE_NAME
loc	strc	W_PDS_ORGANIZATION_D	LOCATION
loc	strc_label	W_PDS_ORGANIZATION_D	LOCATION_NAME
loc	chnc	W_PDS_ORGANIZATION_D	PLANNING_CHANNEL_ID
loc	chnc_label	W_PDS_ORGANIZATION_D	PLANNING_CHANNEL_NAME
loc	ccty	W_PDS_ORGANIZATION_D	PLANNING_COUNTRY_ID
loc	ccty_label	W_PDS_ORGANIZATION_D	PLANNING_COUNTRY_NAME
clnd	day	W_PDS_CALENDAR_D	DAY
clnd	day_label	W_PDS_CALENDAR_D	DAY_LABEL
clnd	week	W_PDS_CALENDAR_D	WEEK
clnd	week_label	W_PDS_CALENDAR_D	WEEK_LABEL
clnd	mnth	W_PDS_CALENDAR_D	MNTH
clnd	mnth_label	W_PDS_CALENDAR_D	MNTH_LABEL
clnd	qrtr	W_PDS_CALENDAR_D	QRTR
clnd	qrtr_label	W_PDS_CALENDAR_D	QRTR_LABEL
clnd	half	W_PDS_CALENDAR_D	HALF
clnd	half_label	W_PDS_CALENDAR_D	HALF_LABEL
clnd	year	W_PDS_CALENDAR_D	YEAR
clnd	year_label	W_PDS_CALENDAR_D	YEAR_LABEL

Hierarchy	Dimension	External Interface Table	External Mapped Column
clnd	woyr	W_PDS_CALENDAR_D	WOYR
clnd	woyr_label	W_PDS_CALENDAR_D	WOYR_LABEL
clnd	stdb	W_PDS_CALENDAR_D	STDB
clnd	stdb_label	W_PDS_CALENDAR_D	STDB_LABEL
clnd	hldy	W_PDS_CALENDAR_D	NA
clnd	hldy_label	W_PDS_CALENDAR_D	Unassigned
clnd	evnt	W_PDS_CALENDAR_D	NA
clnd	evnt_label	W_PDS_CALENDAR_D	Unassigned
clnd	bypd	W_PDS_CALENDAR_D	NA
clnd	bypd_label	W_PDS_CALENDAR_D	Unassigned
clrh	clus	VW_CLRH_HIER	CLUS_ID
clrh	clus_label	VW_CLRH_HIER	CLUS_DESC
patr	patv	W_PDS_UDA_D (VW_PATR_HIER)	PROD_ATTR_VALUE
patr	patv_label	W_PDS_UDA_D (VW_PATR_HIER)	PROD_ATTR_VALUE_DESC
patr	patt	W_PDS_UDA_D (VW_PATR_HIER)	PROD_ATTR
patr	patt_label	W_PDS_UDA_D (VW_PATR_HIER)	PROD_ATTR_DESC

Note: For Calendar Hierarchy (clnd), RMFCS is not sending the labels. Internally, VW_CLND_HIER is defined in PDS against the interface W_PDS_CALENDAR_D table to derive the labels and also default the calendar import to PDS to have two past years, one current year, and two future years based on the current business date. The Administrator can update the same using the Online Administration Tool Tasks under System Admin Tasks -> List/Set/Unset PDS Integration variables and can update the CLND_PAST_YEARS and CLND_FUTURE_YEARS variables. By default, both are set to 2. The customer can also update the start fiscal month by setting the CLND_START_MONTH variable. By default, it is set to 2 to have the fiscal start month label be generated as February.

Note: For Cluster Hierarchy (clrh), there is no direct interface table. Internally, VW_CLRH_HIER is defined in PDS against the interface W_PDS_ORGANIZATION_D table to get the locations as cluster ids.

Note: The VW_PATR_HIER view is an internal view in PDS against the base RDX tables W_PDS_UDA_D, W_PDS_DIFF_D, W_PDS_SUPPLIER_D, and W_PDS_BRAND_D by concatenating all of them as product attributes. It also concatenates the product attribute name with the product attribute values using '_' to make the product attribute values unique. The Product Attribute name for Supplier (W_PDS_SUPPLIER_D) is used as 'supp' and Brand (W_PDS_BRAND_D) is used as 'brnd'. Only Product attributes with UDA_TYPE_CODE as 'LV' from W_PDS_UDA_D are included in the view.

Note: For all APCS hierarchies that are not integrated using RAP integration, the customer needs to explicitly provide those files.

The following table shows the mapping of fact names/measures names to columns for the Data Importer interfaces from the external interface tables in RDX:

Fact Name	External Interface Table	External Mapped Column	External Mapping Condition
drtyeop1c	W_PDS_INV_IT_LC_WK_A	REGULAR_INVENTORY_COST	CLEAR_IND = 'N'
drtyeop1r	W_PDS_INV_IT_LC_WK_A	REGULAR_INVENTORY_RETAIL	CLEAR_IND = 'N'
drtyeop1u	W_PDS_INV_IT_LC_WK_A	REGULAR_INVENTORY_UNITS	CLEAR_IND = 'N'
drtyeop2c	W_PDS_INV_IT_LC_WK_A	REGULAR_INVENTORY_COST	CLEAR_IND = 'Y'
drtyeop2r	W_PDS_INV_IT_LC_WK_A	REGULAR_INVENTORY_RETAIL	CLEAR_IND = 'Y'
drtyeop2u	W_PDS_INV_IT_LC_WK_A	REGULAR_INVENTORY_UNITS	CLEAR_IND = 'Y'
drtynslsclrc	W_PDS_SLS_IT_LC_WK_A	NET_SALES_CLR_COST	
drtynslsclrr	W_PDS_SLS_IT_LC_WK_A	NET_SALES_CLR_RETAIL	
drtynslsclru	W_PDS_SLS_IT_LC_WK_A	NET_SALES_CLR_UNITS	
drtynslsproc	W_PDS_SLS_IT_LC_WK_A	NET_SALES_PRO_COST	
drtynslspror	W_PDS_SLS_IT_LC_WK_A	NET_SALES_PRO_RETAIL	
drtynslsprou	W_PDS_SLS_IT_LC_WK_A	NET_SALES_PRO_UNITS	
drtynslsregc	W_PDS_SLS_IT_LC_WK_A	NET_SALES_REG_COST	
drtynslsregr	W_PDS_SLS_IT_LC_WK_A	NET_SALES_REG_RETAIL	
drtynslsregu	W_PDS_SLS_IT_LC_WK_A	NET_SALES_REG_UNITS	
drtyrtnclrc	W_PDS_SLS_IT_LC_WK_A	RETURNS_CLR_COST	
drtyrtnclrr	W_PDS_SLS_IT_LC_WK_A	RETURNS_CLR_RETAIL	
drtyrtnclru	W_PDS_SLS_IT_LC_WK_A	RETURNS_CLR_UNITS	
drtyrtnproc	W_PDS_SLS_IT_LC_WK_A	RETURNS_PRO_COST	
drtyrtnpror	W_PDS_SLS_IT_LC_WK_A	RETURNS_PRO_RETAIL	
drtyrtnprou	W_PDS_SLS_IT_LC_WK_A	RETURNS_PRO_UNITS	
drtyrtnregc	W_PDS_SLS_IT_LC_WK_A	RETURNS_REG_COST	
drtyrtnregr	W_PDS_SLS_IT_LC_WK_A	RETURNS_REG_RETAIL	
drtyrtnregu	W_PDS_SLS_IT_LC_WK_A	RETURNS_REG_UNITS	
drtyooc	W_PDS_PO_ONORD_IT_LC_WK_A	ON_ORDER_COST	
drtyoor	W_PDS_PO_ONORD_IT_LC_WK_A	ON_ORDER_RETAIL	
drtyoou	W_PDS_PO_ONORD_IT_LC_WK_A	ON_ORDER_UNITS	
drtyporcptc	W_PDS_INVRC_IT_LC_WK_A	PO_RECEIPT_COST	
drtyporcptr	W_PDS_INVRC_IT_LC_WK_A	PO_RECEIPT_RETAIL	
drtyporcptu	W_PDS_INVRC_IT_LC_WK_A	PO_RECEIPT_UNITS	
drtytraninbc	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_COST	TSF_TYPE = 'B'
drtytraninbr	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_RETAIL	TSF_TYPE = 'B'
drtytraninbu	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_UNITS	TSF_TYPE = 'B'
drtytraninic	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_COST	TSF_TYPE = 'I'
drtytraninir	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_RETAIL	TSF_TYPE = 'I'
drtytraniniu	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_UNITS	TSF_TYPE = 'I'
drtytraninr	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_RETAIL	TSF_TYPE = 'N'

Fact Name	External Interface Table	External Mapped Column	External Mapping Condition
drtytraninc	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_COST	TSF_TYPE = 'N'
drtytraninu	W_PDS_INVTSF_IT_LC_WK_A	TSF_IN_UNITS	TSF_TYPE = 'N'
drtytranoutbc	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_COST	TSF_TYPE = 'B'
drtytranoutbr	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_RETAIL	TSF_TYPE = 'B'
drtytranoutbu	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_UNITS	TSF_TYPE = 'B'
drtytranoutic	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_COST	TSF_TYPE = 'I'
drtytranoutir	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_RETAIL	TSF_TYPE = 'I'
drtytranoutiu	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_UNITS	TSF_TYPE = 'I'
drtytranoutr	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_RETAIL	TSF_TYPE = 'N'
drtytranoutu	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_UNITS	TSF_TYPE = 'N'
drtytranoutc	W_PDS_INVTSF_IT_LC_WK_A	TSF_OUT_COST	TSF_TYPE = 'N'
drtyicmkur	W_PDS_MKDN_IT_LC_WK_A	INTERCOMPANY_MARKUP	
drtyicmkdr	W_PDS_MKDN_IT_LC_WK_A	INTERCOMPANY_MARKDOWN	
drtywfslsu	W_PDS_SLSWF_IT_LC_WK_A	FRANCHISE_SALES_UNITS	
drtywfslsc	W_PDS_SLSWF_IT_LC_WK_A	FRANCHISE_SALES_COST	
drtywfslsr	W_PDS_SLSWF_IT_LC_WK_A	FRANCHISE_SALES_RETAIL	
drtywfrtnu	W_PDS_SLSWF_IT_LC_WK_A	FRANCHISE_RETURNS_UNITS	
drtywfrtnr	W_PDS_SLSWF_IT_LC_WK_A	FRANCHISE_RETURNS_COST	
drtywfrtnc	W_PDS_SLSWF_IT_LC_WK_A	FRANCHISE_RETURNS_RETAIL	
addvlocopnd	VW_LOC_DATA	STORE_OPEN_DATE	
addvlocendd	VW_LOC_DATA	STORE_CLOSE_DATE	
addvlocrefd	VW_LOC_DATA	REMODEL_DATE	
drdvprdatt	W_PDS_PRODUCT_ATTR_D (VW_PATV_DATA)	PROD_ATTR_VALUE	
drdvppatvt	VW_PATV_DATA	PATV_VALUE	
drtyudab	VW_UDS_DATA	True for PROD_ATTR	
drtypclsst	VW_PROD_DATA	CLASS_DISPLAY_ID	
drtypsclst	VW_PROD_DATA	SUBCLASS_DISPLAY_ID	
drdvskuimgt	VW_PROD_DATA	PRODUCT_IMAGE_NAME	
drdvskuimgl	VW_PROD_DATA	PRODUCT_IMAGE_ADDR	

Note: For Location specific data, the same W_PDS_ORGANIZATION_D hierarchy table used for the location hierarchy is used. The view VW_LOC_DATA is defined in PDS to point to the same set of data and used as data importer interface. Similarly, VW_PROD_DATA is defined against W_PDS_PRODUCT_D to load any required data as measures such as Image details, RMF CS Unique Class, and Sub-Class Ids.

VW_PATR_DATA is defined against W_PDS_PRODUCT_ATTR_D for UDA_TYPE in ‘LV’ and also gets attribute values for the DIFF*, SUPPLIER, AND BRAND_NAME from the W_PDS_PRODUCT_D table at the item level. It also concatenates the product attribute values with product attribute names using ‘_’ and uses ‘supp’ and ‘brnd’ as product attribute names for Supplier and Brand.

The VW_PATV_DATA internal view defined against the Product Attribute Hierarchy table contains product attribute values without concatenation of product attribute names and it uses similar tables as in VW_PATR_HIER. The VW_UDA_DATA is defined against W_PDS_UDA_D to only contain distinct UDA to uniquely identify the UDAs defined in RMFCS.

Note: If the customer wants to use position filtering to filter a few products only in APCS in a multi-app environment, they can use the extensibility in GA, to set the Position Filter Measure property for product in Hierarchies to the measure DRDVAPFltSkuB and then can use any flex field for the product in W_PDS_PRODUCT_D to mark those items as 'Y' and interface that data to APCS_DRDVAPFltSkuB by changing the interface.cfg mapping for VW_PROD_DATA using extensibility guidelines.

Import Forecasts from AI Foundation

Forecasts can be generated from AI Foundation (AIF) and imported to APCS using RAP integration. AI Foundation can generate different levels of forecasts as needed by different levels of plans. It generates both Pre-Season forecasts (using the Auto-ES Forecast method) and In-Season Forecasts (using the Bayesian Forecast Method). AI Foundation directly gets the actuals through RAP integration. Job flows in AI Foundation need to be scheduled to generate the forecast and import the same to APCS. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

In order to get forecasts from AI Foundation, during implementation, some initial setups need to be done in the AI Foundation platform. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

The following table shows the interface table column details from AI Foundation in RDX used for the interface.

