Oracle® Retail Regular Price Optimization

Implementation Guide

Release 13.2.2.4

April 2011

ORACLE®

Oracle Retail Regular Price Optimization Implementation Guide, Release 13.2.2.4

Primary Author: Barrett Gaines

Licensing Note: This media pack includes a Restricted Use license for Oracle Retail Predictive Application Server (RPAS) - Enterprise Engine to support Oracle® Retail Regular Price Optimization only.

Value-Added Reseller (VAR) Language

Oracle Retail VAR Applications

The following restrictions and provisions only apply to the programs referred to in this section and licensed to you. You acknowledge that the programs may contain third party software (VAR applications) licensed to Oracle. Depending upon your product and its version number, the VAR applications may include:

(i) the **MicroStrategy** Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(ii) the **Wavelink** component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.

(iii) the software component known as **Access Via™** licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(iv) the software component known as **Adobe Flex™** licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

Contents

3 Build Scripts

4 Loading and Extracting Data

5 Script Integration

6 ODI Integration

7 Configuration Considerations

8 Batch Processing

9 Internationalization

List of Tables

List of Figures

Preface

The Oracle Retail Regular Price Optimization Implementation Guide describes post-installation tasks that need to be performed in order to bring Regular Price Optimization online and ready for production use.

Audience

This Implementation Guide is intended for the Regular Price Optimization application integrators and implementation staff, as well as the retailer's IT personnel. This guide is also intended for business analysts who are looking for information about processes and interfaces to validate the support for business scenarios within Regular Price Optimization and other systems across the enterprise.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Retail Regular Price Optimization Release 13.2.2.4 documentation set:

- *Oracle Retail Regular Price Optimization Release Notes*

Customer Support

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

https://support.oracle.com

When contacting Customer Support, please provide the following:

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

Review Patch Documentation

When you install the application for the first time, you install either a base release (for example, 13.2) or a later patch release (for example, 13.2.1). If you are installing the base release, additional patch, and bundled hot fix releases, read the documentation for all releases that have occurred since the base release before you begin installation. Documentation for patch and bundled hot fix releases can contain critical information related to the base release, as well as information about code changes since the base release.

Oracle Retail Documentation on the Oracle Technology Network

Documentation is packaged with each Oracle Retail product release. Oracle Retail product documentation is also available on the following Web site:

http://www.oracle.com/technology/documentation/oracle_retail.html

(Data Model documents are not available through Oracle Technology Network. These documents are packaged with released code, or you can obtain them through My Oracle Support.)

Documentation should be available on this Web site within a month after a product release.

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.

Send Us Your Comments

Oracle Retail Regular Price Optimization Implementation Guide, Release 13.2.2.4

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

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

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

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

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

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

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

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

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

1 Introduction

Oracle Retail Regular Price Optimization (RPO) assists retail price mangers in pricing hard-line and grocery items. It is suited for long lifecycle items with infrequent price changes. It recommends permanent prices based on initial estimates of an item's total sales volume over a planning period and on price-related sales of items and related items.

RPO includes grouping in its pricing analysis because it considers cross-item elasticities; that is, RPO considers how price changes for one item may affect the sales volume of other items. Users can input objective functions and pricing constraints that define the optimization problem. Once these inputs are defined, the pricing optimizer recommends prices. Multiple scenarios can be created and evaluated side by side, and what–if analysis can be performed within the context of a pricing scenario. After analyzing the what-if results and recommended prices, the user can make a final decision to submit the recommended prices for the given set of merchandise items and locations.

For a more detailed overview of the functionality within RPO, see the *RPO User Guide for the RPAS Classic Client* or the *RPO User Guide for the RPAS Fusion Client*.

Contents of this Guide

This implementation guide addresses the following topics:

- Chapter 1, "Introduction". Overview of the RPO business workflow and skills needed for implementation.
- Chapter 2, "Implementation Considerations". Explanation of the factors to take into consideration before performing the implementation.
- Chapter 3, "Build Scripts". Information on building and patching the RPO domain.
- Chapter 4, "Loading and Extracting Data". Lists the measures that should be loaded to load functionally coherent set of data.
- Chapter 5, "Script Integration". Overview of integration and explanation of the RPO data flow and integration script.
- Chapter 6, "ODI Integration". Overview of ODI enabled integration of RPO and APC-RPO.
- Chapter 7, "Configuration Considerations". Information on the functional changes or enhancements that can be made for RPO.
- Chapter 8, "Batch Processing". Information on the batch processing tasks for RPO embedded forecasts.
- Chapter 9, "Internationalization". Translations provided for RPO.

RPO and the Oracle Retail Enterprise

Figure 1–1 shows the architecture of RPO and Oracle Retail Predictive Application Server (RPAS).

Figure 1–1 RPO and the Oracle Retail Enterprise

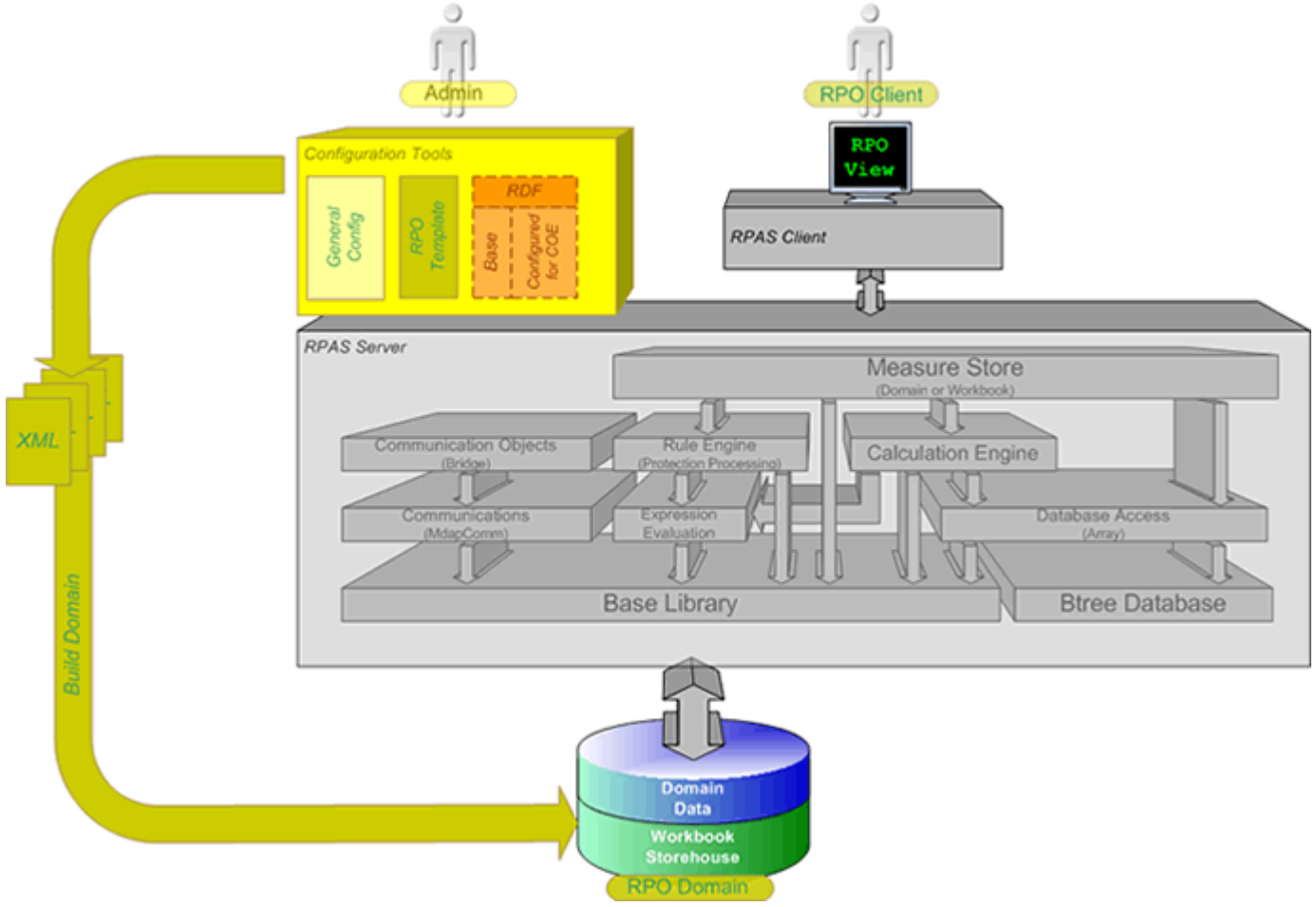


Figure 1–1 describes the RPO application. In the truest sense, RPO and other RPAS applications are not applications in the same way that the RPAS client is an application, since end users are not presented a user interface specific to the template. RPO and other applications are predefined means to view specific types of data in the domain such that the RPAS client user interface is used to read and write to the domain.

The system administrator responsible for maintaining the RPAS Configuration Tools ensures that the appropriate templates are available. Each template has the following associated information to define its predefined attributes:

- Measures
- Special expressions
- Rules
- Workbook layout

A client requests use of one of the templates by using the Configuration Tools. A number of XML files are then output which define the domain to be created. These XML files are used to build the specified domain incorporating all the attributes mentioned above that have been defined specifically.

Once the domain has been created, the end user can access the domain data through the RPAS client.