Interface Name: RSE_FCST_DEMAND_EXP

Table Column	Data Type	Comments
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.
CAL_HIER_LEVEL	Varchar2(30)	The calendar level data is for Fiscal Year, Fiscal Quarter, Fiscal Period, Fiscal Week, and Fiscal Day.
LOC_HIER_LEVEL	Varchar2(30)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, LOCATION, and CHANNEL.
PROD_HIER_LEVEL	Varchar2(30)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, STYLE_COLOR, and ITEM.
FCST_DATE_FROM	Date	The start date which the forecast is for.
LOC_EXT_KEY	Varchar2(80)	The external id of the location. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as AREA~123).

Table Column	Data Type	Comments
PROD_EXT_KEY	Varchar2(80)	The external id of the product hierarchy. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as CLS~123~456~789).
CUSTSEG_EXT_KEY	Varchar2(80)	The external id of the customer segment. It will be NULL if not applicable.
FCST_TYPE	Varchar2(20)	The type of forecast. PI, NPI (PI=Plan Influenced, PI = Non Plan Influenced)
REG_SLS_QTY	Number(38,20)	Regular Sales Units
REG_SLS_AMT	Number(38,20)	Regular Sales Amount
PR_SLS_QTY	Number(38,20)	Promo Sales Units
PR_SLS_AMT	Number(38,20)	Promo Sales Amount
CLR_SLS_QTY	Number(38,20)	Clearance Sales Units
CLR_SLS_AMT	Number(38,20)	Clearance Sales Amount
REG_PR_SLS_QTY	Number(38,20)	Regular and Promo Sales Units
REG_PR_SLS_AMT	Number(38,20)	Regular and Promo Sales Amount
SLS_QTY	Number(38,20)	Total Sales Units
SLS_AMT	Number(38,20)	Total Sales Amount
RET_QTY	Number(38,20)	Return Units
RET_AMT	Number(38,20)	Return Amount

The same Interface table contains the forecast data for different levels of plans differentiated by _LEVEL columns within the interface. The single interface run pulls data for different levels of forecasts which are pre-configured. Customers using non-template versions, if using different levels of plans, can use the supported levels in AI Foundation to generate forecasts. The following sections provide the default levels of forecasts exported for the APCS template version and their mappings.

Item Level Forecasts Mapping

The following table shows the mapping for pre-season and in-season Item Level Forecasts.

Table Column	Mapping for Pre-Season (MPP)	Mapping for In-Season (MPI)
CAL_HIER_LEVEL	Fiscal Week	Fiscal Week
LOC_HIER_LEVEL	LOCATION	LOCATION
PROD_HIER_LEVEL	ITEM	ITEM
FCST_DATE_FROM	WEEK	WEEK
LOC_EXT_KEY	STOR	STOR
PROD_EXT_KEY	SKU	SKU
CUSTSEG_EXT_KEY	NULL	NULL
FCST_TYPE	NPI	PI
REG_PR_SLS_QTY	FCTYFCPMU	FCTYFCIMU
REG_PR_SLS_AMT	FCTYFCPMR	FCTYFCIMR

Sub-Class Level Forecasts Mapping

The following table shows the mapping for pre-season Sub-Class Level Forecasts.

Table Column	Mapping for Pre-Season (MTP)
CAL_HIER_LEVEL	Fiscal Week
LOC_HIER_LEVEL	LOCATION
PROD_HIER_LEVEL	SBC
FCST_DATE_FROM	WEEK
LOC_EXT_KEY	STOR
PROD_EXT_KEY	SCLS
CUSTSEG_EXT_KEY	NULL
FCST_TYPE	NPI
REG_PR_SLS_QTY	FCDVSlS1U
REG_PR_SLS_AMT	FCDVSLS1R

Import DT Parameters from AI Foundation

APCS uses Demand Transference (DT) from AI Foundation to suggest and optimize the assortments. DT is based on Item attributes, Attribute Weights, Functional Fit for Attributes, Assortment Elasticity, and Rate of Sale of Items. Item Attributes and Rate of Sale of item are available from RI interfaces. Other DT parameters such as Attribute Weights, Functional Fit for Attributes, and Assortment Elasticity are interfaced from AI Foundation through RAP integration. The APCS template version gets the DT parameters from RAP at the Sub-Class/Channel level, so it needs to be defined in AI Foundation at that level.

In order to get DT parameters from AI Foundation, during implementation, some initial setups need to be done in the AI Foundation platform. AI Foundation needs a customer segment to be defined for DT interfaces. AI Foundation can use multiple customer segments but AP is not using Customer Segment, so the customer should only set one dummy customer segment in AI Foundation for this DT integration. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

The following table shows the interface table column details from AI Foundation in RDX used for the interface and the corresponding mapping of columns in APCS. Only mapped columns are used by the APCS template version. If RAP integration is enabled and if Enable RSE DT Integration is set to true, then this interface will run as part of the weekly batch.

Interface Name: RSE_ASSORT_ELASTICITY_EXP

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.	

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
LOC_HIER_LEVEL	Varchar2(30)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, LOCATION, and CHANNEL_COUNTRY.	CHANNEL_COUNTRY
PROD_HIER_LEVEL	Varchar2(30)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, STYLE_COLOR, and ITEM.	SBC
LOC_EXT_KEY	Varchar2(80)	The external id of the location. It will use the integration ids as provided to RI.	CHNC
PROD_EXT_KEY	Varchar2(80)	The external id of the product hierarchy. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as CLS~123~456~789).	SCLS
CUSTSEG_EXT_KEY	Varchar2(80)	The external id of the customer segment.	
ASSORT_ELASTICITY	Number(38,20)	The assortment elasticity which DT has calculated.	DRTYASSRTELASV
EFFECTIVE_DT_FROM	Date	The date when this data was activated.	

Interface Name: RSE_ASSORT_ATTR_WGT_EXP

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.	
LOC_HIER_LEVEL	Varchar2(30)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, LOCATION, and CHANNEL_COUNTRY.	CHANNEL_COUNTRY
PROD_HIER_LEVEL	Varchar2(30)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, STYLE_COLOR, and SKU.	SBC
LOC_EXT_KEY	Varchar2(80)	The external id of the location. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as AREA~123).	CHNL
PROD_EXT_KEY	Varchar2(80)	The external id of the product hierarchy. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as CLS~123~456~789).	SCLS
CUSTSEG_EXT_KEY	Varchar2(80)	The external id of the customer segment. It will be NULL if not applicable.	

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
PROD_ATTR_GRP_EXT_KEY	Varchar2(80)	The external ID for the product attribute.	PATT
ATTR_WGT	Number(38,20)	The attribute weight.	DRTYATTRWGTV
FUNC_ATTR_FLG	Varchar2(1)	Y/N Flag indicating this is a functional attribute. A functional attribute is one which fits a specific purpose and cannot be substituted by other products with other values for this attribute.	DRTYFUNCFITB

Import Location Clusters from AI Foundation

Location Clusters can be defined in AI Foundation and can be interfaced to APCS. The APCS template version also supports defining Location Clusters within the application. AI Foundation allows defining clusters at different levels across the product hierarchy but the APCS template version allows interfacing clusters defined at the department level. Location Clusters are defined for a date range and those date ranges can be defined as Assortment Periods in APCS and the same can be exported to AI Foundation to define the Location Clusters. For more details, see Export Assortment Periods for Location Clustering to AI Foundation.

In order to get location clusters from AI Foundation, during implementation, some initial setups need to be done in the AI Foundation platform. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

The following table shows the interface table column details from AI Foundation in RDX used for the interface and the corresponding mapping of columns in APCS. Only mapped columns are used by the APCS template version. If RAP integration is enabled and if Enable RSE Cluster Integration is set to true, then this interface will run as part of the weekly batch.

Interface Name: RSE_LOC_CLUSTER_EXP

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.	
LOC_HIER_LEVEL	Varchar2(30)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, and LOCATION.	LOCATION
PROD_HIER_LEVEL	Varchar2(30)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, STYLE_COLOR, and ITEM.	DEPT

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
LOC_EXT_KEY	Varchar2(80)	The external id of the location. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as AREA~123).	STOR
PROD_EXT_KEY	Varchar2(80)	The external id of the product hierarchy. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as CLS~123~456~789).	DEPT
EFF_START_DT	Date	The starting date which the cluster is effective.	WEEK, DRDVSRTD
EFF_END_DT	Date	The ending date for which the cluster is effective.	DRDVENDD
CLUSTER_ID	Number(10)	The identifier for the cluster.	DRDVSTRCLUST
CLUSTER_LABEL	Varchar2(50)	A descriptive name/label for the cluster.	DRDVSTRCLUSL

Import Size Profiles from AI Foundation

Size Profiles can be defined in AI Foundation and can be interfaced to APCS. The APCS template version uses Size Profiles defined from AIF or can use the size profiles set at the Admin level within the application. The APCS template version imports size profiles from AIF at the sub-class level. It is used to define Buy Quantity by Size and Receipts by Sizes while planning them.

In order to get Size Profiles from AI Foundation, during implementation, some initial setups need to be done in the AI Foundation platform. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

The following table shows the interface table column details from AI Foundation in RDX used for the interface and the corresponding mapping of columns in APCS. Only mapped columns are used by the APCS template version. For complete details of all available columns, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*. If RAP integration is enabled and Enable RSE Size Profile Integration is set to true, then this interface will run as part of the weekly batch. If integration is enabled, it will use the same interface table data to also load the Size Hierarchy file (SIZH) using the internal view VW_SIZH_HIER and then also load the Size Profile.

Interface Name: RSE_SIZE_PROFILE_EXP

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.	
LOC_LEVEL_NAME	Varchar2(255)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, and LOCATION.	LOCATION

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
PROD_LEVEL_NAME	Varchar2(255)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, and STYLE_COLOR.	SBC
LOC_EXT_KEY	Varchar2(80)	The external id of the location. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as AREA~123).	STOR
PROD_EXT_KEY	Varchar2(80)	The external id of the product hierarchy. It will use the integration ids as provided to RI (preferably the RMS id, and not an integration id such as CLS~123~456~789).	SCLS
SIZE_EXT_ID	Number(10)	Size External Identifier	SIZD
SIZE_PCT_UNITS	Number(10,5)	Size Profile Percentage	DRDVSIZE
USED_BY_AIP	Varchar2(1)	Flag to indicate that record is for AIP Y	PRFP
SIZE_LABEL	Varchar2(255)	Size Label	SIZD_LABEL
SIZE_GROUP_ID	Varchar2(255)	Size Group Identifier	SRNG
SIZE_GROUP_LABEL	Varchar2(255)	Size Group Label	SRNG_LABEL

Note:

For Size Hierarchy (sizh), the internal view (VW_SIZE_HIER) is defined against the AIF RSE_SIZE_PROFILE_EXP table to get the unique sizes (SIZE_EXT_ID, SIZE_LABEL) and size ranges (SIZE_GROUP_ID and SIZE_GROUP_LABEL) which are marked to be used AP (USED_BY_AIP flag as 'Y').

SIZE_EXT_LABEL, also available as part of this view, can be used instead of SIZE_LABEL to get extended labels for sizes from AIF.

Export Assortment Periods for Location Clustering to AI Foundation

In the APCS template version, the customer can define the Assortment periods at pre-defined product levels (DEPT). Assortment Periods are date ranges to plan the assortments; it can vary for different product levels. The customer can export the defined Assortment Periods by enabling the Boolean measure Export Period for Clustering at the Assortment Period level.

In order to import and use this data from AI Foundation, during implementation, some initial setups need to be done in the AI Foundation platform. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

The following table shows the interface tables and column details from APCS in RDX used for the interface and the corresponding mapping of columns in APCS. Only

mapped columns are used by the APCS template version. If RAP integration is enabled and if Enable RSE Cluster Integration is set to true, then this interface will run as part of the weekly batch.

Interface Name: AP_ASSORT_PERIOD_EXP

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.	
PROD_LEVEL	Varchar2(80)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, STYLE_COLOR, and SKU.	DEPT
PROD_KEY	Varchar2(80)	Product Identifier	DEPT
LOC_LEVEL	Varchar2(80)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, and LOCATION.	COMPANY
LOC_KEY	Varchar2(80)	Location Identifier	COMP
ASSORT_PERIOD_KEY	Varchar2(80)	Assortment Period Key	BPER
ASSORT_PERIOD_START_DATE	Date	Assortment Period Start Date	BCDVSRTD
ASSORT_PERIOD_END_DATE	Date	Assortment Period End Date	BCDVENDD
EXT_NAME	Varchar2(80)	External Cluster Name	BCDVCLUSTERT
CLUSTER_DESCR	Varchar2(255)	Cluster Description	BCDVPRDL

Export Active Assortments to AI Foundation

In the APCS template version, the customer can plan active assortments for an assortment period and those details can be exported to AI Foundation at the Item/Store level.

In order to import and use this data from AI Foundation, during implementation, some initial setups need to be done in the AI Foundation platform. For more details, see the *Oracle Retail Analytics and Planning Integration Implementation Guide*.

The following table shows the interface tables and column details from APCS in RDX used for the interface and the corresponding mapping of columns in APCS. Only mapped columns are used by the APCS template version. If RAP integration is enabled, then this interface will run as part of the weekly batch.

Interface Name: AP_ACTIVE_ASSORT_EXP

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
RUN_ID	Number(10)	The export Run ID as obtained from the RAP_INTF_UTIL.	
PROD_KEY	Varchar2(80)	Product Identifier	SKU
LOC_KEY	Varchar2(80)	Location Identifier	STOR

Table Column	Data Type	Comments	Dimension / Measure / Value Mapping
PROD_LEVEL	Varchar2(80)	The product hierarchy level the data is for CMP, DIV, GRP, DEPT, CLS, SBC, STYLE, STYLE_COLOR, and SKU.	ITEM
LOC_LEVEL	Varchar2(80)	The location hierarchy level data is for COMPANY, CHAIN, AREA, REGION, DISTRICT, and LOCATION.	LOCATION
ACTIVE_START_DATE	Date	Active Assortment Start Date	BPDBSRTD
ACTIVE_END_DATE	Date	Active Assortment End Date	BPDBENDD

Export Assortment Plans to Retail Insights

Approved plans from APCS can be exported to Retail Insights within RAP integration. The APCS template version allows creating and exporting plans at the week/store/item level for both OP and CP versions. Plans defined at the week/store/style-color level are evenly spread to the week/store/item level before this exports in the batch process. Plans for different versions can be exported to Retail Insights on a weekly basis. It exports all the approved plans for the un-elapsed periods. With the AP non-template version, customers can create a different level of plans and they can also configure various metrics. The interface staging table in Retail Sights contains more metrics columns and various flex columns. The customer can update and configure the interface.cfg mappings to export additional columns that can be used by Retail Insights.

For more details about the list of columns available in the Retail Insights Interface Staging table if the customer plans to use extensibility or use the non-template version to send additional data, see the *Oracle Retail Insights Implementation Guide*. This guide contains only the mapped columns for the APCS template version.

This plan export is for exporting both the OP and CP versions of approved Assortment Plans. VERSION_NUM 0 is used to export the OP version and 1 is used to export the CP version. The following table only shows the mapping for OP versions. The CP versions of mapping remain the same; the Version Number used is 1.

Interface Name: AP_PLAN1_EXP

Staging Table Column	Description	Dimension/Measure Mapping
PROD_KEY	Product Dimension	sku
LOC_KEY	Location Dimension	stor
CLND_KEY	Calendar Dimension	week
PROD_DH_ATTR	Attribute Dimension for RI	-1
SUPPLIER_NUM	Supplier Dimension for RI	-1
CAL_DATE	Last Day of Week	IEOPPWKED
VERSION_NUM	Version Number	0
PROD_LEVEL	Product Level	ITEM

Staging Table Column	Description	Dimension/Measure Mapping
LOC_LEVEL	Location Level	LOCATION
SLS_QTY	Op Sales Reg+Promo U	IEOPSLS1U
SLS_RTL_AMT	Op Sales Reg+Promo R	IEOPSLS1R
SLS_COST_AMT	Op Sales Reg+Promo C	IEOPSLS1C
EOH_COST_AMT	Op EOP C	IEOPEOPC
EOH_RTL_AMT	Op EOP R	IEOPEOPR
EOH_QTY	Op EOP U	IEOPEOPU
INVRC_COST_AMT	Op Receipts C	IEOPRCPTC
INVRC_RTL_AMT	Op Receipts R	IEOPRCPTR
INVRC_QTY	Op Receipts U	IEOPRCPTU

Implementation Steps with RAP Integration

If RAP integration is enabled in the environment (that is, if the customer is going to get data from RMFCS using RDX integration), follow these steps for implementation. The steps assume that RPAS, RASL, UI, and RDX are already deployed:

1. Run the Batch Process in RAP in Retail Insights (RI) to load the required initial data into the RDX staging tables.
2. Upload any application-specific hierarchy files and data files that are not coming from RDX into Object Storage.
3. Once the AP Cloud Service environment is provisioned, use the bootstrap Build Application task to build the application and use the batch task as set_rdx to just set the Enable RDX Boolean before the initial batch, or run post_hier to enable the RDX Boolean and load/import only the hierarchy files, or run postbuild_rdx to enable the RDX and to load/import initial hierarchy and data files and also run the initial batch. Batch step post_hier can also be run from OAT, to enable the RDX and load the available hierarchy files after building the domain. Use postbuild only if planning to load and use only the GA data set.
4. Schedule the regular weekly flow in the RI, AIF, and Planning applications in JOS/POM to interface the initial data into the application to get data from both RDX and Object Storage.

> **Note:**
>
> At least post_hier or postbuild_rdx should be run once with at least the calendar hierarchy file before trying to run the weekly batch using JOS/POM.

A

Appendix: Integration with MFP Cloud Service

Assortment Planning Cloud Service can be integrated with MFP Cloud Service to interface the Plan data. This appendix provides details about the integration.

APCS needs MFPCS data to use Merchandise Financial Planning data as a target to align with final assortments. If both MFPCS and APCS are co-deployed in the same environment and both are using the GA version, then all integrated MFPCS measures will be readily available in APCS. When plans are approved in MFPCS, it will be readily available in APCS. No separate batch or integration is needed to interface the MFP Plan data.

Integration with MFP Cloud Service

The following table shows the list of measures shared between MFPCS and APCS pre-configured applications. All the ML measures are at the Week/Sub-Class/Store level and the LP measures are at the Week/Department/Store level. ML measures are directly used in the AP planning process templates but LP (Location Plan) measures are used only in the batch process which spreads Warehouse data to the location level if the customer uses the option to spread using Location Plan from MFP instead of last year actuals.

MFP Cloud Service to Assortment Planning Cloud Service

MFP Measure	MFPCS Label	APCS Label	File Name
mlcpeopc	ML Cp EOP C	MFP Loaded CP EOP C	mfp_mpcp.csv.ovr
mlcpeopr	ML Cp EOP R	MFP Loaded CP EOP R	mfp_mpcp.csv.ovr
mlcpeopu	ML Cp EOP U	MFP Loaded CP EOP U	mfp_mpcp.csv.ovr
mlcprcptc	ML Cp Receipts C	MFP Loaded CP Receipts C	mfp_mpcp.csv.ovr
mlcprcptr	ML Cp Receipts R	MFP Loaded CP Receipts R	mfp_mpcp.csv.ovr
mlcprcptu	ML Cp Receipts U	MFP Loaded CP Receipts U	mfp_mpcp.csv.ovr
mlcprtn1r	ML Cp Returns Reg+Promo R	MFP Loaded CP Returns Reg+Promo R	mfp_mpcp.csv.ovr
mlcprtn1u	ML Cp Returns Reg+Promo U	MFP Loaded CP Returns Reg+Promo U	mfp_mpcp.csv.ovr
mlcprtn2r	ML Cp Returns Clr R	MFP Loaded CP Returns Clear R	mfp_mpcp.csv.ovr
mlcprtn2u	ML Cp Returns Clr U	MFP Loaded CP Returns Clear U	mfp_mpcp.csv.ovr
mlcpsls1r	ML Cp Sales Reg+Promo R	MFP Loaded CP Sales Reg+Promo R	mfp_mpcp.csv.ovr
mlcpsls1u	ML Cp Sales Reg+Promo U	MFP Loaded CP Sales Reg+Promo U	mfp_mpcp.csv.ovr
mlcpsls2r	ML Cp Sales Clr R	MFP Loaded CP Sales Clr R	mfp_mpcp.csv.ovr
mlcpsls2u	ML Cp Sales Clr U	MFP Loaded CP Sales Clr U	mfp_mpcp.csv.ovr

MFP Measure	MFPCS Label	APCS Label	File Name
mlcpslsc	ML Cp Sales C	MFP Loaded CP Sales Reg+Promo C	mfp_mpcp.csv.ovr
mlwpooadjc	ML Wp On Order Adj C	MFP Loaded WP On Order Adj C	mfp_otb.csv.rpl
mlwpooadjr	ML Wp On Order Adj R	MFP Loaded WP On Order Adj R	mfp_otb.csv.rpl
mlwpooadju	ML Wp On Order Adj U	MFP Loaded WP On Order Adj U	mfp_otb.csv.rpl
mlwpotbc	ML Wp OTB C	MFP Loaded WP OTB C	mfp_otb.csv.rpl
mlwpotbr	ML Wp OTB R	MFP Loaded WP OTB R	mfp_otb.csv.rpl
mlwpotbu	ML Wp OTB U	MFP Loaded WP OTB U	mfp_otb.csv.rpl
lplaeopc	Ap EOP C	LP AP EOP C	mfp_lpap.csv.ovr
lplapeopr	Ap EOP R	LP AP EOP R	mfp_lpap.csv.ovr
lplapeopu	Ap EOP U	LP AP EOP U	mfp_lpap.csv.ovr
lplaprcptc	Ap Receipts C	LP AP Receipts C	mfp_lpap.csv.ovr
lplaprcptr	Ap Receipts R	LP AP Receipts R	mfp_lpap.csv.ovr
lplaprcptu	Ap Receipts U	LP AP Receipts U	mfp_lpap.csv.ovr
lplaprtnr	Ap Returns R	LP AP Returns R	mfp_lpap.csv.ovr
lplaprtnu	Ap Returns U	LP AP Returns U	mfp_lpap.csv.ovr
lplaslsc	Ap Sales C	LP AP Sales C	mfp_lpap.csv.ovr
lplaslsr	Ap Sales R	LP AP Sales R	mfp_lpap.csv.ovr
lplaslsu	Ap Sales U	LP AP Sales U	mfp_lpap.csv.ovr

If the customer is using their own customized MFP, they can provide the MFP Plan data in csv file format with the measures grouped by File Name as defined in the above table interface with headers as the measure names.

B

Appendix: RMFCS Integration

Assortment Planning Cloud Service supports integration with Oracle Retail Merchandising Foundation Cloud Service (RMFCS). If a retailer has RMFCS as the source system for inventory and transactional data, they can readily integrate to get foundation hierarchy data, inventory, and transactional data from RMF Cloud Service. This appendix provides details on the list of hierarchies and data files that are integrated with Oracle Retail Merchandising Foundation Cloud Service.

Integration Assumptions

Following is a list of requirements/assumptions for integration with RMFCS:

- Warehouses are sent to APCS as locations. RMFCS Warehouses holds inventory and on-order data. RMFCS Virtual Warehouses will be present within the Channel.
- RMFCS sends hierarchy and data files on a weekly basis.
- RMFCS sends the data at the item level, so APCS needs to be implemented with the lowest data load intersection as item.
- RMFCS sends all data based on only one primary currency. It also sends currency conversion rates for different currencies. That can be used within APCS to plan and view data on different local currencies.

Integrated Hierarchy and Data Files

RMFCS sends the following sets of hierarchy and data files. All RMFCS files are copied to the common cloud service share location from where APCS can pick up the files for further processing. Interface files need to be transformed to format the files into the APCS required format to filter columns not used and split and create different group of files as needed by APCS. The APCS Configured Batch Process will transform the files to the APCS required formats, rename the final files as needed by APCS, and copy them to the domain input location for subsequent hierarchy or measure load processing. For each of the hierarchy and data files, RMFCS also sends a trigger file with the same name as the original file with the extension .complete. The Assortment Planning Cloud Service batch process waits for the trigger files to start processing the corresponding data or hierarchy file. After processing, it deletes those trigger files.

Hierarchy Files

The following hierarchy files are provided by RMFCS and need to be transformed to the APCS format before loading them into IP Cloud Service. For each hierarchy file, RMFCS only sends the base and alternate dimensions of the hierarchies and not the user alternate rollups.

Calendar Hierarchy - rms_clnd.csv.dat

RMFCS sends the calendar hierarchy file without calendar labels. Calendar labels are added in the weekly batch process by calling the APCS custom script (ra_custom) with the parameter rms_clnd_label, which adds the calendar labels. If a customer wants to use

different calendar labels, they can load their preferred calendar labels using generic hierarchy position translations enabled for the calendar hierarchy.

RMFCS sends the calendar hierarchy file with dates for five years. Based on the current date, two year historical dates, current year dates, and two future year dates.

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Field Description	APCS Mapping
Day	The date from which the 4-5-4 data was derived, in YYYYMMDD format.	day
Week	The end of week date for the day, in YYYYMMDD format.	week
Month	The 4-5-4 month of the year, valid values 1 to 12.	mnth
Quarter	The 4-5-4 quarter of the year, valid values 1 to 4.	qtr
Half	The 4-5-4 half of the year, valid values 1 or 2.	half
Year	The 4-5-4 year.	year
week_of_year	The 4-5-4 week of the year, valid values 1 to 53.	woyr
day_of_week	The day number within the week, valid values 1 to 7.	

Product Hierarchy - rms_prod.csv.dat

RMFCS exports all sellable and inventoried items.

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process. Some measure data files are also extracted and loaded from this RMFCS product hierarchy file.