Key Features of RPO

RPO provides the following features:

- Optimization of retail, grocery, and hard-line item regular prices for a defined time period subject to constraints.
- Choice of optimization metrics: revenue, margin, highest item price, volume, Competitive Price Index (CPI), or any combination thereof.
- Attention to global constraints on revenue, margin, and volume.
- Attention to item and group price range and inter-item constraints.
- Attention to discrete price ladder constraints.
- Differentiation of prices across price zone or store level.

Skills Needed for Implementation

The implementer needs an understanding of 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
- How to set up an RPAS domain
- A solid understanding of RPAS configuration and how to use the RPAS Configuration Tools
- Technical architecture of RPAS-based applications
- Understanding of how RPAS rule language works
- Understanding of measures and dimension constructs

- Basic merchandising
- Basic forecasting
- The interaction of prices and volumes for items in assortment

2

Implementation Considerations

The following information needs to be considered before implementing RPO:

- Input Data
- Item and Cross-Item Elasticities Data
- Hardware Space Impacts
- Domain Partitioning
- Formatting
- Plug-ins
- Patch Considerations
- Batch Scheduling
- Optimization Options
- Security
- Internationalization

Input Data

RPO uses the following required data:

- Base demand
- Current cost
- Current price
- Category - scenario group mapping

The following is optional data:

- Price ladder values
- Unit of measure (UOM)
- Equivalent unit of measure (EUOM)
- Sales history
- Price history
- Future promotion plans
- Actual prices
- Competition prices

- Item and cross-item elasticities
- Item default price ladder assignments
- Category/store to price zone mapping

Item and Cross-Item Elasticities Data

Item and cross-item elasticities quantify the effect of price on volume for both a single item and for other items whose demand is correlated with that item. Elasticity data is used in various capacities for the different optimization options. For the rule management option, elasticity data is not needed. For the price simulation option, just item elasticities are required. For the full optimization option, both item and cross-item elasticities are required. For more information, see Optimization Options.

Hardware Space Impacts

The following factors can affect hardware space requirements:

- SKU - Number of distinct items.
- Store - Number of physical, Web, and other distinct retail outlets.
- Price Zones - Number of distinct price zones into which stores are grouped.
- Calendar - Number of historical and future time periods in the domain. This impacts the overall size of the environment.
- Workbooks - Amount of space used by workbooks. This is typically greater than the domain itself. The number of workbooks is related to the number of users.

Domain Partitioning

Partitioning is done to avoid contention for resources. Building a workbook and committing data are two processes that can cause contention.

How data is partitioned has an impact on the business process. The RPO domain is defined as a global domain. For performance reasons, a simple domain is not recommended. There should be an even distribution of users across a set of local domains. For example, dairy products could be in one domain, dry cereal in another, and canned goods in a third. When a user is committing data in the dairy products domain, it does not affect the users in the dry cereal or canned goods domains because of the use of partitioning.

Consider the following questions when defining the partitioning of the domain:

- How do I partition to meet my business needs?
- How do I partition my users?
- How do I create groups of users to further partition the solution?

Domain partitioning is supported on any dimension of the Merchandise (PROD) hierarchy at or above Category. It is recommended that the RPO domain only be partitioned along the Product hierarchy. It is not recommended that the user partition along the Location (LOC) hierarchy, primarily because Price Zone is not along the partitioned branch (it is an alternate, dynamic rollup) of the Location hierarchy, and all Price Zone level data would be stored in the global domain.

In RPO, measures are not created during workbook creation.

Formatting

Formatting can be done in the configuration or the workbook after the domain is built.

- Using RPAS Configuration Tools, an implementer can create generic styles for the measures and assign them to measure components or realized measures. For each measure, these styles can be overridden on each workbook template. For more information, see the *Oracle Retail Predictive Application Server Configuration Tools User Guide*.
- Once the domain is built, the implementer can set up worksheet sizes and placements, exception value formatting, gridlines, and other formatting. The implementer instantiates a workbook of the template to set up specific formatting by using the Format menu. The updated format is then saved to the template so that it is available to all users for any newly created workbooks. For information on how to edit formatting, see the *Oracle Retail Predictive Application Server User Guide for the Classic Client* or the *Oracle Retail Predictive Application Server User Guide for the Fusion Client*.

Plug-ins

Note: There is no Configuration Tools plug-in for RPO.

Plug-ins are application-specific Java code modules that run inside and automate the RPAS Configuration Tools to assist the implementer with specific application configuration. There are rules that an implementer must follow when configuring an application. A plug-in makes such adherence easier by automating parts of the configuration process and validity-checking entries that are made.

Patch Considerations

With a new release, there are two types of patches that can affect the RPO domain:

- Changes to the code in the RPAS libraries

  The configuration is not affected by this type of patch. For these types of changes, applying the patch is a straight forward process.

- Changes to the configuration

  These types of changes can be more complex. If you have customizations in the configuration, you can use the rpasConfigMgr utility to determine the differences between your existing configuration and the new one. Then, you can use the utility to merge the two configurations. Any changes that cannot be applied are written to a change log. For more information, see the *RPAS Configuration Tools User Guide*.

Batch 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.

RPO has batch scripts for loading hierarchies and data, generating the forecast (if using embedded forecast), scenario optimization, and exporting recommendations. The time that batch scripts are scheduled to run varies by implementation.

Optimization Options

Many retailers want a way to manage rules for everyday pricing, while still maintaining the flexibility and foundation to layer on elasticity understanding and optimization as their organization and pricing practices mature. Some retailers just need to manage their rules (competitive rules, margin requirements, price families, inter-item rules) and do not need to consider elasticities or optimization. Other retailers would like to use elasticities to predict and understand volumes. Then, there are some retailers who would not only like to manage rules and consider elasticities but also perform optimization to recommend prices.

As a result of these business needs, RPO can be used in three different ways, allowing the user to benefit from the various degrees of complexity of RPO. These three options—rule management, price simulation, and full optimization— are described below.

Rule Management

Rule management provides robust rules management capability. It does not consider price elasticities; it only satisfies the constraints set by the user. Using rule management, the price change does not impact the sales volume during the what-if simulation. To use RPO for rule management, select rule management as the optimization capacity option in the Price Analysis workbook.

Price Simulation

Price simulation provides robust rules management capability and what-if simulation which shows the item volume/financial impact based on the item elasticity. Using price simulation, the price change impacts the sales volume during the what-if simulation, but it does not consider the cross-item elasticities. To use RPO for price simulation, select price simulation as the optimization capacity option in the Price Analysis workbook.

Full Optimization

Full optimization builds on the functionality of price simulation, but it also considers cross-item elasticities in the what-if simulation and in the optimization. In a full optimization, the revenue, gross margin, volume, CPI, and combination of global goals can be employed. By using cross-item elasticities, RPO captures cannibalization and halo effects among items. Using full optimization, the price change impacts the sales volume during the what-if simulation, but both item elasticity and cross-item elasticity are considered.

To use RPO for full optimization, select full optimization as the optimization capacity option in the Price Analysis workbook.

Changing Optimization Options

To change RPO from one optimization option to another, there is no need to patch the domain or upgrade any code. However, for the price simulation option, you must load the item elasticity data. For the full optimization option, you must load both the item elasticity data and the cross-item elasticity data.

For more information about using the different optimization options, see the *RPO User Guide for the RPAS Classic Client* or the *RPO User Guide for the RPAS Fusion Client*.

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 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 cannot edit any values or commit the workbook. The read-only workbook can be refreshed.

When users save workbooks, they assign one of three access permissions:

- World - Allow any user to open and edit the workbook.
- Group - Allow only those users in their same group to open and edit the workbooks.
- User - Allow no other users to open and edit the workbook.

Note: A user must have access to the workbook template in order to access the workbook, even if the workbook has world access rights.

For more information on security, see the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* or the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client.*

Internationalization

For more information on translation for RPO, see Chapter 9, "Internationalization".

3

Build Scripts

This chapter describes the setup that must be done before building the RPO domain and the batch script that must be executed to build the domain.

Installation Dependencies

RPAS and RPO must be installed before setting up and configuring RPO.

- For information on installing RPAS, see the *Oracle Retail Predictive Application Server Installation Guide*.
- For information on installing RPO, see the *Oracle Retail Regular Price Optimization Installation Guide*.

Environment Setup

Before downloading the installation package to the UNIX server, a central directory structure to support the environment needs to be created. This central directory is referred to as `<rpo_directory>`. Set `<rpo_directory>` to the full path name to RPO home.

Note: The `$RPO_HOME` variable is not used.

RPAS Installation

The Java-based RPAS installation programs that are included with the installation package are used to install the server-side RPAS components on UNIX operating systems.

The RPAS installer performs the following functions:

- Installs the RPAS server
- Installs the Configuration Tools on the server

  On Windows, an InstallShield package is used to install the Configuration Tools.
- Defines the DomainDaemon port

RPAS Client Installation

The RPAS server installation package also includes the following RPAS clients:

- RPAS Classic Client: A Windows-based client interface for end users and system administrators of an RPAS domain.
- RPAS Fusion Client: A Web-based client developed using Oracle Application Development Framework (ADF). It includes the features available in the RPAS Windows-based client and meets the scalability requirements set for the RPAS platform.

Each RPAS client installation package includes a separate installer to help you install the client. For more information on installing the RPAS clients, refer to the *Oracle Retail Predictive Application Server Installation Guide*.