Field Name	Functional Name	APCS Mapping
ITEM	Item ID	sku
ITEM_DESC	Item Description	sku_label
ITEM_PARENT_DIFF	Parent/Diff ID	skup
ITEM_PARENT_DIFF_DESC	Parent/Diff Description	skup_label
ITEM_PARENT	Parent ID	skug
ITEM_PARENT_DESC	Parent Description	skug_label
SUBCLASS_ID	Subclass ID	scls
SUB_NAME	Subclass Name	scls_label
CLASS_ID	Class ID	clss
CLASS_NAME	Class Name	clss_label
DEPT	Department	dept
DEPT_NAME	Department Name	dept_label
GROUP_NO	Group	pgrp
GROUP_NAME	Group Name	pgrp_label
DIVISION	Division	dvsn

Field Name	Functional Name	APCS Mapping
DIV_NAME	Division Name	dvsn_label
COMPANY	Company	cmpp
CO_NAME	Company Name	cmpp_label
FORECAST_IND	Forecastable Item flag	
CLASS_DISPLAY_ID	Class Display ID	drtyclsst
SUBCLASS_DISPLAY_ID	Subclass Display ID	drtysclst
BRAND_NAME	Brand ID	drdvprdatt
BRAND_DESCRIPTION	Brand Description	
SUPPLIER	Supplier Site	drdvprdatt
SUP_NAME	Supplier Name	
DIFF_TYPE1	Diff Type1	
DIFF_ID1	Diff 1	drdvprdatt
DIFF_TYPE2	Diff Type2	
DIFF_ID2	Diff 2	drdvprdatt
DIFF_TYPE3	Diff Type3	
DIFF_ID3	Diff 3	drdvprdatt
DIFF_TYPE4	Diff Type4	
DIFF_ID4	Diff 4	drdvprdatt

Location Hierarchy - rms_loc.csv.dat

All Virtual Warehouses in RMFCS will be exported as unique locations.

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Functional Name	APCS Mapping
LOCATION	Location ID	stor, strc
LOC_NAME	Location Name	stor_label,strc_label
DISTRICT	District ID	dstr
DISTRICT_NAME	District Name	dstr_label
REGION	Region ID	regn
REGION_NAME	Region Name	regn_label
AREA	Area ID	chnl
AREA_NAME	Area Name	chnl_label
CHAIN	Chain ID	chan
CHAIN_NAME	Chain Name	chan_label
COMPANY	Company ID	comp
CO_NAME	Company Name	comp_label
COMPANY_CURRENCY	Primary Currency	
LOC_TYPE	Location Type	loct

Field Name	Functional Name	APCS Mapping
LOC_TYPE_NAME	Location Type Description	loct_label
PHYSICAL_WH	Physical WH ID	phwh
PHYSICAL_WH_NAME	Physical WH Name	phwh_label
CHANNEL_ID	WH Channel ID	
CHANNEL_NAME	WH Channel Name	
STORE_CLASS	Store Class	
STORE_CLASS_DESCRIPTION	Store Class Description	
STORE_FORMAT	Store Format	
STORE_FORMAT_NAME	Store Format Name	
PLANNING_CHANNEL_ID	Planning Channel ID	chnc
PLANNING_CHANNEL_NAME	Planning Channel Name	chnc_label
PLANNING_COUNTRY_ID	Planning Country ID	ccty
PLANNING_COUNTRY_NAME	Planning Country Name	ccty_label

Currency Hierarchy - rms_curr.csv.ovr

The Currency conversion rate data file from RMFCS will be transformed to be loaded as a Currency Hierarchy file to load the unique currency codes.

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Functional Name	APCS Mapping
EFFECTIVE_DATE	Effective Date	
FROM_CURRENCY_CODE	From Currency	
TO_CURRENCY_CODE	To Currency	curc, curc_label
EXCHANGE_TYPE	Exchange Rate Type	
EXCHANGE_RATE	Exchange Rate	

Product Attribute Hierarchy - rms_uda.csv.dat, rms_brand.csv.dat, rms_vendor.csv.dat, rms_diff.csv.dat

UDA, Suppliers (Vendor), Brand, and Diffs in RMFCS are treated as product attributes in Item Planning. RMFCS provides them as individual files in the same format; those will be merged and loaded as one product attribute file. UDA_VALUE (Product Attribute Values) are not unique across UDA_ID in RMFCS. While transforming and loading as a product attribute, UDA_ID position will be concatenated with UDA_VALUE using an underscore ("_").

The following table contains the list of column contents from the file.

Field Name	Functional Name	APCS Mapping
UDA_ID	UDA	patt
UDA_DESC	UDA Description	patt_label
UDA_VALUE	UDA Value	patv

Field Name	Functional Name	APCS Mapping
UDA_VALUE_DESC	UDA Value Description	patv_labael

Data Files

The following data files are provided by RMFCS and need to be transformed to the APCS format before loading them into IP Cloud Service. All RMFCS files are transformed first and loaded together using the single Load Task Load - RMS Interface Data.

Inventory Data File - rms_inv.csv.ovr

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Functional Name	APCS Mapping
EOW_DATE	End of week date	week
ITEM	Item ID	sku
LOCATION	Location ID	stor
LOC_TYPE	Location Type	
CLEAR_IND	Clearance flag	
REGULAR_INVENTORY_UNITS	Inventory Units	drtyeopu
REGULAR_INVENTORY_COST	Inventory Cost	drtyeopc
REGULAR_INVENTORY_RETAIL	Inventory Retail	drtyeopr
UNIT_COST	Unit Cost	drtyslsprcc
AV_COST	Average Cost	
UNIT_RETAIL	Unit Retail	drtyslsprcr

Transaction Data File - rms_tran.csv.ovr

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process. RMFCS sends Net Sales in all regular, promotion, and clearance buckets. But APCS GA also needs Gross Sales. After loading, additional batch calculations, which run as part of RMFCS batch transforms, calculate those required measures. Also, RMFCS sends multiple type transaction details specific to receipts; batch calculations will calculate the effective receipts data.

Field Name	Functional Name	APCS Mapping
EOW_DATE	End of Week Date	week
ITEM	Item ID	sku
LOCATION	Location ID	stor
LOC_TYPE	Location Type	
CLEAR_IND	Clearance flag	
STANDARD_UOM	Standard UOM	
CURRENCY_CODE	Currency Code	
NET_SALES_REG_UNITS	Net Sales Units - Reg	drtynslsregu

Field Name	Functional Name	APCS Mapping
NET_SALES_REG_COST	Net Sales Cost - Reg	drtynslsregc
NET_SALES_REG_RETAIL	Net Sales Retail - Reg	drtynslsregr
NET_SALES_PROMO_UNITS	Net Sales Units - Promo	drtynslsprou
NET_SALES_PROMO_COST	Net Sales Cost - Promo	drtynslsproc
NET_SALES_PROMO_RETAIL	Net Sales Retail - Promo	drtynslspror
NET_SALES_CLEAR_UNITS	Net Sales Units - Clear	drtynslsclru
NET_SALES_CLEAR_COST	Net Sales Cost - Clear	drtynslsclrc
NET_SALES_CLEAR_RETAIL	Net Sales Retail - Clear	drtynslsclrr
NET_SALES_REG_RETAIL_VAT_EXCL	Net Sales (VAT Exclusive) Retail - Reg	
NET_SALES_PROMO_RETAIL_VAT_EXCL	Net Sales (VAT Exclusive) Retail - Promo	
NET_SALES_CLR_RETAIL_VAT_EXCL	Net Sales (VAT Exclusive) Retail - Clear	
RETURNS_REG_UNITS	Returns Units - Reg	drtyrtnregu
RETURNS_REG_COST	Returns Cost - Reg	drtyrtnregc
RETURNS_REG_RETAIL	Returns Retail - Reg	drtyrtnregr
RETURNS_PROMO_UNITS	Returns Units - Promo	drtyrtnprou
RETURNS_PROMO_COST	Returns Cost - Promo	drtyrtnproc
RETURNS_PROMO_RETAIL	Returns Retail - Promo	drtyrtnpror
RETURNS_CLEAR_UNITS	Returns Units - Clear	drtyrtnclru
RETURNS_CLEAR_COST	Returns Cost - Clear	drtyrtnclrc
RETURNS_CLEAR_RETAIL	Returns Retail - Clear	drtyrtnclrr
REG_MARKDOWN_RETAIL	Regular Markdown	
PROMO_MARKDOWN_RETAIL_REG	Promotion Markdown (Regular)	
PROMO_MARKDOWN_RETAIL_CLEAR	Promotion Markdown (Clearance)	
CLEAR_MARKDOWN_RETAIL	Clearance Markdown	
WF_MARKDOWN_RETAIL	Franchise Markdown	
WF_MARKUP_RETAIL	Franchise Markup	
SHRINK_UNITS	Shrink Units	
SHRINK_COST	Shrink Cost	
SHRINK_RETAIL	Shrink Retail	
DEAL_INCOME_SALES	Deal Income Sales Based	
PO_RECEIPT_UNITS	PO Receipt Units	drtyporcptu
PO_RECEIPT_COST	PO Receipt Cost	drtyporcptc
PO_RECEIPT_RETAIL	PO Receipt Retail	drtyporcptr
NON_SHRINK_ADJ_UNITS	Non-Shrink Adjustments Units	
NON_SHRINK_ADJ_COST	Non-Shrink Adjustments Cost	
NON_SHRINK_ADJ_RETAIL	Non-Shrink Adjustments Retail	

Field Name	Functional Name	APCS Mapping
DEAL_INCOME_PURCHASES	Deal Income Purchases Based	
MARKUP	Markup	drtymkupr
MARKDOWN_CANCEL	Markdown Cancel	drtymkdcanr
INTERCOMPANY_MARKUP	Intercompany Markup	drtyicmkur
INTERCOMPANY_MARKDOWN	Intercompany Markdown	drtyicmkdr
RTV_UNITS	RTV Units	
RTV_COST	RTV Cost	
RTV_RETAIL	RTV Retail	
TSF_IN_UNITS	Transfer/Allocation Inbound Units	drtytraninu
TSF_IN_COST	Transfer/Allocation Inbound Cost	drtytraninc
TSF_IN_RETAIL	Transfer/Allocation Inbound Retail	drtytraninr
TSF_IN_UNITS_BOOK	Book Transfer/Allocation Inbound Units	drtytraninbu
TSF_IN_COST_BOOK	Book Transfer/Allocation Inbound Cost	drtytraninbc
TSF_IN_RETAIL_BOOK	Book Transfer/Allocation Inbound Retail	drtytraninbr
TSF_OUT_UNITS	Transfer/Allocation Outbound Units	drtytranoutu
TSF_OUT_COST	Transfer/Allocation Outbound Cost	drtytranoutc
TSF_OUT_RETAIL	Transfer/Allocation Outbound Retail	drtytranoutr
TSF_OUT_UNITS_BOOK	Book Transfer/Allocation Outbound Units	drtytranoutbu
TSF_OUT_COST_BOOK	Book Transfer/Allocation Outbound Cost	drtytranoutbc
TSF_OUT_RETAIL_BOOK	Book Transfer/Allocation Outbound Retail	drtytranoutbr
RECLASS_IN_UNITS	Reclass In Units	
RECLASS_IN_COST	Reclass In Cost	
RECLASS_IN_RETAIL	Reclass In Retail	
RECLASS_OUT_UNITS	Reclass Out Units	
RECLASS_OUT_COST	Reclass Out Cost	
RECLASS_OUT_RETAIL	Reclass Out Retail	
TSF_IN_UNITS_ICT	Intercompany Transfer/Allocation Inbound Units	drtytraniniu
TSF_IN_COST_ICT	Intercompany Transfer/Allocation Inbound Cost	drtytraninic
TSF_IN_RETAIL_ICT	Intercompany Transfer/Allocation Inbound Retail	drtytraninir
TSF_OUT_UNITS_ICT	Intercompany Transfer/Allocation Outbound Units	drtytranoutiu
TSF_OUT_COST_ICT	Intercompany Transfer/Allocation Outbound Cost	drtytranoutir
TSF_OUT_RETAIL_ICT	Intercompany Transfer/Allocation Outbound Retail	drtytranoutic
INTERCOMPANY_MARGIN	Intercompany Margin	
TSF_RECEIPT_UNITS	Transfer/Allocation Receipt Units	
TSF_RECEIPT_COST	Transfer/Allocation Receipt Cost	

Field Name	Functional Name	APCS Mapping
TSF_RECEIPT_RETAIL	Transfer/Allocation Receipt Retail	
RTV_RESTOCK_FEE	RTV Restocking Fee	
FRANCHISE_SALES_UNITS	Franchise Sales Units	drtywfslsu
FRANCHISE_SALES_COST	Franchise Sales Cost	drtywfslsc
FRANCHISE_SALES_RETAIL	Franchise Sales Retail	drtywfslsr
FRANCHISE_RETURNS_UNITS	Franchise Returns Units	drtywfrtnu
FRANCHISE_RETURNS_COST	Franchise Returns Cost	drtywfrtnc
FRANCHISE_RETURNS_RETAIL	Franchise Returns Retail	drtywfrtnr
FRANCHISE_RESTOCK_FEE	Franchise Restocking Fee	

On Order Data File - rms_oo.csv.ovr

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Functional Name	APCS Mapping
EOW_DATE	End of week date	week
ITEM	Item ID	sku
LOCATION	Location ID	stor
LOC_TYPE	Location Type	
CLEAR_IND	Clearance flag	
ON_ORDER_UNITS	On Order Units	drtyoou
ON_ORDER_COST	On Order Cost	drtyooc
ON_ORDER_RETAIL	On Order Retail	drtyoor

Currency Conversion Rate File - rms_curr.csv.ovr

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Functional Name	APCS Mapping
EFFECTIVE_DATE	Effective Date	day
FROM_CURRENCY_CODE	From Currency	
TO_CURRENCY_CODE	To Currency	curc
EXCHANGE_TYPE	Exchange Rate Type	
EXCHANGE_RATE	Exchange Rate	drtylcratex

Location File - rms_store.csv.ovr

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process.