RPO Installation

In addition to the RPAS installer, the installation package also includes the Java-based RPAS installation program for the RPO application.

The RPO installer performs the following functions:

- Downloads the configuration and batch scripts into the `<rpo_directory>/config` and `<rpo_directory>/bin` directories
- Downloads a set of sample hierarchy and data files into the `<rpo_directory>/input` directory
- Builds a sample domain at `<rpo_directory>/domain/RPO`

Custom Domain Build

To do a custom build of a domain:

1. Update the globaldomainconfig.xml file with the correct domain paths.
2. If needed, update the default environment variables in environment.sh.
3. Execute the buildRegPrice.sh script: `./buildRegPrice.sh`

RPO Taskflow for the RPAS Fusion Client

The RPO installation software enables you to install the taskflow and online help files for the RPAS Fusion Client. In order to install the taskflow files, the RPAS Fusion Client must already be installed. For more information on installing the RPAS Fusion Client, see the *Oracle Retail Predictive Application Server Installation Guide*.

During the RPAS Fusion Client installation, the installer automatically sets up the RPAS domain connection configurations in the ProfileList.xml file. If you choose to set up the domain connection after the installation or set up an additional domain, you must manually set up the connection. For more information, see the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

Environment Variables

In addition to the regular RPAS environment variables, including RPAS_HOME, you need to set up the following environment variables and export them:

- RIDE_HOME
- PATH

Files Needed to Build the RPO Domain

Before the domain is built, the following types of files need to be set up:

- Standard RPAS Hierarchy Files
- RPO Hierarchy Files
- Required Data Files

Standard RPAS Hierarchy Files

The following hierarchy files are needed:

- Calendar (CLND) Hierarchy File
- Merchandise or Product (PROD) Hierarchy File
- Location (LOC) Hierarchy File

> **Note:** As with all standard RPAS hierarchies, these hierarchies are configurable. For information about configuring these hierarchies, see Chapter 7, "Configuration Considerations".

Calendar (CLND) Hierarchy File

File name: clnd.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–1 Calendar Hierarchy Fields

Field	Description
Day	Day ID
Day label	Day label
Week	Week ID
Week label	Week label
Mnth	Month ID
Mnth label	Month label
Qrtr	Quarter ID
Qrtr label	Quarter label
Year	Year ID
Year label	Year label

Example:

```
20050130,01/30/2005,w01_2005,01/30/2005,JAN_2005,January 2005,Q1_2005,Quarter 1
2005,A2005,Year 2005
```

Merchandise or Product (PROD) Hierarchy File

File name: prod.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–2 Merchandise or Product Hierarchy Fields

Field	Description
Item	SKU ID
SKU label	SKU label
Clss	Class ID
Clss label	Class label
Scat	Sub Category ID
Scat label	Sub Category label
Cat	Category ID
Cat label	Category label
Dept	Department ID
Dept label	Department label
Chn	Chain ID
Chn label	Chain label
Brnd	Brand ID
Brand label	Brand label

Example:

```
10426485,Bskt Lg Gld Wlw W/Red Vlv Lnr,217771339880,Cmas Storage
Baskets,21777133,Basket Collections,21777,Baskets & Storage,217,Home Decor,0,Chain
Top Level, BrandA, Brand A
```

Location (LOC) Hierarchy File

File name: loc.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–3 Location Hierarchy Fields

Field	Description
Str	Store ID
Str label	Store label
Zone	Zone ID
Zone label	Zone label
Regn	Region ID
Regn label	Region label
Cnty	Country ID
Cnty label	Country label
Chn	Chain ID
Chn label	Chain label
Entp	Enterprise ID
Entp label	Enterprise label
Przn	Price Zone ID
Przn label	Price Zone label

Example:

```
2068,Atl-Cumming,8,Atlanta GA DMA,H2,Hobby Mart,1,US,1,retailer chnl
,0,retailer,9,Price Zone 9
```

RPO Hierarchy Files

The following are required hierarchy files for RPO:

- RHS Merchandise or Product (PROR) Hierarchy File
- Competition Merchandise or Product (COMP) Hierarchy File
- Price Ladder Hierarchy File
- Price Point Hierarchy File
- Item Group Hierarchy File
- Inter-Item Group Constraint Hierarchy File
- Competition Item Group Constraints Hierarchy File
- Scenario Hierarchy File

RHS Merchandise or Product (PROR) Hierarchy File

RHS Merchandise or Product (PROR) hierarchy has the same format and content as Product (PROD) hierarchy. You should copy the Product hierarchy file and load it again for the RHS Product hierarchy.

File name: pror.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–4 RHS Merchandise or Product Hierarchy Fields

Field	Description
Item	SKU ID
SKU label	SKU label
Clss	Class ID
Clss label	Class label
Scat	Sub Category ID
Scat label	Sub Category label
Cat	Category ID
Cat label	Category label
Dept	Department ID
Dept label	Department label
Chn	Chain ID
Chn label	Chain label
Brnd	Brand ID
Brand label	Brand label

Example:

```
10426485,Bskt Lg Gld Wlw W/Red Vlv Lnr,217771339880,Cmas Storage
Baskets,21777133,Basket Collections,21777,Baskets & Storage,217,Home Decor,0,Chain
Top Level, BrandA, Brand A
```

Competition Merchandise or Product (COMP) Hierarchy File

File name: comp.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–5 Competition Merchandise or Product Hierarchy Fields

Field	Description
Cmpt	Competitor ID
Cmpt label	Competitor label

Example:

```
CompA,Competitor A
```

Price Ladder Hierarchy File

A price ladder is a collection of acceptable price points for an item. Price ladders are loaded to ensure appropriate recommended prices are provided based upon optimization results. RPO only recommends a price that is a price point on the price ladder.

For example, if the optimization engine recommends a price of $11.93, but your price strategy is that all prices must end in .00 and price breaks must occur every $2.00, then RPO recommends $12.00 instead of $11.93.

> **Note:** Only a single price ladder can be deployed for each item at a time. It is suggested to load as few price ladders as possible.

File name: pl.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–6 Price Ladder Hierarchy Fields

Field	Description
Pldr	Price Ladder ID
Pldr label	Price Ladder label

Example:

```
1,Price Ladder from 0.99 to 99.99
```

Price Point Hierarchy File

A price point is an acceptable price for an item. A price ladder is a collection of price points for an item. RPO only recommends a price that is a price point on the price ladder.

File name: pp.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–7 Price Point Hierarchy Fields

Field	Description
Ppnt	Price Point ID
Ppnt label	Price Point label

Example:

```
1,price point 001
2,price point 002
3,price point 003
…
99,price point 099
```

Item Group Hierarchy File

Item groups can be created manually within the RPO application or they can be loaded in a periodic hierarchy load. There are two major purposes for item groups in RPO. First, item groups allow the user to group all the items together and set constraints for all of them rather than for each item individually. For example, if you have an item group called Brand X Soda, you can set group level constraints so that the price of Brand X soda is always less than Brand Y soda.

Secondly, it allows users to set up the price family constraint. For example, if a group is defined for all Brand X 12-ounce sodas (diet, regular, caffeine free, cherry), you can designate a price family to that group so that all types of Brand X's 12-ounce soda have the same price. See the Item Group Loading section in this chapter for details on how to load the item group.

File name: ig.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–8 Item Group Hierarchy Fields

Field	Description
Igrp	Item Group ID
Igrp label	Item Group label

Example:

```
itgp001,Item Group Example
```

Item Group Constraint Hierarchy File

This hierarchy is used to assign a unique identifier to constraints for individual items or item groups.

File name: ic.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–9 Item Group Constraint Hierarchy Fields

Field	Description
Iccc	Item Group Constraint ID
Iccc label	Item Group Constraint label

Example:

```
C001, constraint 001
```

Inter-Item Group Constraint Hierarchy File

Inter-item group constraints hierarchy is used to assign a unique identifier to each individual inter-item group constraint. Inter-item group constraints can be created manually within the RPO application, but they are usually loaded in a periodic hierarchy load.

File name: iig.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–10 Inter-Item Group Constraint Hierarchy Fields

Field	Description
Iigc	Inter-item Group Constraint ID
Iigc label	Inter-item Group Constraint label

Example:

```
C001,Inter-item group constraint 001
```

Competition Item Group Constraints Hierarchy File

Competition Item Group Constraints hierarchy is used to assign a unique identifier to each individual competition item group constraint. Competition item group constraints can be created manually within the RPO application, but they are usually loaded in a periodic hierarchy load.

File name: cig.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–11 Competition Item Group Constraints Hierarchy Fields

Field	Description
Cigc	Competition Item Group Constraint ID
Cigc label	Competition Item Group Constraint label

Example:

```
C001,Competition item-group constraint 001
```

Scenario Hierarchy File

Scenario hierarchy is a dynamic hierarchy which can be loaded in a periodic hierarchy load, but it is usually created manually within the RPO application. This hierarchy requires an example position. Therefore, you should include `example, example` in the first line of hierarchy file.

File name: scn.csv.dat
File format: comma-separated values file

The following table describes the fields in the file:

Table 3–12 Scenario Hierarchy Fields

Field	Description
Scnr	Scenario ID
Scnr label	Scenario label
Srgp	Scenario Group
Srgp label	Scenario Group label

Example:

```
example,example,
Max_margin,max margin scenario, softdrink, soft drink scenario group
```

Required Data Files

The following data files are required:

- Base Demand File
- Current Item Cost File
- Current Item Price File
- Category - Scenario Group Mapping File

> **Note:** In this guide, all the required data files are assumed to be at the item/price zone optimization level. The location dimension in the file's intersection means price zone.

Base Demand File

The demand forecast for the item at a specific location and week. It is assumed that this forecast already includes the seasonality factor. The location dimension is dependent on the optimization level, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

File name: dl1itbdsp.csv.rpl
File format: comma-separated values file
Intersection: item/*location*/week

The following table describes the fields in the file:

Table 3–13 Base Demand Fields

Field	Description
Week	Week ID
Item	Item ID
Location	Optimization level's location dimension
Value	The forecast value

Example:

```
w01_2006,10033240,6,8.50
```

Current Item Cost File

The current cost of the item at a location. The location dimension is dependent on the optimization level, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

File name: dl1orgitcst.csv.rpl
File format: comma-separated values file
Intersection: item/*location*

The following table describes the fields in the file:

Table 3–14 Current Item Cost Fields

Field	Description
Item	Item ID
Location	Optimization level's location dimension

Table 3–14 (Cont.) Current Item Cost Fields

Field	Description
Cost Value	Item cost at a certain location

Example:

```
10033240,1,3.17
```

Current Item Price File

The current price of the item at a location. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

File name: ol1orgitpc.csv.rpl
File format: comma-separated values file
Intersection: item/*location*

The following table describes the fields in the file:

Table 3–15 Current Item Price Fields

Field	Description
Item	Item ID
Location	Optimization level's location dimension
Price Value	Item price at a certain location

Example:

```
10033240,1,4.12
```

Category - Scenario Group Mapping File

This file defines the categories and scenarios that are associated to one another.

File name: scncatmp.csv.rpl
File format: comma-separated values file
Intersection: category/scenario group

The following table describes the fields in the file:

Table 3–16 Category - Scenario Group Mapping Fields

Field	Description
Cat	Category ID
Srgp	Scenario Group ID
Value	Set to "T" in order to map the category and scenario. The absence of a record means there is no mapping between the category and scenario.

Example:

```
Srgp1, Cat1, T
```

Optional Data Files

These data files are optional:

- Price Ladder Value File
- Item Default Price Ladder Assignment File
- Unit of Measure File
- Equivalent Unit of Measures File
- Sales History File
- Future Promotion Plan File
- Actual Price File
- Competitor Current Price File
- Competitor Last Price Check Date File
- Competitor Price Type File
- Category/Store Price Zone Mapping File
- Cross-Item Elasticities File
- Minimum History Price File
- Maximum History Price File
- Anchor Price File

Price Ladder Value File

The price ladder defines the price value of each price point on a price ladder. If this file is not loaded, users can generate price ladders in the Business Administration workbook.

File name: pldrppt.csv.rpl
File format: comma-separated values file
Intersection: price ladder/price point

> **Note:** Only a single price ladder can be deployed for each item at a time. It is suggested to load as few price ladders as possible.

The following table describes the fields in the file:

Table 3–17 Price Ladder Value Fields

Field	Description
Pldr	Price Ladder ID
Ppnt	Price Point ID
Value	The price value

Example:

```
1,1,0.49
2,1,0.59
3,1,0.69
```

Item Default Price Ladder Assignment File

The item's default price ladder used when pricing for a specified location. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

The item default price ladder can be set in the Business Administration workbook.

File name: ol1itpcldr.csv.rpl
File format: comma-separated values file
Intersection: item/*location*

The following table describes the fields in the file:

Table 3–18 Item Default Price Ladder Assignment Fields

Field	Description
Item	Item ID
Location	Optimization level's location dimension
Value	The default Price Ladder ID

Example:

```
21777131,1,1
21777131,2,1
```

Unit of Measure File

Unit of Measures (UOM) describes the unit of measure of the item (such as 6-pack, 12-pack, case, and so on). The UOM is not a required attribute for every item. It is useful when the user needs to make a distinction between several pack sizes of the same product. For example, Brand X packages its canned soda in 6-packs and 12-packs. The UOMs for those two products are 6 and 12, respectively.

This file is necessary if the user needs to set up inter-item constraints at UOM type. The unit price of 6-pack soda should be greater than the unit price of the 12-pack.

Note: The Unit of Measure (UOM) must be an integer.

File name: ol1ituom.csv.rpl
File format: comma-separated values file
Intersection: item

The following table describes the fields in the file:

Table 3–19 Unit of Measure Fields

Field	Description
Item	Item ID
UOM Value	The sellable unit packsize. This must be an integer.

Example:

```
10213430 , 6
```

Equivalent Unit of Measures File

Equivalent Unit of Measures (EUOM) describes the multiple of a standard unit of measure (such as ounces, pounds, liters) contained in one unit. Like UOM, EUOM is not a required attribute for every item. It is useful when the user needs to make a distinction between several pack sizes, pack weights, or volumes of the same product. For example, Brand X packages its bottled soda in 1- and 2- liter bottles. The EUOM of the 1-liter soda is 1.0. The EUOM of the 2-liter is 2.0.

This file is needed if the user needs to set up inter-item constraints at EUOM type. The unit price of the 1 liter soda should be greater than the unit price of the 2 liter.

File name: ol1itequiu.csv.rpl
File format: comma-separated values file
Intersection: item

The following table describes the fields in the file:

Table 3–20 Equivalent Unit of Measures Fields

Field	Description
Item	Item ID
EUOM Value	The multiple of a standard unit of measure (such as ounces, pounds, etc.) contained in one unit.

Example:

```
10213430 , 2.0
```

Sales History File

The sales history of an item. This is needed if the user wants to generate a baseline forecast using the embedded forecasting functionality in RPO.

File name: pos.csv.rpl
File format: comma-separated values file
Intersection: item/str/*week*

The following table describes the fields in the file:

Table 3–21 Sales History Fields

Field	Description
Week	Week ID
Item	Item ID
Str	Store ID
Sales	The sales units.

Example:

```
Week1, 5785456, Store1, 4.49
```

Future Promotion Plan File

The future promotion plan that is displayed for reference only. If this file is not loaded, it does not impact the functionality of RPO.

File name: ol1futprom.csv.rpl
File format: comma-separated values file
Intersection: item/location/*week*

The following table describes the fields in the file:

Table 3–22 Future Promotion Plan Fields

Field	Description
Week	Week ID
Item	Item ID
Location	The optimization level's location dimension.
Promo flag	If set to T, a promotion is planned.

Example:

```
Week1, 5486599, przn1, T
```

Actual Price File

The actual price of an item executed in the store. If this file is not loaded, it does not impact the functionality of RPO.

File name: ol1exeltpc.csv.rpl
File format: comma-separated values file
Intersection: item/location/week

The following table describes the fields in the file:

Table 3–23 Actual Price Fields

Field	Description
Week	Week ID
Item	Item ID
Location	The optimization level's location dimension.
Price	The actual item price executed in the store.

Example:

```
Week1, 5486599, przn1, 4.49
```

Competitor Current Price File

The competitor's current price of an item at that location. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

This file is needed if users want to create competition constraints.

File name: ol1cmpcurpc.csv.rpl
File format: comma-separated values file
Intersection: item/cmpt/*location*

The following table describes the fields in the file:

Table 3–24 Competitor Current Price Fields

Field	Description
Item	Item ID
Cmpt	Competitor ID
Location	Optimization level's location dimension
Price	Competitor's Item Price

Example:

```
5486599, CompA, przn1,4.49
```

Competitor Last Price Check Date File

The date of the last time the competitor's item price was checked at that location. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

This file is used only to display the competitor's price checked date. If this file is not loaded, it does not impact the functionality of RPO.

File name: ol1cmppcchkdt.csv.rpl
File format: comma-separated values file
Intersection: item/cmpt/*location*

The following table describes the fields in the file:

Table 3–25 Competitor Last Price Check Date Fields

Field	Description
Item	Item ID
Cmpt	Competitor ID
Location	Optimization level's location dimension
Date	The date of the last time the competitor's item price was checked at that location

Example:

```
5486599, CompA, przn1,20080212
```

Competitor Price Type File

The competitor's item price type, such as promotion price or regular price, at that location. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone. If the competitor price type is not loaded, it does not impact the functionality of RPO.

File name: ol1cmppctp.csv.rpl
File format: comma-separated values file
Intersection: item/cmpt/*location*

The following table describes the fields in the file:

Table 3–26 Competitor Price Type Fields

Field	Description
Item	Item ID
Cmpt	Competitor ID
Location	Optimization level's location dimension
Type	Competitor's item price type, such as promotional, regular, etc.

Example:

```
5486599, CompA, przn1,Promo Price
```

Category/Store Price Zone Mapping File

Defines how a store maps to the price zone for various merchandise categories. This is not needed if the store to price zone rollup does not depend on the merchandise or product hierarchy.

File name: pzdynhrmp.csv.rpl
File format: comma-separated values file
Intersection: cat/store

The following table describes the fields in the file:

Table 3–27 Category/Store Price Zone Mapping Fields

Field	Description
Cat	Category ID
Str	Store ID
Price Zone Value	Price Zone ID

Example:

```
Cat_21777,store_1536,pricezone_7
```

Cross-Item Elasticities File

Defines the cross-item elasticity among items. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

The cross-item elasticities data can come from the APC-RPO application or any other analytic application.