Field Name	Functional Name	APCS Mapping
STORE	Store ID	stor
STORE_NAME	Store Name	
DISTRICT	District ID	
STORE_CLOSE_DATE	Store Close Date	addvlocendd
STORE_OPEN_DATE	Store Open Date	
REMODEL_DATE	Remodel Date	addvlocrefd
STORE_CLASS	Store Class	
STORE_CLASS_DESCRIPTION	Store Class Description	
STORE_FORMAT	Store Format	
STORE_FORMAT_NAME	Store Format Name	
CURRENCY	Currency Code	
STORE_TYPE	Store Type	addvwfpoct
STOCKHOLDING_IND	Stockholding	

Item Attribute Mapping File - prdattt.csv.ovr

The following table contains the list of column contents from the file. Only required columns for APCS are filtered by the transform process. This file only contains Product Attribute mappings for UDA attributes. Item Attribute mappings for Brand, Supplier, and Diffs are derived and loaded from the RMFCS Product file.

Field Name	Functional Name	APCS Mapping
ITEM	Item ID	sku
UDA_ID	UDA ID	patt
UDA_DESC	UDA Description	
UDA_VALUE	UDA Value	drdvprdatt
UDA_VALUE_DESC	UDA Value Description	
FORECAST_IND	Forecastable Item	

Warehouse Data Aggregation

In RMFCS, warehouses hold inventory, receipts, and on-order data that are not mapped to any specific channels. In APCS, plans are only created at the Store/Location Cluster level. In order to get the full picture of inventory for locations within a channel, each warehouse location's inventory within a channel needs to be spread to locations based on their actuals. That is handled by weekly batch aggregations to spread the warehouse data within a channel to all the locations.

The weekly batch aggregation process will spread the Warehouse data at the channel level to the store level using the Warehouse to Store Spread Method which has options to use Location Plan, Location Sales, Actuals, and Actual Sales with the default option using Location Plan interfaced from MFPCS. The customer can change this option in the Batch Setup view in the Admin workbooks.

New Placeholder Item Integration

New Placeholder Items (Style/Color or Styles) created in APCS can be integrated with RMFCS. It involves multiple levels of process which are detailed below:

- First, the RMFCS Server should be connected with APCS during deployment, after mapping the dimensions (style/color, style) to item in RMFCS. New Placeholder items created in APCS will directly use the reserved item numbers from RMFCS. For more details, refer to the RPASCE-RMS Integration for Item Management - Slow Item Build section in the *Oracle Retail Predictive Application Server Cloud Edition Administration Guide*. The customer should ensure that *max_item_resv_qty* and *max_item_expiry_days* are set on the RMFCS DB side to allow reserving of items from the RMFCS side.
- The customer can then create new placeholder items within APCS and assign attributes to them. RMFCS does not contain style/color as a separate level, so an attribute should be selected as the RMFCS Color attribute in the Product Setup view in Planning Administration. All new style/colors created should be assigned with a unique color attribute. The customer should then select the style/color ready for Export to RMFCS. The process only allows for creating new placeholder items and assigning initial attributes. Any further updates/maintenance of the item or RMFCS specific item attributes should be managed in the RMFCS side.
- The customer can run the Export to RMF CS custom menu which exports all items marked for Export to RMFCS in real time to the RMFCS Item Induction process. It exports new item details, and its assigned attributes (UDA).
- In batch, the customer needs to run Rename Style/Color that renames the style/color positions created as style concatenated with a color attribute as it will be identified in the RMFCS product hierarchy file. This process is scheduled to run as part of weekly batch before the hierarchy load process.
- Once RMFCS approves the newly interfaced items, the same items will be coming in the product hierarchy file as regular items. Since it will use the same position names for Item and Style/Color, the standard hierarchy load process in the APCS side will then formalize those informal placeholder Item and Style/Color.

Batch Tasks Specific to RMFCS Integration

The following sets of Online Administration Tools tasks are pre-configured in Configured Batch Tasks to support RMFCS integration. For more details about running the tasks, see the *Oracle Retail Assortment Planning Cloud Service Administration Guide*.

- Run Batch Task Group > Weekly Batch

  If Enable RMF CS Integration is set to true, the existing Weekly Batch task will run additional tasks to process the RMFCS files. It also runs the transformation of RMFCS Warehouse data as part of the weekly batch.
- Run Batch Task Group > Transform RMF CS Files

  This task only transforms all hierarchy and data files to the APCS required format, adds the header information, and copies the files to the input location so that any subsequent data or hierarchy load process can process the same data.
- Run Batch Task Group > Set RMF CS Dimension Mapping

RMFCS can be linked with the APCS domain in real time to get the reserved item numbers for new placeholder items created. In order to do that, connections between the RMFCS servers and RPAS server needs to be set up during installation time. For more details about those setups, see the *Oracle Retail Predictive Application Server Cloud Edition Administration Guide*. Once those are set, running this task will map the sku (Item) and skup (style/color) dimensions to Item in RMFCS. Once this mapping is done, any new place holder sku or skup created will use the reserved RMFCS item numbers directly from RMFCS.

- Run Batch Task Group > Rename RMF CS Style/Color

  This task renames the Style/Color DPM positions which are exported to RMFCS, with Style concatenated with the Color Attribute Id, as it comes in the Product Hierarchy Interface.

C

Appendix: Standard Exports

Assortment Planning Cloud Service provides multiple standard exports that retailers can use to integrate with other systems that need AP Plan data. It also exports new Placeholder items and new Product Attributes that can be used to interface them to their source systems.

For more information related to extracted measures, the approval process, and the administration process related to exports, see the *Oracle Retail Assortment Planning Cloud Service User Guide*. For more details about executing the export using Online Administration Tools using the Admin task Export Data, see the *Oracle Retail Assortment Planning Cloud Service Administration Guide*.

Assortment Planning Cloud Service Exports

This section lists the different Assortment Planning Cloud Service standard exports and the list of measures exported. All exported files will be in CSV format. A few exports are available for export only if the particular solution is enabled.

In APCS, Assortment Plans are approved at the Style-Color/Week/Store level, but before export, those approved plans are spread down to the Item/Week/Store level for Item Plan level exports.

Item Plan - Pre-Season

Export Set Name: Item Plan - Pre-Season

Export Set: ip_ipop

Exported File: ipop_strc_plan.txt

Export Criteria: All approved Pre-Season Item Plans for all un-elapsed time periods.

Table C-1 Item Plan - Pre-Season Export Measures

Measure	Measure Label	Data Type	Aggregation Type
week	Week		
sku	Item		
stor	Store		
IEOPSls1U	Op Sales Reg + Promo U	real	Total
IEOPSls1R	Op Sales Reg + Promo R	real	Total
IEOPSls1C	Op Sales Reg + Promo C	real	Total
IEOPEOPU	Op EOP U	real	Total
IEOPEOPR	Op EOP R	real	Total
IEOPEOPC	Op EOP C	real	Total
IEOPRcptU	Op Receipts U	real	Total

Table C-1 (Cont.) Item Plan - Pre-Season Export Measures

Measure	Measure Label	Data Type	Aggregation Type
IEOPRcptR	Op Receipts R	real	Total
IEOPRcptC	Op Receipts C	real	Total

Item Plan - In-Season

Export Set Name: Item Plan - In-Season

Export Set: ip_ipcp

Exported File: ipcp_strc_plan.txt

Export Criteria: All approved In-Season Item Plans for all un-elapsed time periods.

Table C-2 Item Plan - In-Season Export Measures

Measure	Measure Label	Data Type	Aggregation Type
week	Week		
skup	Item		
stor	Store		
IECPSls1U	Cp Sales Reg + Promo U	real	Total
IECPSls1R	Cp Sales Reg + Promo R	real	Total
IECPSls1C	Cp Sales Reg + Promo C	real	Total
IECPEOPU	Cp EOP U	real	Total
IECPEOPR	Cp EOP R	real	Total
IECPEOPC	Cp EOP C	real	Total
IECPRcptU	Cp Receipts U	real	Total
IECPRcptR	Cp Receipts R	real	Total
IECPRcptC	Cp Receipts C	real	Total

Item Plan - OTB

Export Set Name: Item Plan - OTB

Export Set: ip_otb

Exported File: ipwp_otb.txt

Export Criteria: Planned OTB for all un-elapsed time periods.

Table C-3 Item Plan - OTB Export Measures

Measure	Measure Label	Data Type	Aggregation Type
week	Week		
skup	Style/Color		
stor	Store		
ISWPOTBU	WP OTB U	real	Total
ISWPOTBR	WP OTB R	real	Total
ISWPOTBC	WP OTB C	real	Total

New Place-holder Items

Export Set Name: New Place-holder Items

Export Set: ph_item

Exported File: as_new_item.txt

Export Criteria: All newly created placeholder items which are not formalized. It also exports additional product rollup information for that item.

Table C-4 New Place-holder Items Export Measures

Measure	Measure Label	Data Type	Aggregation Type
sku	Item		
DRDVPSKUL	Place-holder Item Label	String	
DRDVPSKUPT	Style-Color Identifier	String	
DRDVPSKUPL	Style-Color Label	String	
DRDVPSKUGT	Style-Color Label	String	
DRDVPSKUGL	Style Identifier	String	
DRDVPSCLST	Style Label	String	
DRDVPSCLSL	Subclass Identifier	String	
DRDVPCLSST	Subclass Label	String	
DRDVPCLSSL	Class Identifier	String	
DRDVPDEPTT	Class Label	String	
DRDVPDEPTL	Department Identifier	String	
DRDVPDEPTL	Department Label	String	

New Product Attributes

Export Set Name: New Product Attributes

Export Set: ph_patt

Exported File: as_new_patt.txt

Export Criteria: All Item - Product Attribute association for newly created product attributes in Assortment Planning.

Table C-5 New Product Attributes Export Measures

Measure	Measure Label	Data Type	Aggregation Type
sku	Item		
patt	Product Attribute	String	
ADDVPrdAttT	Product Attribute Value	String	

Base Unit Price

Export Set Name: Base Unit Price

Export Set: as_sprc

Exported File: as_base_price.txt

Export Criteria: Base Unit Retail and Cost set for all Item/Store.

Table C-6 Base Unit Price Export Measures

Measure	Measure Label	Data Type	Aggregation Type
sku	Item		
stor	Store		
ADDVSlsPrcR	Retail Price	real	Average
ADDVSlsPrcC	Cost	real	Average

Export to Allocation

Export Set Name: Export to Allocation

Export Set: af_alloc

Exported File: ap_to_alloc.txt

Export Criteria: Approved Receipt Quantities for Allocation

Table C-7 Export to Allocation Measures

Measure	Measure Label	Data Type	Aggregation Type
week	Week		
skup	Style/Color		
stor	Store		
APDVWeek2DayTx	EOW Date	real	Total
APDVStyleLblTx	Style Label	real	Total
APDVColorTx	Color Code	real	Total
APCPRcptU	Cp Receipts U	real	Total

> **Note:**
>
> For Allocation, the standard exported file is further formatted as plitpt.0l (for Style/Color) files by the batch process by calling the custom function ap_exp_alloc.

D

Appendix: Application Specific Custom Functions, Procedures, and Expressions

Assortment Planning Cloud Service uses the RPAS Enterprise Edition Batch Framework to set up the batch process. Refer to the *Oracle Retail Assortment Planning Cloud Service Implementation Guide* for more details about understanding batch_control files and its uses in the batch process. Any application-specific custom functions are packaged in the RPAS Applications Standard Library (RASL) libraries and those are available to use in the batch_exec_list.txt control file with the service name as **ra_custom** to trigger those functions and parameter as the function name. The following list of predefined custom functions are available RASL libraries installed as part of RPAS that are used by Item Planning.

This appendix also contains details about special procedures/expressions used in the APCS configuration.

Custom Function: rms_clnd_label

RMFCS interfaces the calendar hierarchy file (rms_clnd.csv.dat), but does not include any calendar labels. This function formats the file, filters the required columns for the APCS GA calendar hierarchy, and adds the calendar labels. There is no additional parameter needed for this function call, but this requires the RMFCS calendar file to be present in the cloud share location with the name rms_clnd.csv.dat. After the call, this function creates the calendar hierarchy file clnd.csv.dat in the domain input directory for the subsequent hierarchy load process.

Example:

batch_rms_xform | ra_custom | rms_clnd_label

Custom Function: ap_set_datr

The Platform UI feature allows product attributes to be assigned during item creation if the attributes are defined as dimension attributes. This application-level function registers all the loaded product attributes as dimension attributes. It needs to be called each time a new set of product attributes is loaded.

Example:

batch_datr | ra_custom | ap_set_datr

Custom Function: ap_load_strc

This custom function allows transforming the AI Foundation Location Cluster file into the required location cluster file format. It also loads the location cluster file.

Example:

batch_strc | ra_custom | ap_load_strc

Custom Function: ap_exp_rms_item

This custom function is used to export new items to RMFCS in XML format. It merges and formats the exported files into XML format and uploads to Object Storage.

Example:

exp_rms | ra_custom | ap_exp_rms_item

Custom Function: ap_set_rms_dim

This custom function is used to link the dimension in the domain to the RMFCS item dimension. It sets the domain property, rms_im_levels, with dimension mappings passed as a parameter. If the domain is linked with RMFCS and dimensions are mapped with the example call shown below, then any new DPM positions created for those dimensions will use the reserved numbers from RMFCS for that mapped item. More than one dimension can be mapped using following mapping "rpasDimName:rmsDimName,...".