File name: ol1gammasp.csv.rpl
File format: comma-separated values file
Intersection: item/RHS item/*location*

The following table describes the fields in the file:

Table 3–28 Cross-Item Elasticities Fields

Field	Description
Item	Item ID
Iter	Right Hand Side (RHS) Item ID
Location	Optimization level's location dimension
Value	Cross-item elasticity value.

Example:

```
10033240,10320667,1,-0.0192
10033240,10320665,1,0.0104
10033240,10320664,1,-0.0051
10033240,10320661,1,0.0465
```

Minimum History Price File

Defines the minimum historical regular price used in the price elasticity calculation. The minimum history price is used in RPO for risk calculation. If this file is not provided, the simple risk model is used.

The minimum history price data can come from the APC-RPO application or any other analytic application.

File name: ol1hisloprc.csv.rpl
File format: comma-separated values file
Intersection: item/*location*

The following table describes the fields in the file:

Table 3–29 Minimum History Price Fields

Field	Description
Item	Item ID.
Location	Optimization level's location dimension
Value	Minimum history price

Example:

```
54886599, Store1, 4.49
```

Maximum History Price File

Defines the maximum historical regular price used in the price elasticity calculation. The maximum history price is used in RPO for risk calculation. If this file is not provided, the simple risk model is used.

The maximum history price data can come from the APC-RPO application or any other analytic application.

File name: ol1hishipric.csv.rpl
File format: comma-separated values file
Intersection: item/*location*

The following table describes the fields in the file:

Table 3–30 Maximum History Price Fields

Field	Description
Item	Item ID
Location	Optimization level's location dimension
Value	Maximum history price

Example:

```
54886599, Store1, 6.49
```

Anchor Price File

Defines the anchor price of an item, which is used as the baseline price when calculating price elasticity. If this file is not provided, the current price is used as the anchor price to calculate price drift metrics.

The anchor price data can come from the APC-RPO application or any other analytic application.

File name: ol1anchprc.csv.rpl
File format: comma-separated values file
Intersection: item/*location*

The following table describes the fields in the file:

Table 3–31 Anchor Price Fields

Field	Description
Item	Item ID
Location	Optimization level's location dimension
Value	Anchor Price

Example:

```
54886599, Store1, 3.99
```

Loading Optional Dynamic Hierarchies

The following section describes how to load the item group hierarchy, which is optional. If the item group hierarchy is not loaded, it does not impact the functionality of RPO.

Item Group Loading

The steps to load the item group are as follows.

1. Load Item Group Hierarchy File

See the Item Group Hierarchy File section in this chapter for instructions.

2. Load Item Group Description

Defines the item group's description.

File name: igdesc.csv.rpl
File format: comma-separated values file
Intersection: igrp (Item Group ID)

Table 3–32 Item Group Description Fields

Field	Description
Igrp	Item Group ID
Description	Item Group description

Example:

```
Igrp1,description of item group igrp1
```

3. Load Item/Item Group Association

Defines the item group and item association. This association is used to range the data while building the workbook.

File name: ol1igitasmt.csv.rpl
File format: comma-separated values file
Intersection: igrp/item (Item Group ID, Item ID)

Table 3–33 Item/Item Group Association Fields

Field	Description
Item	Item ID
Igrp	Item Group ID
Value	Set to "T" to associate the item to the item group. The absence of a record means there is no association.

Example:

```
5486599,Igrp1, T
```

Output from RPO to Legacy Systems

RPO creates two outputs for legacy systems:

- Final Price Recommendation
- Item Final Forecast Volume Recommendation

RPAS has a standard utility to export the data to a CSV flat file.

Final Price Recommendation

The final price recommendation. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

File name: fappltpc.csv
File format: comma-separated values file
Intersection: item/week/*location*

Table 3–34 Item and Cross-Item Elasticities Fields

Field	Description
Item	Item ID
Week	Week ID
Location	Optimization level's location dimension
Price	Output price from RPO

Example:

```
5486599,week01,store1,5.99
```

Item Final Forecast Volume Recommendation

The item's final forecast volume. The location dimension is dependent on the optimization level the user chooses, meaning that if the optimization level is item/price zone, then the location dimension is price zone.

File name: fappltvol.csv
File format: comma-separated values file
Intersection: item/week/*location*

Table 3–35 Item and Cross-Item Elasticities Fields

Field	Description
Item	Item ID
Week	Week ID
Location	Optimization level's location dimension
Volume	Output forecast volume from RPO

Example:

```
5486599,week01,store1, 12.0
```

Building the RPO Domain

The script used to build or patch the RPO domain is described in this section. The script is located in the <rpo_directory>/batch directory.

Batch Designs

This section contains detailed information on the buildRegPrice.sh build script:

Building a Domain

Script
buildRegPrice.sh

Usage
buildRegPrice.sh <options -cdilbo> <flags -tg>

Table 3–36 BuildRegPrice.sh Build Script

Argument	Allowed Values	Description
Options	c	Configuration directory Default is <rpo_directory>/config
	d	Domain path Default is <rpo_directory>/domain
	i	Input directory Default is <rpo_directory>/input
	l	Log directory Default is <rpo_directory>/logs
	b	Demand Level
	o	Optimization Level
Flags	t	Set this flag to make a test build
	g	Set this flag to use debug function libraries (no argument)
	p	Set this flag to make a patch build

Notes:

- The script uses the Configuration Tools rpasInstall utility to build a domain. See the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* or the *Oracle Retail Predictive Server Administration Guide for the Fusion Client* for details on this utility.
- The script also uses the mace and loadmeasure RPAS utilities. See the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* or the *Oracle Retail Predictive Server Administration Guide for the Fusion Client* for details on this utility.
- All hierarchy and measure files are placed in <rpo_directory>/input

Creating Users and User Groups

For greater security, users and user groups are not automatically created when you build or patch a domain. To create users and user groups, you must use the usermgr utility. To learn more about usermgr, see the Operational Utilities chapter of the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* or the *Oracle Retail Predictive Server Administration Guide for the Fusion Client*.

Loading and Extracting Data

Data is loaded into RPO using the standard RPAS approach. See the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* or the *Oracle Retail Predictive Server Administration Guide for the Fusion Client* for details on formatting the load data files and on the utilities that enable administrators to load data into RPAS.

For specific information about loading and extracting data in RPO, see Chapter 4, "Loading and Extracting Data".

4

Loading and Extracting Data

Data is loaded into RPO using the regular RPAS approach. See the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* or the *Oracle Retail Predictive Server Administration Guide for the Fusion Client* for details on formatting load data files and for details on utilities that enable administrators to load data into RPAS.

It is recommended that CSV (comma separated variable) format files be used to reduce the size of load files.

The following subsections list the measures that should be loaded to load a functionally coherent set of data, such as demand data. These sections list the measure names and detail any referential integrity requirements.

Loading Demand Data

The following table lists all measures that must be loaded for every item/location if using the Full Optimization option as the optimization capacity.

Table 4–1 Demand Data Measures

Measure	Measure ID	Intersection
Base Demand	dl1itbdsp	Product / Demand Level Location
Cross-item Elasticity	ol1gammasp	Product / Optimization Level Location

Loading Prices and Costs

The following table lists all measures that must be loaded for prices and costs for all item locations.

Table 4–2 Price and Cost Measures

Measure	Measure ID	Intersection
Current Cost	dl1orgitcst	Product / Demand Level Location
Current Price	ol1orgitpc	Product / Optimization Level Location

Loading Attribute Data

The following table lists all measures that must be loaded as attributes of items at various optimization levels. These attributes are used in creating price-family constraint, and in creating inter-item constraints that apply on UOM (Unit of Measure) or Equivalent UOMs.

Table 4–3 Attribute Measures

Measure	Measure ID	Intersection
Price Family	ol1pcfml	Merchandise
Unit of Measure	ol1ituom	Merchandise
Equivalent Unit of Measure	ol1itequiu	Merchandise

Loading Price Ladders

Price ladders are series of ascending price points. Each item must have a price ladder assigned to it, or the optimizer displays an error message. Price ladders can be assigned at run time.

The following tables list the hierarchies and measures that must be loaded for price ladders.

Table 4–4 Price Ladder Hierarchies

Hierarchy	Hierarchy ID
Price Point	pp
Price Ladder	pl

Table 4–5 Price Ladder Measures

Measure	Measure ID	Intersection
Price Ladder	pldrppt	Price Point / Price Ladder
Price Ladder Assignment	ol1itpcldr	Product / Optimization Level Location

Loading Competitor Data

The following table lists all measures that must be loaded for competitor data.

Table 4–6 Competitor Data Measures

Measure	Measure ID	Intersection
Competitor Current Price	ol1cmpcurpc	Bottom Level Competitor / Optimization Level Location
Competitor last price check date	ol1cmppcchkdt	Bottom Level Competitor / Optimization Level Location
Competitor price type	ol1cmppctp	Bottom Level Competitor / Optimization Level Location
Competitor to Retailer Item linkage	ol1ilgcomp	Product / Bottom Level Competitor / Optimization Level Location

Loading Dynamic Clustering of Stores to Price Zones

Price Zone clustering must be loaded in RPO. These two measures are used to load the Store to Price Zone mapping. See the section on dynamic hierarchies in the *RPAS Configuration Tools User Guide* for information regarding the content of these measures.