Example:

set_rms_sib | ra_custom | ap_set_rms_dim~sku:ITEM,skup:ITEM

Custom Function: ap_exp_alloc

This custom function formats the Standard Exports to the Allocation's system into the required format, reads the exported file ap_to_alloc.txt and formats, and converts the file as p0itpt.01 (Style only items) or p1itpt.01 (for Style/Color) and uploads to Object Storage.

Example:

exp_alloc | ra_custom | ap_exp_alloc

Custom Function: ap_sib_color

This custom function is used to rename the Style/Color positions using the Style:<color attribute name> format for newly created DPM positions as needed by RMFCS integration so that if actual positions are integrated, it will be formalized.

It expects the ap_sib_color.dat file containing the DPM style/color position, style position, and color attribute in the temp directory. It creates the required product hierarchy renaming the file from it and calls eebatch_rename_positions.ksh to rename those positions.

No additional inputs required to call this function.

Example:

batch_sib | ra_custom | ap_sib_color

Custom Function: load_udd

This custom function can be used to load any User Defined Dimension (UDD). It internally calls the loadUdd utility to load the custom User Defined Dimension from a file. It can be used to customize the batch process if the customer wants to load any User Defined Dimension as part of their batch process. This function expects only one parameter and it should be the UDD name. It also expects the customer to upload a CSV file to their Object Storage location as <UDD>.csv.dat. If the file is not present, it

will ignore with a warning in the logs. Only one UDD can be loaded from a single file. The UDD file should only have three columns. First column should be the base parent dimension position for UDD, second column should be the UDD position name, and third column should be the UDD position label.

Example:

load_udd | ra_custom | load_udd~stdb

Custom Function: set_stdb

This custom function can be used to automatically set the STDB User Defined Dimension (UDD) dynamically based on RPAS_TODAY with values STD/BTA. It expects the STDB User Defined Dimension to be present in the calendar hierarchy and it should be a rollup of the week dimension. STD is set as the value for elapsed periods and BTA as the value for unelapsed periods based on RPAS_TODAY used in the domain. It internally generates the UDD file and calls the custom function load_udd to load the same.

This function takes two optional parameters to override the labels that need to be used for STD and BTA positions. If not provided, it will use the position name as the labels. If the optional parameters are provided, it will use the custom labels provided to this function call as the parameter. If there are spaces in the labels, then provide the parameters within double quotation marks.

Example:

load_hier | ra_custom | set_stdb

or

load_hier | ra_custom | set_stdb~"Elapsed Weeks"~"Unelapsed Weeks"

Custom Function: key_lookup

This custom function can be used to look up key mapping columns and replace it with key position names in an input file. Key mappings for the key positions can be defined as a single dimensional measure and that can be passed as an input. For matched mapping columns, this function replaces it with the key position values. Typical use case for this function is in an integration if the source application is using different key position names and key positions than used in the RPASCE application and if the keys need to be replaced before processing within the RPASCE solution, this function can be used, if the mappings are available in a measure. Example use case in GA is, AI Foundation uses different sub-class id and class-id than used by the Planning solutions when integrated with RMFCS. If an interface from AI Foundation uses sub-class id in the format of AI Foundation, it can be transformed to use the sub-class id keys used in Planning, if the customer can load the mapping with the sub-class id in Planning to AI Foundation in a single dimensional mapping measure.

This function takes three parameters, Input File Name, Position of the column where the dimension needs to be changed, and the Mapping measure name.

Example: a sample file from AI Foundation (for example, test.csv.ovr) with sub-class id as the first column:

SBC~100~1000~1001,1,232

SBC~100~1000~1001,2,252

SBC~200~1000~1001,1,343

SBC~200~1000~1001,2,533

If the mapping measure adhdpsclst is defined within the application at the sub-class level with following mapping information for the subclass:

1001001, SBC~100~1000~1001

2001001, SBC~200~1000~1001

For the above example, if the function is called in the batch control file:

batch_test | ra_custom | key_lookup~test.csv.ovr~1~adhdpsclst

After transformation, the same file should be:

1001001,1,232

1001001,2,252

2001001,1,343

2001001,1,533

This function looks up the input files in the following directories in the order of Object Storage and RPAS_CUST_ROOT/temp. If files are found, it will do the transformation, otherwise, it will abort for file not found. If the mapping measure is empty, it will not do any transformation. If the mappings are not found, it will also not do the transform for those keys. The transformed file will be present in <DOMAIN>/input and RPAS_CUST_ROOT/temp with the same name, so that the customer can use the same transformed file in batch for subsequent file processing (either to load that file or do further data transformations).

Special Expressions and Procedures Used in APCS Configuration

This section contains details about the special expressions that are not standard RPAS expressions/procedures but are used in the APCS configuration.

AttributesPickListExpr - Attribute Picklist Expression

This procedure is used to determine a picklist of values from a base string measure along the positions of the specified hierarchy. It also takes in a Filter Boolean as input to control which intersections are to be used to pick data for the picklist.

Syntax:

OUTMEAS <- AttributesPickListExpr(FILTERMEAS, INPUTMEAS, HIER, OUTFORMAT)

Example:

ADHDLikePoCT <- AttributesPickListExpr(ADHDLikePoCB,ADHDLikePoCL, "LOC", "0")

Expression Parameters:

The following table shows the parameter details for the expression.

Parameter Name	Parameter Type	Data Type	Description
OUTMEAS	Output	String	Output Picklist Measure which will be a string value.
INPUTMEAS	Input	String	Input String Measure containing the picklist values stored along a hierarchy.
FILTERMEAS	Input	Boolean	Boolean measure with the same intersection of OUTMEAS. It should be set to true for the intersections picklist that needs to be created.
HIER	Input	String	Hierarchy Name specified as a String constant along which the picklist values need to be used. This hierarchy should be present in INPUTMEAS.
OUTFORMAT	Input	String	Output format with valid values "0" or "1". Using "0" will create a picklist using position names as picklist internal names and labels as attribute values. "1" will create a picklist using position values as name and labels for the picklist.

BopEopCalcExpr - Inventory Roll Calculation Expression

All planning applications typically need to create some batch rules to do inventory roll operations for all their actualized positions. Inventory roll calculations involve Beginning Inventory (BOP) and Ending Inventory (EOP) metrics, where BOP is calculated as lag of EOP (previous periods EOP) and EOP will be calculated from that period BOP plus additional metrics (Receipts, Sales, and so on). RPAS supports the use of the lag function to calculate BOP and creation of a separate expression to calculate EOP.

They normally run as cyclic rules in a batch rule group. Problem is this dual cyclic expression is highly performance intensive if it needs to roll inventory at the item/sku/store level. This is an optimized procedure to do inventory rollup calculations as a single expression instead of using them as cyclic expressions. This procedure can be used only in batch mode and should not be used in workbook calculation cycle rule groups.

Syntax:

EOP:<MEAS>,BOP:<MEAS> <- BopEopCalcExpr(TODAYINDEX:<MEAS>, INCEOP:<MEAS>, ACTEOP:<MEAS>, TSMASK:<MEAS>, INITIALBOP:<MEAS>)

Example:

EOP:ISWPEOPU,BOP:ISWPBOPU <- BopEopCalcExpr(TODAYINDEX:DRDVUnElapI, INCEOP:ISDVEOPU, ACTEOP:DRDVEOPU, TSMASK:DRDVEOPB, INITIALBOP:DRDVBOSU)

The above expression is equivalent to running the following two expressions in batch as a cyclic rule group:

ISWPBOPU = if (DRDVEOPB, if (current == first, DRDVBOSU, lag(ISWPEOPU)), ignore)

ISWPEOPU = if (DRDVEOPB, if (DRDVUnElapB, ISWPBOPU + ISDVEOPU, DRTYEOPU), ignore)

Expression Parameters:

The following table shows the parameter details for the expression.

Parameter Name	Parameter Type	Data Type	Description
EOP	Output	Real	Calculated Ending Inventory value measure.
BOP	Output	Real	Beginning Inventory value measure, same intersection as EOPMEAS.
ACTEOP	Input	Real	Actual Ending Inventory for Elapsed Periods. Same Intersection as BOP and EOP.
INCEOP	Input	Real	Incremental Inventory value for UnElapsed Periods. Same intersection as BOP and EOP.
TODAYINDEX	Input	Integer	Index of Current RPAS Today period. Periods above this will be copied with actual EOP, periods after this will be calculated with BOP and Incremental EOP. Base Intersection higher or equal to BOP/EOP but should not have the calendar dimension.
TSMASK	Input	Boolean	Intersections to which inventory roll calculations need to be performed. Same intersection as BOP and EOP but without calendar dimension.
INITIALBOP	Input	Real	First Period Beginning Inventory value measure. Same Intersection of BOP and EOP but without the calendar dimension.

ClusterExpr - Cluster Expression

This procedure is used to define a dynamic cluster based on input parameters.

Syntax:

POINTMEMBERSHIP: <MEAS>, POINTMEMBERSHIPSTR: <MEAS>, CENTROID: <MEAS> <- ClusterExpr(MEASURE: <MEAS>, METHOD: <STRING>, NUMCLUSTERS: <MEAS>, CLUSTERHIER: <HIER>, BYGROUPDIMS: <DIMENSION>)

Example:

POINTMEMBERSHIP: SCDVBangGrpID, POINTMEMBERSHIPSTR: SCDVBangGrpT, CENTROID: SCDVBangGrpX <- ClusterExpr(MEASURE: SCDVCmbIdxV, METHOD: "BANG", NUMCLUSTERS: SCHDMaxPrfGrpU, CLUSTERHIER: "LOC", BYGROUPDIMS: "CHNL")

Expression Parameters:

The following table shows the parameter details for the expression.

Parameter Name	Parameter Type	Data Type	Description
POINTMEMBERSHIP	Output	Integer	Output Cluster Group Id at Location level.
POINTMEMBERSHIPSTR	Output	String	Output Cluster Group String at Location level.

Parameter Name	Parameter Type	Data Type	Description
CENTROID	Output	Real	Output Centroid value at Aggregate level.
MEASURE	Input	Real	Input Score measure at Location level.
METHOD	Input	String	String constant with valid value, "BANG" for Bang Algorithm to use.
NUMCLUSTERS	Input	Integer	Number of Clusters to Create at Aggregate of Location level.
CLUSTERHIER	Input	String	String constant with hierarchy name of Location hierarchy for which cluster is created.
BYGROUPDIMS	Input	String	String constant of dimension name in Location hierarchy at Aggregate level.

CMNewItemSimilarityV2Expr - Item Similarity Expression

This procedure is a wrapper procedure used for calling AI Foundation functions which calculate the Item Similarity Scores based on Product Attributes and Product Attribute Weights. This similarity score can be internally calculated and used in Demand Transference procedures as input to determine the Demand Transference across similar items. It is also used in Assortment Improvement Procedures using Demand Transference.

Syntax:

SIM_OUT <- CMNewItemSimilarityV2Expr(SIM_IN, PROD_ATTR, ATTR_WGT, DT_MODE, FUNC_FIT, SAME_ITEM_MAP, VALID_ATTR, SIM_MODE)

Example:

BWHDSimOutVp <- CMNewItemSimilarityV2Expr(BWHDSimInV, ADDVPrdAttT, BWHDAttrWgtV, DRDVDTModeX, BWHDFuncFitB, DRDVSkupMapB, BWHDPrdAttB, DRDVSimModeV)

Expression Parameters:

The following table shows the parameter details for the expression.

Parameter Name	Parameter Type	Data Type	Description
SIM_OUT	Output	Real	Similarity score across LHS and RHS Items at Item/Cluster/RHS Item level.
SIM_IN	Input	Real	It is needed only if a different Similarity Mode other than 0 is used. For default Similarity Mode 0, this can be initialized to 0 which calculates similarity for all items.
PROD_ATTR	Input	String	Product Attribute values at Item/ Product Attribute level.
ATTR_WGT	Input	Real	Attribute Weight at Item/Attribute level.

Parameter Name	Parameter Type	Data Type	Description
DT_MODE	Input	Integer	Demand Transference Mode to calculate the similarity scores, value of 4 is for calculating Bi-Directional Similarity Score, 3 for One Directional Score.
FUNC_FIT	Input	Boolean	Functional Fit Boolean at Sub-Class/ Product Attribute level.
SAME_ITEM_MAP	Input	Boolean	Same Item mapping across LHS-RHS Items at LHS-RHS Item level.
VALID_ATTR	Input	Boolean	Product Attribute Eligibility at Sub-Class/Attribute level.
SIM_MODE	Input	Real	Similarity mode at sub-class level with default value as 0 for similarity values calculating for all items.

CMDemandSignTranV2Expr - Demand Transference Expression

This procedure is a wrapper procedure used for calling AI Foundation functions which in turn calculate the Demand Transferences and Substitutable Sales Units. It also calculates the significant similar items and substitutable sales percentage against each similar item based on the similarity scores. It internally calculates the similarity scores using the attribute inputs and uses it to calculate the Demand transference outputs.

Syntax:

DT_SIG_SKU_COUNT, DT_SIG_SUB_PCT, DT_SALES, DT_SUB_SALES <- CMDemandSignTranV2Expr(DT_MASK, SIM_OUT, ITEM_KAD, INPUT_SALES, SAME_ITEM_MAP, ASSORT_ELASTICITY, DT_CUTOFF, DT_GUARD, PROD_ATTR, ATTR_WGT, SIM_MODE, FUNC_FIT, VALID_ATTR)

Example:

BWHDDTSigSkuV, BWHDDTSigSkuVp, BWHDDTSlsU, BWHDDTSubSlsU <- CMDemandSignTranV2Expr(BWHDRunMaskB, BWH1SimOutVp, BWHDDTKADX, BWHDDTROSU, DRDVSkupMapB, BWHDAssrtElasV, DRDVDTCutOffVp, DRDVDTGuardB, ADDVPrdAttT, BWHDAttrWgtV, DRDVSimModeV, BWHDFuncFitB, BWHDPrdAttB)

Expression Parameters:

The following table shows the parameter details for the expression.

Parameter Name	Parameter Type	Data Type	Description
DT_SIG_SKU_COUNT	Output	Integer	Significant Substitutable Items count at Item/Cluster level.
DT_SIG_SUB_PCT	Output	Real	Significant Sku Substitutable Percentage at Item/Cluster/RHS Item level.
DT_SALES	Output	Real	Demand Transference ROS/Sales at Item/Cluster level (At same level of input ROS).

Parameter Name	Parameter Type	Data Type	Description
DT_SUB_SALES	Output	Real	Substitutable ROS/Sales at Item/ Cluster level.
DT_MASK	Input	Boolean	Mask at Sub-class/Cluster level for which DT needs to be calculated.
SIM_OUT	Input	Real	It is needed only if different Similarity Mode other than 0 is used. For default Similarity Mode 0, this can be initialized to 0 procedure internally calculates similarity for all items.
ITEM_KAD	Input	Integer	Item Keep/Add/Drop Status with valid values as 1 for Add, 2 for Keep, and 3 for Drop Status of Items at Item/Cluster level.
INPUT_SALES	Input	Real	Initial Forecasted/Historical Sales or Rate of Sales at Item/Cluster level.
SAME_ITEM_MAP	Input	Boolean	Same Item mapping across LHS-RHS Items at LHS-RHS Item level.
ASSORT_ELASTICITY	Input	Real	Assortment Elasticity at Sub-Class/Cluster level.
DT_CUTOFF	Input	Real	DT Cut-Off % at Sub-Class/ Cluster level.
DT_GUARD	Input	Real	DT Guard Boolean at Sub-Class/ Cluster level.
PROD_ATTR	Input	String	Product Attribute values at Item/ Product Attribute level.
ATTR_WGT	Input	Real	Attribute Weight at Item/Attribute level.
SIM_MODE	Input	Real	Similarity mode at sub-class level with default value as 0 for similarity values calculating for all items.
FUNC_FIT	Input	Boolean	Functional Fit Boolean at Sub-Class/Product Attribute level.
VALID_ATTR	Input	Boolean	Product Attribute Eligibility at Sub-Class/Attribute level.

CMAssortImprovementV2Expr - Assortment Improvement using DT Expression

This procedure is a wrapper procedure used for calling AI Foundation functions which in turn use the Demand Transferences and suggest the Improved Assortments after applying Demand Transferences based on the input assortment improvement parameters.

Syntax:

OPT_ASSORT, OPT_SALES <- CMAssortImprovementV2Expr(DT_MASK, INIT_SALES, SAME_ITEM_MAP, SIM_OUT, NEW_ITEM, INIT_ASSORT, IMPROVE_METRIC, ELIGIBLE_ITEM, IMPROVE_COUNT, MIN_TO_KEEP, OPT_MODE, ASSORT_ELASTICITY, ITEM_MANDATORY, OPT_TARGET, PROD_ATTR, ATTR_WGT, SIM_MODE, FUNC_FIT, VALID_ATTR)

Example:

BWHDDTImprAsrtB, BWHDDTImprAsrtU <-
CMAssortImprovementV2Expr(BWHDRunMaskB, BWHDDTInitAsrtU,
DRDVSkupMapB, BWH1SimOutVp, BWHDDTNewB, BWHDDTInitAsrtB,
BWHDDTMetricImprV, BWHDDTEligB, BWHDDTImprAsrtV, BWHDMinKeepVp,
BWHDDTOptModeV, BWHDAssrtElasV, BWHDDTMandB, BWHDDTOptTgtV,
ADDVPrdAttT, BWHDAttrWgtV, DRDVSimModeV, BWHDFuncFitB, BWHDPrdAttB)

Expression Parameters:

The following table shows the parameter details for the expression.

Parameter Name	Parameter Type	Data Type	Description
OPT_ASSORT	Output	Boolean	Optimized Assortment after adding/removing items based on input parameters at Item/Cluster level.
OPT_SALES	Output	Real	Optimized Sales/ROS by calculating Improved Assortment after applying Demand Transference at Item/Cluster level.
DT_MASK	Input	Boolean	Mask at Sub-class/Cluster level for which Optimized Assortment needs to be calculated.
INIT_SALES	Input	Real	Initial Forecasted or Historical Sales or ROS at Item/Cluster level for all eligible items. It needs Sales/ROS for both historical and new items.
SAME_ITEM_MAP	Input	Boolean	Same Item mapping across LHS-RHS Items at LHS-RHS Item level.
SIM_OUT	Input	Real	It is needed only if different Similarity Mode other than 0 is used. For default Similarity Mode 0, this can be initialized to 0 procedure internally calculates similarity for all items.
NEW_ITEM	Input	Boolean	New Items within the Assortment at Item/Cluster level.
INIT_ASSORT	Input	Boolean	Initial Planned Assortment before running Assortment Improvement at Item/Cluster level.
IMPROVE_METRIC	Input	Real	Metric to Improve, 1.0 for Sales Units (at same level of input ROS/ Sales). For Sales Retail or Gross Margin, multiply it with Sales Price or Price Margin at Item/Cluster level.
ELIGIBLE_ITEM	Input	Boolean	All Eligible Items that can be included into the assortment for improvement at Item/Cluster level.

Parameter Name	Parameter Type	Data Type	Description
IMPROVE_COUNT	Input	Integer	Number of Items to be added, removed, or swapped from the assortment at Sub-Class/Cluster level.
MIN_TO_KEEP	Input	Real	Minimum Percentage of Initial Assortment to Keep during Assortment Improvement at Sub-Class/Cluster level.
OPT_MODE	Input	Integer	Optimization Mode with values 1 for Add with Swap, 2 for Drop with Swap, 3 for Swap, 4 for Add, and 5 for Remove at Sub-Class/Cluster level.
ASSORT_ELASTICITY	Input	Real	Assortment Elasticity at Sub-Class/Cluster level.
ITEM_MANDATORY	Input	Boolean	Mandatory Items to Keep after running Assortment Improvement at Item/Cluster level.
OPT_TARGET	Input	Real	Target to Achieve by Swapping assortments at Sub-Class/Cluster level.
PROD_ATTR	Input	String	Product Attribute values at Item/ Product Attribute level.
ATTR_WGT	Input	Real	Attribute Weight at Item/Attribute level.
SIM_MODE	Input	Real	Similarity mode at sub-class level with default value as 0 for similarity values calculating for all items.
FUNC_FIT	Input	Boolean	Functional Fit Boolean at Sub-Class/Product Attribute level.
VALID_ATTR	Input	Boolean	Product Attribute Eligibility at Sub-Class/Attribute level.

E

Appendix: APCS Scheduling in JOS/POM

APCS provides a default job schedule with a detailed list of jobs with pre-defined dependencies with RI and AIF schedules in RAP integration. The customer can easily customize their batch process by controlling the batch control files without changing the batch schedule. Each batch job in APCS is connected to hooks in the batch control files that can be changed by customers. The APCS job flow only contains daily and weekly jobs. Jobs scheduled daily are scheduled to run every day. Jobs scheduled to run weekly are scheduled to run only on Sunday by default.

The following table shows the list of jobs in the APCS Schedule. Only jobs with Template configuration entries are pre-configured for the APCS Template version. The rest of the jobs are placeholder jobs to allow changes to the schedule of the jobs or to add more jobs by changing the batch control file. Each APCS job calls the same set names in the batch control file without _JOB in the name which in turn allows it to call the respective *hook jobs, so even if it is not linked to a batch set. The batch can be controlled without changing the job flow by adding or changing the hooks.

For more details about the implementation of JOS/POM, see the *Oracle Retail Predictive Application Server Cloud Edition Implementation Guide*. For more details about how to use JOS/POM to schedule and maintain the job flows, see the *Oracle Retail Process Orchestration and Monitoring User Guide*.

APCS Job Name	Description/Comments	Daily/Weekly	Template Configuration
AP_PRE_EXP_START_JOB	Dummy Job, to specify the start of pre-batch exports.	Daily	
AP_PRE_EXP_RDX_D_JOB	To schedule any daily plan exports to RAP integration before the calc cycle.	Daily	
AP_PRE_EXP_OBS_D_JOB	To schedule any daily plan exports to Object Storage before the calc cycle.	Daily	Exports Standard Plans, if the Export Daily Plans Boolean is enabled.
AP_PRE_EXP_RDX_W_JOB	To schedule any weekly plan exports to RAP integration before the calc cycle.	Weekly	Exports Plans to RI, if RDX is enabled.
AP_PRE_EXP_OBS_W_JOB	To schedule any weekly plan exports to Object Storage before the calc cycle.	Weekly	Exports Standard Plans Weekly, if the Export Weekly Plans Boolean is enabled. Also exports to Allocation if DRDVALLOCB is enabled.
AP_PRE_EXP_END_JOB	Dummy job to specify the end of pre-batch exports.	Daily	
AP_COM_HIER_IMP_START_JOB	Dummy job to specify the start of a common hierarchies import.	Daily	
AP_COM_HIER_IMP_RDX_D_JOB	To schedule any daily common hierarchy imports from RAP integration.	Daily	

APCS Job Name	Description/Comments	Daily/Weekly	Template Configuration
AP_COM_HIER_IMP_OBS_D_JOB	To schedule any daily common hierarchy imports from Object Storage.	Daily	
AP_COM_HIER_IMP_RDX_W_JOB	To schedule any weekly common hierarchy imports from RAP integration.	Weekly	Imports hierarchies from RI, if RDX is enabled.
AP_COM_HIER_IMP_OBS_W_JOB	To schedule any weekly common hierarchy imports from Object Storage.	Weekly	Loads hierarchy files from Object Storage, if uploaded.
AP_COM_HIER_IMP_END_JOB	Dummy Job, to specify the end of a common hierarchies import.	Daily	
AP_COM_DATA_IMP_START_JOB	Dummy job to specify the start of common data imports.	Daily	
AP_COM_DATA_IMP_RDX_D_JOB	To schedule any daily common data imports from RAP integration.	Daily	
AP_COM_DATA_IMP_OBS_D_JOB	To schedule any daily common data imports from Object Storage.	Daily	
AP_COM_DATA_IMP_RDX_W_JOB	To schedule any weekly common data imports from RAP integration.	Weekly	Imports data files from RI, if RDX is enabled.
AP_COM_DATA_IMP_OBS_W_JOB	To schedule any weekly common data imports from Object Storage.	Weekly	Loads data files from Object Storage, if uploaded.
AP_COM_DATA_IMP_END_JOB	Dummy job to specify the end of common data imports.	Daily	
AP_HIER_IMP_START_JOB	Dummy job to specify the start of application-specific hierarchies import.	Daily	
AP_HIER_IMP_RDX_D_JOB	To schedule any daily application-specific hierarchy imports from RAP integration.	Daily	
AP_HIER_IMP_OBS_D_JOB	To schedule any daily application-specific hierarchy imports from Object Storage.	Daily	
AP_HIER_IMP_RDX_W_JOB	To schedule any weekly application-specific hierarchy imports from RAP integration.	Weekly	Imports application-specific hierarchies from RI, if RDX is enabled.
AP_HIER_IMP_OBS_W_JOB	To schedule any weekly application-specify hierarchy imports from Object Storage.	Weekly	Imports any application-specific hierarchies from Object Storage.
AP_HIER_IMP_END_JOB	Dummy job to specify the end of application-specific hierarchies import.	Daily	
AP_PRE_DATA_IMP_START_JOB	Dummy job to specify the start of application-specific data imports before the calc cycle.	Daily	

APCS Job Name	Description/Comments	Daily/Weekly	Template Configuration
AP_PRE_DATA_IMP_RDX_D_JOB	To schedule any daily application-specific data imports from RAP integration before the calc cycle.	Daily	
AP_PRE_DATA_IMP_OBS_D_JOB	To schedule any daily application-specific data imports from Object Storage before the calc cycle.	Daily	
AP_PRE_DATA_IMP_RDX_W_JOB	To schedule any weekly application-specific data imports from RAP integration before the calc cycle.	Weekly	
AP_PRE_DATA_IMP_OBS_W_JOB	To schedule any weekly application-specific data imports from Object Storage before the calc cycle	Weekly	
AP_PRE_DATA_IMP_END_JOB	Dummy Job to specify the end of application-specific data imports before the calc cycle	Daily	
AP_BATCH_AGG_START_JOB	Dummy job to specify the start of the Batch Aggregation Cycle.	Daily	
AP_BATCH_AGG_D_JOB	To schedule any daily batch aggregation jobs.	Daily	Runs Daily Batch Calcs or Aggregations.
AP_BATCH_AGG_W_JOB	To schedule any weekly batch aggregation jobs.	Weekly	Runs Weekly Batch Calcs or Aggregations.
AP_BATCH_AGG_END_JOB	Dummy job to specify the end of the Batch Aggregation Cycle.	Daily	
AP_EGRS_EXP_START_JOB	Dummy job to specify the start of Egress Export jobs.	Daily	
AP_EGRS_EXP_RDX_D_JOB	Daily export of hierarchy and actuals data to RDX for Egress.	Daily	
AP_EGRS_EXP_RDX_W_JOB	Weekly export of hierarchy and actuals data to RDX for Egress.	Weekly	
AP_EGRS_EXP_END_JOB	Dummy job to specify the end of Egress Export jobs.	Daily	
AP_POST_DATA_IMP_START_JOB	Dummy job to specify the start of application-specific data imports after the calc cycle.	Daily	
AP_POST_DATA_IMP_RDX_D_JOB	To schedule any daily application-specific data imports from RAP integration after the calc cycle.	Daily	