Table 4–7 Price Zone Measures

Measure	Measure ID	Intersection
Price Zone Map	pzdynhrmp	Trigger Product / Store (Location)
Price Zone Label	pzdynhrlbl	Trigger Product / Store (Location)

Loading Item Group Definitions

Item group definitions are not required, but they can be loaded into RPO. Because item groups are owned by demand groups, referential integrity rules have to be followed when creating load files:

- Each loaded Item Group must be assigned to one and only one demand group, even though the measure structure allows many-to-one mapping between demand groups and item group.
- All items in an item group must also be in the assigned demand group.

Table 4–8 Item Group Hierarchy

Hierarchy	Hierarchy ID
Item Group	ig

Table 4–9 Item Group Measures

Measure	Measure ID	Intersection
Description	igdesc	Item Group
Item Group - Item Assignment	ol1igitasmt	Product / Item Group

Extracting Price Recommendations

Only the following measure is meant for extraction from RPO.

Table 4–10 Price and Volume Measures

Measure	Measure ID	Intersection
Item Price Plan	fappltpc	Product / Optimization Level Location / Calendar

5

Script Integration

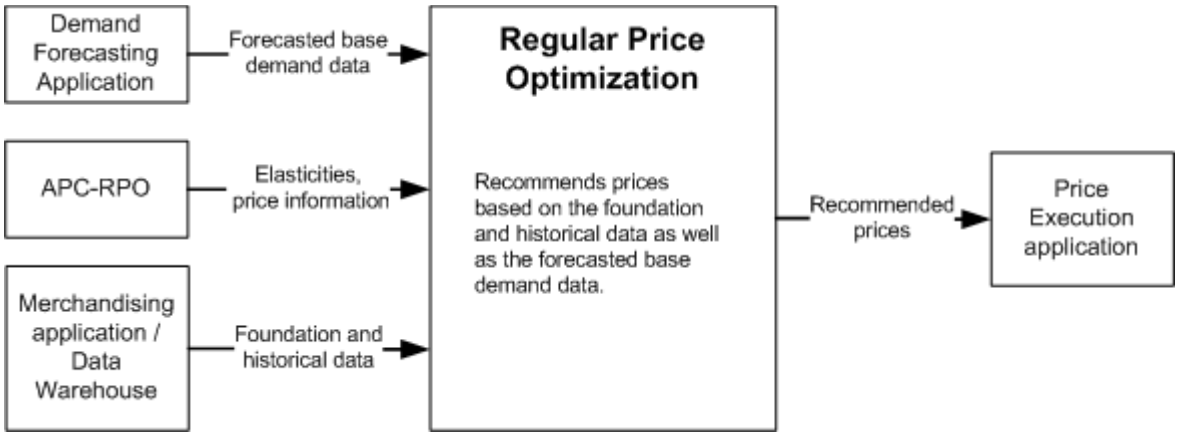
This chapter describes the interaction between RPO and other applications and the script used to load demand data.

For information about ODI integration for RPO, APC-RPO, and RDF, see Chapter 6, "ODI Integration".

Overview of RPO Data Flow

RPO is formally integrated only with APC-RPO and RDF, although it interacts with other applications and data warehouses as well in order to create optimized price recommendations. Figure 5–1 shows the interaction of RPO with other applications and the flow of data among the applications.

Figure 5–1 RPO Data Flow



Integration Interface Data Flow Description

These descriptions explain the each of the data flows in Figure 5–1.

From a Merchandising Application and Data Warehouse to RPO

- Sends foundation and historical data such as item cost, item price, price ladder, item linkage.

From RPO to a Price Execution Application

- Sends recommended prices.

From a Demand Forecasting Applications to RPO

See the From RDF to RPO section.

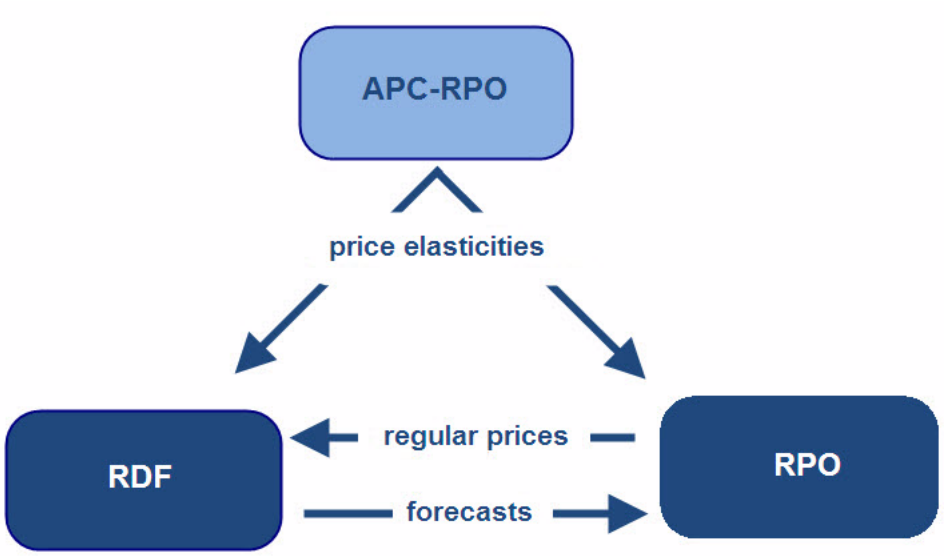
From APC-RPO to RPO

See the From APC-RPO to RPO section.

APC-RPO, RPO, RDF Script Integration

This section describes the integration and data flow among APC-RPO, RPO, and RDF.

Figure 5–2 APC-RPO, RPO, RDF Integration



From APC-RPO to RPO

- Sends item and cross-item elasticities of items. RPO uses these elasticities to optimize prices.
- Sends maximum and minimum historical prices of items. RPO uses this data to optimize prices.
- Sends anchor prices of items. Anchor prices are the baseline prices that APC-RPO uses to calculate price elasticity. RPO uses the anchor prices to calculate price drift metrics.

From APC-RPO to RDF

- Sends regular price item elasticities to RDF. These item elasticities, together with the price plan, allow RDF to calculate the item elasticity lift.
- Sends regular price cross-item elasticities to RDF. There are two types of cross-item elasticities: halo and cannibalization. These cross elasticities, together with the price plan, allow RDF to calculate the cross-item lift for both halo and cannibalization effects related to the corresponding elasticities.
- Sends anchor prices to RDF for reference purposes.
- Sends historical prices to RDF that are used to calculate the regular price lifts.

From RPO to RDF

Sends the price plan that allows RDF to calculate the three components of the regular price lift: regular price item effect, regular price cannibalization effect, and regular price halo effect.

From RDF to RPO

Sends forecasts to RPO. These forecasts represent the base demand at the item/store level. RPO aggregates the forecasts to the item/price zone level and uses that data to optimize prices.

RDF Integration Script

The following script loads base demand data from RDF to RPO:

```
rdf_rpo_integration.sh
```

This script loads forecast demand data at the item/store/week level. By using this specified intersection, the data may come from any forecasting system.

> **Note:** This script can also work for data from other demand forecasting applications if the data is in the correct format.

Use the following data format in the fixed width file.

Table 5–1 Demand Data

Data Entry	Description	Start	Width	Format	Example
Week	Date	1	8	YYYYMMDD	20091031
Item	Item ID	9	20	Item_ID	TZ748562A
Store	Store ID	29	20	Store_ID	0543
Value	The value of the forecasted demand of an item for a specified store during a specified week.	49	13	Any real number	27.40

Example:

```
20090101 TZ748562A            0543                 27.00
```

> **Note:** CSV format is also allowed.

APC-RPO Integration Script

The APC-RPO integration script loads the following data from APC-RPO:

- Minimum historical price
- Maximum historical price
- Anchor price
- Self and cross elasticities

This data is loaded with the following integration script:

`apc_rpo_integration.sh`

6

ODI Integration

This chapter describes the integration of APC-RPO, RPO, and RDF using Oracle Data Integrator (ODI). For information about script integration among these applications, see Chapter 5, "Script Integration".

ODI Integration

Oracle Retail Enabled ODI Integration for APC-RPO, RPO, and RDF is a package that allows users to easily execute data transfers between these applications. It leverages ODI to store information about data interfaces among applications. ODI presents a user-friendly graphical interface for user-initiated data transfers and runtime monitoring. It also provides the ability to host application domains on different machines on a network, an ability not available with prior non-ODI integration strategies.

For more information about the ODI integration, see the *Oracle Retail Enabled Oracle Data Integrator Integration Implementation Guide*.

Measure Data Integration

The following data integration points for each application-to-application package are described in this section:

- APC-RPO to RPO Package
- APC-RPO to RDF Package
- RDF to RPO Package
- RPO to RDF Package

The scripts listed in each section are listed in the following directory:

$RPAS_HOME/scripts/integration/ODI

APC-RPO to RPO Package

The following information is about the APC-RPO to RPO integration package. This package sends anchor prices, self elasticities, and historical minimum and maximum prices from APC-RPO to RPO.

Data Mapping for APC-RPO to RPO Package

Data is sent when the expression is populated with a value other than the default value, which is zero.

Table 6–1 APC-RPO to RPO Data

APC-RPO Expression	RPO Target Measure
exprtelst	ol1gammasp
anchorprice	ol1anchprc
minhistprice	ol1hisloprc
maxhistprice	ol1hishiprc

APC-RPO to RDF Package

The following information is about the APC-RPO to RDF integration package. This package sends anchor prices, self elasticities, and historical minimum and maximum prices from APC-RPO to RDF.

Data Mapping for APC-RPO to RDF Package

Data is sent when the expression is populated with a value other than the default value, which is zero.

Table 6–2 APC-RPO to RDF Data

APC-RPO Expression	RDF Target Measure
achprstr	rdfanchprc
exptelsstr	rdfgamma
price	rdfprice

RDF to RPO Package

The following information is about the RDF to RPO integration package. This package sends item-based demand from RDF to RPO.

Data Mapping for RDF to RPO Package

Data is sent when the expression is populated with a value other than the default value, which is zero.

Table 6–3 RDF to RPO Data

RDF Expression	RPO Target Measure
appf01xb	dl1itbdsp

RPO to RDF Package

The following information is about the RPO to RDF integration package. This package sends item prices from RPO to RDF.

Data Mapping for RPO to RDF Package

Data is sent when the expression is populated with a value other than the default value, which is zero.

Table 6–4 RPO to RDF Data

RPO Expression	RDF Target Measure
fappitpc	rdfprice

7

Configuration Considerations

This chapter provides information on the functional changes or enhancements that can be made for RPO. It is recommended that implementers refrain from removing or changing any existing functionality beyond what is listed here. However, it is acceptable to add new measures and rules in such a way that there is no interference with the operation of existing rules.

Changing the Structure of Hierarchies

RPO has eleven hierarchies, many of which have been created to support abstract notions specific to RPO. The structure of these hierarchies should not be changed (refer to Table 7–1); however, their load formats can be changed to suit the needs of the customer, except that the prefix length for their dimensions should always be zero (0). The structure of the other hierarchies can be changed but is subject to some conditions that are listed in the following subsections.

New hierarchies can be added for a particular implementation and interspersed between existing hierarchies. However, no existing hierarchies can be removed, nor should the order of existing hierarchies be changed.

No changes should be made to dimensions and their configuration unless no other solution is possible. For example, there is no need to rename a dimension if only the label needs to be changed. Keeping the structure and names as standard as possible makes it easier to obtain product support when needed. Any changes that are made should be documented clearly, in case you need to obtain support.

Table 7–1 Non-Modified Hierarchies

Hierarchy Name	Hierarchy Label
CIG	Competition Item Group Constraint
IIG	Inter-Item Group Constraint
IG	Item Group
PL	Price Ladder
PP	Price Point
SCN	Scenario

Calendar (CLND)

The Calendar hierarchy represents time in all RPAS solutions. It is a required hierarchy and must have a dimension named day (DAY). As it relates to RPO, the Calendar hierarchy is needed to store a time series of the seasonality component of the demand forecasts of items. These demand forecasts are used to calculate initial estimates of sales volume for all items.

Seasonality values can be stored at various levels along the Location and Product hierarchies; however, they are all stored along only one level of the Calendar hierarchy. RPO stores seasonality along the week (WEEK) dimension. A particular customer implementation might store seasonality along some other level on the Calendar hierarchy, but this level should represent the calendar and not notions such as week-of-the-month or week-of-the-year that aggregate non-sequential positions from the calendar.

If a customer implementation does store seasonality along a level other than week, the week dimension is not required in the Calendar hierarchy, and the base intersections of all measures in the RPO that utilize the week dimension must be changed to use this other dimension.

Other than abiding by these constraints, a customer implementation can structure the Calendar hierarchy in any way that best suits the customer's functional needs. Dimensions other than day and week have been included in RPO for the purpose of illustration. They can be modified or removed without requiring changes to any other elements of the configuration. Other dimensions and hierarchy branches can also be added without requiring changes to other elements of the configuration.

Merchandise or Product (PROD)

The Merchandise or Product hierarchy represents the retailer's merchandise (that is, merchandise that the retailer retails through its retail channels). RPO does not enforce any constraints on the structuring of this hierarchy, but the configuration does use a few levels of this hierarchy extensively in workbook wizards, labeled intersections, rules, position queries, and measure values (Single Hier Select measures). Any changes to this hierarchy must be accompanied by changes to all these elements if they employ the particular level that is being modified or removed. Adding levels or branches or changing labels should not require any changes to the configuration.

From the hierarchy, the configuration employs Category (CAT), Class (CLSS), and Item (ITEM) in measure intersections, rules, and so on. To reduce patching and upgrade efforts, it is recommended that the names of these levels be left unchanged unless absolutely necessary.

Category is used as the trigger dimension for clustering Store (STR) into Price Zone (PRZN). Class and Item are used in specification of the three optimization and demand levels in the configuration, and changing them requires several changes in the configuration. For more information, see the sections Changing Optimization Levels and RHS Merchandise or Product (PROR) in this chapter.

Note: When adding a new item to RPO, you can assign new parameters to the new item in the Item Management workbook. For more information, see the *RPO User Guide for the RPAS Classic Client* or the *RPO User Guide for the RPAS Fusion Client*.

RHS Merchandise or Product (PROR)

The RHS (right hand side) Product is intended to be a clone of the Product hierarchy. This hierarchy must have the same structure and the same positions as the Product hierarchy. Any changes to the Product hierarchy should be accompanied by the same changes to the RHS Product (PROR) hierarchy. This hierarchy is required to establish item linkage in inter-item pricing constraints. It is also used to store cross-item elasticities when full optimization is selected as the optimization capacity.

For ease of administration and to reduce the possibility of human error in loading hierarchy data, it is recommended that both the RHS Product and Product hierarchies be loaded in the same batch, from the same hierarchy load file.

Competition (COMP)

The Competition hierarchy represents the competitors' information and is required for loading competition prices and for specifying competition constraints.

In order to display the minimum, maximum, and average competition prices across all competitors in the competition hierarchy load file, the MIN, MAX, and AVG positions must be included.

Location (LOC)

The Location hierarchy represents the retailer's retail locations and their roll-ups. RPO imposes a few constraints on the structure of this hierarchy, but for the most part it is flexible.

The Location hierarchy must always have a level named PRZN (Price Zone). This hard-coded name is extensively used by the custom wizard code to filter data at various mappings in order to appropriately manage planning scopes. This must be a dynamic roll-up from the Store (STR) level, and has reasonable infrastructure built around this hierarchical structure. It is not advised to change this structure. Other than STR and PRZN, all other levels are present for illustration purposes only and can be removed or modified as necessary.

Note: When creating a new price zone, it does not have any elasticity parameters until you load the parametric data from a parametric estimator application. For more information, see the *RPO User Guide for the RPAS Classic Client* or the *RPO User Guide for the RPAS Fusion Client*.

Changing Optimization Levels

An optimization level is the product/location level where price recommendations are made. Also known as a planning level, it is the lowest level that prices can be edited and viewed by a user. RPO comes preconfigured with one optimization level: item/price zone. This section discusses how this level can be changed to be along a different intersection.

RPO has hundreds of measures at the item/price zone optimization level, and the intersection of each must be updated if changes occur to the intersection of the optimization level. To make this task easier, RPO employs labeled intersections in the RPAS Configuration Tools. There are a few dozen labeled intersections that encompass all of the measures. By simply updating a labeled intersection, you can update all of the intersections of the measures that are encompassed in that labeled intersection.

For example, if you wanted to change the optimization level from item/price zone to class/chain, then you would open the Configuration Tools interface, navigate to the Labeled Intersections area, and change the intersection specification of all labeled intersections to use the dimension CHNL instead of PRZN and to use CLSS instead of ITEM. This automatically updates the measure intersections.

Once the intersection specification is updated, perform the following:

1. Open the measure management tool and filter all measures that have names starting with Dl# and "Ol#" (# represents the demand level or optimization level).
2. Filter the Single Hier Select To column to those that have ITEM and change that column's value to CLSS.
3. Change the NA Value of the measure named Ol#IntNm to CLSSCHNL, where CLSSCHNL is the RPAS specification of the intersection. Each dimension's RPAS name is concatenated in any order. If the dimension name is fewer than four characters, it is padded with an underscore (_) to the right.
4. Update the Range attribute of the measure DmndLvl to reflect the selection of the new optimization level.

   Note: Making changes to the optimization level requires you to load the input data such as base demand, cost, price, and so on, at the same level.

5. Update the NA Value of the measure named Dl#IntNm to reflect the RPAS specification of the intersection.
6. Change the Range column for the measure name OptLvl to a value that helps you recognize the new level

All measure specifications are updated to account for the change in the optimization level.

Once measures have been updated, rules need to be updated to account for the change. Rule sets named PrcAnl and Rpo_Batch have several rule groups. These updates include changing dimension specification for specific expressions to match the optimization level. In these expressions, M# should be replaced with optimization level along the Product hierarchy, and R# should be replaced with optimization level along the RHS Product hierarchy. L# should be replaced with optimization level along the Location hierarchy.

Adding New Analytic Measures

New measures can be added to any workbook to suit your needs. However, when adding measures and rules to calculate those measures, do not edit any of the existing rules; that is, you should not add expressions to existing rules and should not edit calculation of any measure in RPO (except those that have been explicitly mentioned in this guide).

In several cases it may be required to add measures that are calculated from other measures.