APCS Job Name	Description/Comments	Daily/Weekly	Template Configuration
AP_POST_DATA_IMP_OBS_D_JOB	To schedule any daily application-specific data imports from Object Storage after the calc cycle.	Daily	
AP_POST_DATA_IMP_RDX_W_JOB	To schedule any weekly application-specific data imports from RAP integration after the calc cycle.	Weekly	Imports data from, enabled Interfaces from AIF, if RDX is enabled.
AP_POST_DATA_IMP_OBS_W_JOB	To schedule any weekly application-specific data imports from Object Storage after the calc cycle.	Weekly	
AP_POST_DATA_IMP_END_JOB	Dummy Job, to specify the end of application-specific data imports after the calc cycle.	Daily	
AP_POST_EXP_START_JOB	Dummy job to specify the start of post-batch exports.	Daily	
AP_POST_EXP_RDX_D_JOB	To schedule any daily data exports from RAP integration after the calc cycle.	Daily	
AP_POST_EXP_OBS_D_JOB	To schedule any daily data exports from Object Storage after the calc cycle.	Daily	
AP_POST_EXP_RDX_W_JOB	To schedule any weekly data export from RAP integration after the calc cycle.	Weekly	Exports data from, enabled Interfaces to AIF, if RDX is enabled.
AP_POST_EXP_OBS_W_JOB	To schedule any weekly data exports from Object Storage after the calc cycle.	Weekly	
AP_POST_EXP_END_JOB	Dummy job to specify the end of post-batch exports.	Daily	
AP_WB_BUILD_START_JOB	Dummy job to specify the start of workbook build/refresh jobs.	Daily	
AP_WB_BUILD_D_JOB	To schedule any workbook build/refresh steps daily.	Daily	Runs daily scheduled workbook builds and refresh.
AP_WB_BUILD_W_JOB	To schedule any workbook build/refresh steps weekly.	Weekly	Runs weekly scheduled workbook builds and refresh.
AP_WB_BUILD_END_JOB	Dummy job to specify the end of workbook build/refresh jobs.	Daily	
AP_ADHOC_XX_JOB	To schedule any adhoc jobs with the batch execution set name as a parameter. XX varies from 01 to 10. 10 Jobs available to schedule any adhoc steps with any set name as the parameter.	Adhoc	

Even if a customer is using a non-template version of the application, they should have the following entries in their batch control file (batch_exec_list.txt) with the necessary hooks connected to the customer job steps in order to execute those steps using pre-defined AP jobs from JOS/POM.

Generic Schedule Mapping in JOS/POM

AP_PRE_EXP_RDX_D | exec | *hook_AP_PRE_EXP_RDX_D

AP_PRE_EXP_OBS_D | exec | *hook_AP_PRE_EXP_OBS_D

AP_PRE_EXP_RDX_W | exec | *hook_AP_PRE_EXP_RDX_W

AP_PRE_EXP_OBS_W | exec | *hook_AP_PRE_EXP_OBS_W

AP_COM_HIER_IMP_RDX_D | exec | *hook_AP_COM_HIER_IMP_RDX_D

AP_COM_HIER_IMP_OBS_D | exec | *hook_AP_COM_HIER_IMP_OBS_D

AP_COM_HIER_IMP_RDX_W | exec | *hook_AP_COM_HIER_IMP_RDX_W

AP_COM_HIER_IMP_OBS_W | exec | *hook_AP_COM_HIER_IMP_OBS_W

AP_COM_DATA_IMP_RDX_D | exec | *hook_AP_COM_DATA_IMP_RDX_D

AP_COM_DATA_IMP_OBS_D | exec | *hook_AP_COM_DATA_IMP_OBS_D

AP_COM_DATA_IMP_RDX_W | exec | *hook_AP_COM_DATA_IMP_RDX_W

AP_COM_DATA_IMP_OBS_W | exec | *hook_AP_COM_DATA_IMP_OBS_W

AP_HIER_IMP_RDX_D | exec | *hook_AP_HIER_IMP_RDX_D

AP_HIER_IMP_OBS_D | exec | *hook_AP_HIER_IMP_OBS_D

AP_HIER_IMP_RDX_W | exec | *hook_AP_HIER_IMP_RDX_W

AP_HIER_IMP_OBS_W | exec | *hook_AP_HIER_IMP_OBS_W

AP_PRE_DATA_IMP_RDX_D | exec | *hook_AP_PRE_DATA_IMP_RDX_D

AP_PRE_DATA_IMP_OBS_D | exec | *hook_AP_PRE_DATA_IMP_OBS_D

AP_PRE_DATA_IMP_RDX_W | exec | *hook_AP_PRE_DATA_IMP_RDX_W

AP_PRE_DATA_IMP_OBS_W | exec | *hook_AP_PRE_DATA_IMP_OBS_W

AP_BATCH_AGG_D | exec | *hook_AP_BATCH_AGG_D

AP_BATCH_AGG_W | exec | *hook_AP_BATCH_AGG_W

AP_EGRS_EXP_RDX_D_JOB | exec | *hook_AP_EGRS_EXP_RDX_D

AP_EGRS_EXP_RDX_W_JOB | exec | *hook_AP_EGRS_EXP_RDX_W

AP_POST_DATA_IMP_RDX_D | exec | *hook_AP_POST_DATA_IMP_RDX_D

AP_POST_DATA_IMP_OBS_D | exec | *hook_AP_POST_DATA_IMP_OBS_D

AP_POST_DATA_IMP_RDX_W | exec | *hook_AP_POST_DATA_IMP_RDX_W

AP_POST_DATA_IMP_OBS_W | exec | *hook_AP_POST_DATA_IMP_OBS_W

AP_POST_EXP_RDX_D | exec | *hook_AP_POST_EXP_RDX_D

AP_POST_EXP_OBS_D | exec | *hook_AP_POST_EXP_OBS_D

AP_POST_EXP_RDX_W | exec | *hook_AP_POST_EXP_RDX_W

AP_POST_EXP_OBS_W | exec | *hook_AP_POST_EXP_OBS_W

AP_WB_BUILD_D | exec | *hook_AP_WB_BUILD_D

AP_WB_BUILD_W | exec | *hook_AP_WB_BUILD_W

> **Note:**
>
> AP_EGRS_EXP_* jobs are placeholder jobs needed only for Egress customers migrated to a new release to export their hierarchy and actuals data to RDX so that AI Foundation can use the data to generate forecasts. Those jobs will execute batch steps in GA only when the Egress Export Hierarchies and Egress Export Actuals Boolean flags are enabled using the Measure Analysis template. New customers in RAP can ignore these jobs if they are getting standard hierarchies and actuals from RAP.

> **Note:**
>
> In Multi-App setup, only one application should import the common shared hierarchies using the *COM_HIER_IMP * jobs. In GA apps, it is controlled by the shared Boolean measure Primary to Load Common Data. If APCS is not the first planning application deployed in RAP and if another planning application such as MFPCS or IPOCS-Demand Forecasting is already deployed, then set Primary to Load Common Data to true in only one of the applications under the Planning Administration -> Batch Setup in show/hide, to avoid trying to load the common data again for shared planning applications through the JOS/POM schedule. If not set by the user in GA apps, that Boolean will be set to true only for the first deployed GA application for which the initial post build batch is run.

External Dependencies:

In RAP Integration, APCS gets data from RI and AIF. Those dependencies are pre-defined in the schedule. The following table shows the pre-defined dependencies of the APCS jobs The customer needs to enable those dependencies based on the modules enabled on the RI and AIF side, so that the dependent job on the APCS side waits for the completion of that interface on the source side.

JobName	ExternalScheduleName	ExternalPredecessorJobName
RPASCE_HIER_IMP_START_JOB	RI	RDX_EXPORT_DIM_MILE_MARKER_JOB
RPASCE_DATA_IMP_START_JOB	RI	RDX_EXPORT_FACT_MILE_MARKER_JOB
AP_POST_DATA_IMP_START_JOB	RSP	RSE_AP_FCST_EXPORT_JOB

JobName	ExternalScheduleName	ExternalPredecessorJobName
AP_POST_DATA_IMP_START_JOB	RSP	DT_EXPORT_END_JOB
AP_POST_DATA_IMP_START_JOB	RSP	AC_CLUSTER_SET_EXPORT_END_JOB
AP_POST_DATA_IMP_START_JOB	RSP	SPO_EXPORT_END_JOB

Schedule and Mapping in JOS/POM for Multi-Instance:

Customers can provision second or multiple instances of APCS. But second or multi-instance can only be deployed as a Custom (non-template) configuration and when it is provisioned, a new schedule will also be provided for multi-instance with <APP_INSTANCE_CODE> (example: apcs2 or apcs3) as RPASCE-<APP_INSTANCE_CODE>. Multi-instance uses the standard schedule, and the customer can customize it for the steps that it needs to execute using the corresponding entries in their custom batch control files.

The customer should have the following entries in their batch control file (batch_exec_list.txt) with the necessary hooks connected to the customer job steps in order to execute those steps using pre-defined jobs from JOS/POM. Replace the <APP> with the corresponding instance <APP_INSTANCE_CODE> code value during deploy.

Schedule also as <APP>_ADHOC_XX_JOB for the adhoc schedule purpose.

Batch Control in JOS/POM for Multi-Instance:

<APP>_PRE_EXP_RDX_D | exec | *hook_PRE_EXP_RDX_D

<APP>_PRE_EXP_OBS_D | exec | *hook_PRE_EXP_OBS_D

<APP>_PRE_EXP_RDX_W | exec | *hook_PRE_EXP_RDX_W

<APP>_PRE_EXP_OBS_W | exec | *hook_PRE_EXP_OBS_W

<APP>_HIER_IMP_RDX_D | exec | *hook_HIER_IMP_RDX_D

<APP>_HIER_IMP_OBS_D | exec | *hook_HIER_IMP_OBS_D

<APP>_HIER_IMP_RDX_W | exec | *hook_HIER_IMP_RDX_W

<APP>_HIER_IMP_OBS_W | exec | *hook_HIER_IMP_OBS_W

<APP>_PRE_DATA_IMP_RDX_D | exec | *hook_PRE_DATA_IMP_RDX_D

<APP>_PRE_DATA_IMP_OBS_D | exec | *hook_PRE_DATA_IMP_OBS_D

<APP>_PRE_DATA_IMP_RDX_W | exec | *hook_PRE_DATA_IMP_RDX_W

<APP>_PRE_DATA_IMP_OBS_W | exec | *hook_PRE_DATA_IMP_OBS_W

<APP>_BATCH_AGG_D | exec | *hook_BATCH_AGG_D

<APP>_BATCH_AGG_W | exec | *hook_BATCH_AGG_W

<APP>_POST_DATA_IMP_RDX_D | exec | *hook_POST_DATA_IMP_RDX_

<APP>_POST_DATA_IMP_OBS_D | exec | *hook_POST_DATA_IMP_OBS_D

<APP>_POST_DATA_IMP_RDX_W | exec | *hook_POST_DATA_IMP_RDX_W

<APP>_POST_DATA_IMP_OBS_W | exec | *hook_POST_DATA_IMP_OBS_W

<APP>_POST_EXP_RDX_D | exec | *hook_POST_EXP_RDX_D

<APP>_POST_EXP_OBS_D | exec | *hook_POST_EXP_OBS_D

<APP>_POST_EXP_RDX_W | exec | *hook_POST_EXP_RDX_W

<APP>_POST_EXP_OBS_W | exec | *hook_POST_EXP_OBS_W

<APP>_WB_BUILD_D | exec | *hook_WB_BUILD_D

<APP>_WB_BUILD_W | exec | *hook_WB_BUILD_W

The following Interschedule Dependencies are pre-defined with the RI, AIF, and RPASCE schedules for Hierarchy and Data Imports. If any external export dependency is needed in RI/AIF for any exports from multi-instance, that needs to be added directly in the POM UI. For information about how to add Inter-Schedule Dependencies with other applications, see the *Oracle Retail Process Orchestration and Monitoring User Guide* and *Oracle Retail Process Orchestration and Monitoring Implementation Guide*.

JobName	ExternalScheduleName	ExternalPredecessorJobName
<APP>_HIER_IMP_START_JOB	RI	RDX_EXPORT_DIM_MILE_MARKER_JOB
<APP>_PRE_DATA_IMP_START_JOB	RI	RDX_EXPORT_FACT_MILE_MARKER_JOB
<APP>_POST_DATA_IMP_START_JOB	RSP	RSE_AP_FCST_EXPORT_JOB
<APP>_HIER_IMP_START_JOB	RPASCE	RPASCE_HIER_IMP_END_JOB