Adding Product Attributes

Attributes and dimension splitting can be used to aid in the creation of item groups. Additionally, attributes can be used in a customer environment to help with price analysis by sorting or grouping (dimension splitting) items based on attributes. Any number of attributes can be added, and RPO does not impose any restriction on the addition of attributes. However, none of the existing attributes can be removed.

8

Batch Processing

This chapter describes the batch processing that can be performed for RPO.

About RPO Batch Scripts

Batch scripts, which are configured externally, are required by RPO to create embedded forecasts. This section describes the batch scripts that may need to be configured during the RPO implementation to achieve your business needs.

The following batch scripts are available for RPO:

- Forecast Batch Scripts
- Price Optimization Batch Scripts

Forecast Batch Scripts

Through batch scripts, RPO can generate forecasts based on historical data that predict the future product demand. These forecasts, which are adjusted for promotional affects, are used by RPO to optimize permanent prices.

These forecasts are created at the source level of the historical data and then spread to the final level of optimization. Both the source level and the final level are defined in the RPAS Configuration Tools. The source level is the intersection of the product and location hierarchies. The final level can be the intersection at which the price optimization is performed, or it can be the intersection from which the forecast is aggregated to the optimization level. For example, the source level can be the category/price zone level, and the final level can be at item/price zone, a lower, more specific level. The forecast is spread from the source level to the final level through a dynamic profile that is automatically generated during the batch process.

The forecast parameters used in the forecast expression are instantiated at configuration time, but they can also be edited within the RPO application using the Measure Analysis workbook. For instructions on editing these measures within the Measure Analysis workbook, see the Defining the Forecast Start Date section.

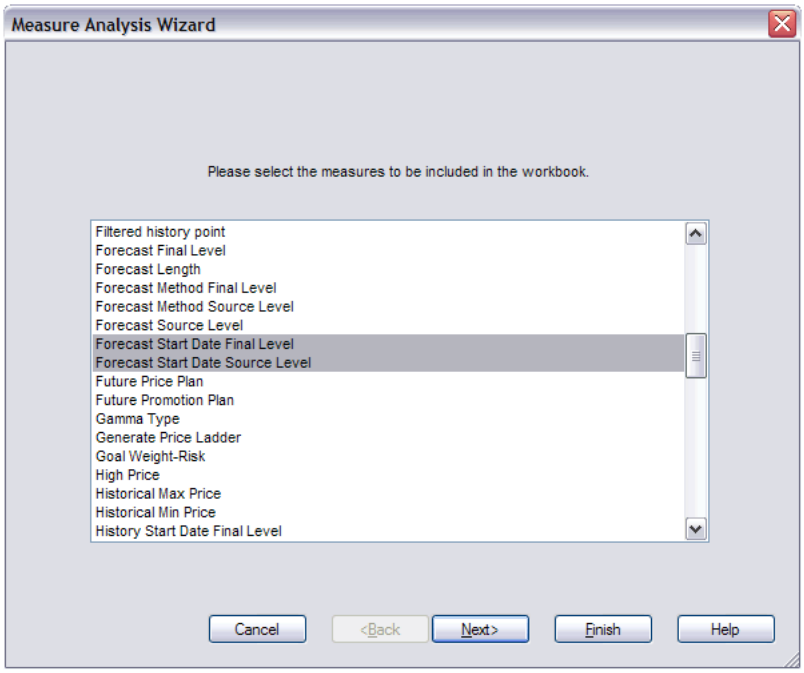
After forecasts are generated in the batch process, they are displayed in the Price Analysis workbook within the RPO application. This workbook displays details about the forecast, future prices, and promotion plan. For more information about this workbook, see the *Oracle Retail Regular Price Optimization User Guide for the RPAS Classic Client* or the *Oracle Retail Regular Price Optimization User Guide for the RPAS Fusion Client*.

Defining the Forecast Start Date

Before running the batch scripts, perform the following steps in the RPO application.

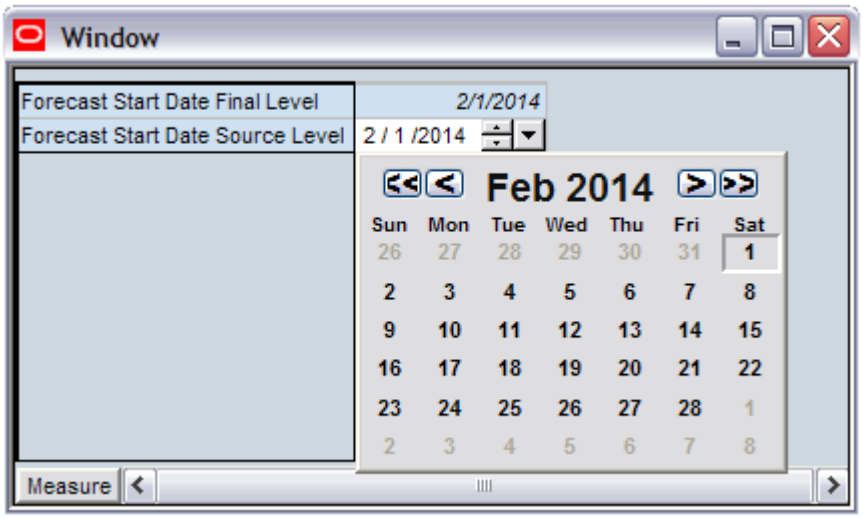
1. Build a new Measure Analysis workbook in the RPO application.
2. In the Measure Analysis Wizard, select **Forecast Start Date Final Level** and **Forecast Start Date Source Level**. Click **Finish**.

Figure 8–1 Measure Analysis Wizard



3. The Measure Analysis worksheet appears. In both measures, select the date that you want the forecast to begin. Both measures should contain the same date. Save and commit the workbook.

Figure 8–2 Measure Analysis Worksheet



Running the Forecasts Batch Process

Forecasts can be generated for the global domain or for individual local domains. These two options are described below.

Forecasts Run in a Local Domain

To generate a forecast for a local domain, run the rpo_forecast_localdomain.sh script in that local domain.

```
. . / rpo_forecast_localdomain.sh
```

This batch script generates a forecast for that local domain. Repeat this process for each additional local domain that needs a forecast.

Forecasts Run in the Global Domain

To generate a forecast for the global domain and hence all local domains, run the batch_forecast_run.sh batch script:

```
. . / batch_forecast_run.sh
```

This batch script calls the rpo_forecast_localdomain.sh script for every local domain within the global domain. Forecasts are generated for all domains.

Price Optimization Batch Scripts

In addition to running price optimization within the RPO application, users can run price optimization in batch. Running the optimization in batch during off hours rather than within the RPO application can save users time.

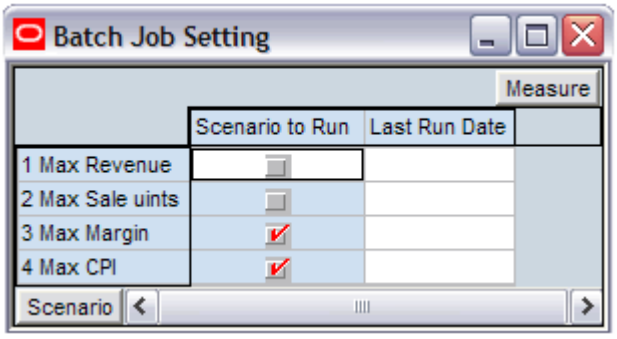
After users set up scenarios with goals and constraints in the Price Analysis workbook and commit that workbook to the database, they must select the scenarios to optimize and run the batch script. For instructions on selecting scenarios, see the Selecting Scenarios for Price Optimization section. For instructions on running the batch scripts, see the Running the Price Optimization Batch Process section.

Selecting Scenarios for Price Optimization

Before running the batch scripts, select the scenarios you want to optimize.

1. Build a Business Administration workbook in the RPO application.
2. In the Batch Job Setting worksheet, select the scenarios that you want to optimize. Save and commit the workbook.

Figure 8–3 Batch Job Setting Worksheet



Running the Price Optimization Batch Process

Price optimizations can be generated for the global domain or for individual local domains. These two options are described below.

Price Optimizations Run in a Local Domain

To run optimization for a local domain, run the rpo_batch_localdomain.sh script in that local domain.

```
. . / rpo_batch_localdomain.sh
```

This batch script completes an optimization run for that local domain. Repeat this process for each additional local domain that needs a forecast.

Price Optimizations Run in the Global Domain

To run optimization for the global domain and hence all local domains, run the batch_priceopt_run.sh batch script:

```
. . / batch_priceopt_run.sh
```

This batch script calls the rpo_batch_localdomain.sh script for every local domain within the global domain. Optimization runs are completed for all domains.

After the optimization batch process has been completed, users can review the optimized prices in the Price Analysis workbook in the RPO application.

Note: There is no autoapproval of scenarios that are run in batch. The current optimization batch is not run in parallel.

9

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.

Translation

Translation is the process of interpreting and adapting text from one language into another. Although the code itself is not translated, components of the application that are translated include the following:

- Graphical user interface (GUI)
- Error messages

The following components are not translated:

- Documentation (online help, release notes, installation guide, user guide, operations guide)
- Batch programs and messages
- Log files
- Configuration tools
- Reports
- Demonstration data
- Training materials

The user interface has been translated into 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 Administration Guide for the Classic Client* or the *Oracle Retail Predictive Server Administration Guide for the Fusion Client*.