

Oracle® Retail Merchandise Financial Planning

Operations Guide

Release 13.1.2.9

May 2010

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

Primary Author: Barrett Gaines

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Licensing Note: This media pack includes a Restricted Use license for Oracle Retail Predictive Application Server (RPAS) - Enterprise Engine to support Oracle® Retail Merchandise Financial Planning 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 software component known as **ACUMATE** developed and licensed by Lucent Technologies Inc. of Murray Hill, New Jersey, to Oracle and imbedded in the Oracle Retail Predictive Application Server - Enterprise Engine, Oracle Retail Category Management, Oracle Retail Item Planning, Oracle Retail Merchandise Financial Planning, Oracle Retail Advanced Inventory Planning, Oracle Retail Demand Forecasting, Oracle Retail Regular Price Optimization, Oracle Retail Size Profile Optimization, Oracle Retail Replenishment Optimization applications.
- (ii) 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.
- (iii) the **SeeBeyond** component developed and licensed by Sun Microsystems, Inc. (Sun) of Santa Clara, California, to Oracle and imbedded in the Oracle Retail Integration Bus application.
- (iv) the **Wavelink** component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.
- (v) the software component known as **Crystal Enterprise Professional and/or Crystal Reports Professional** licensed by SAP and imbedded in Oracle Retail Store Inventory Management.
- (vi) 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.
- (vii) 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.
- (viii) the software component known as **Style Report™** developed and licensed by InetSoft Technology Corp. of Piscataway, New Jersey, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration application.
- (ix) the software component known as **DataBeacon™** developed and licensed by Cognos Incorporated of Ottawa, Ontario, Canada, to Oracle and imbedded in the Oracle Retail Value Chain Collaboration 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

Send Us Your Comments	xi
Preface	xiii
Audience.....	xiii
Related Documentation.....	xiii
Customer Support.....	xiv
Review Patch Documentation	xiv
Oracle Retail Documentation on the Oracle Technology Network	xiv
Conventions	xv
1 Introduction	
Contents of this Guide	1-1
Overview of the Fashion Planning Bundle	1-1
Architecture of the Template Applications	1-2
Business Process Flow	1-3
Key Features of MFP	1-3
Skills Needed for Implementation	1-4
Applications.....	1-4
Technical Concepts	1-4
2 Implementation Considerations	
Historical Data	2-1
Sizing Impacts	2-1
Partitioning	2-2
Formatting	2-2
Plug-ins	2-2
Patch Considerations	2-3
Batch Scheduling	2-3
Security	2-3
Internationalization	2-4

3 Build Scripts

Installation Dependencies	3-1
Environment Setup	3-1
RPAS Installation	3-1
MFP Installation	3-2
Custom Domain Build.....	3-2
Handling Common Hierarchy Files in the Fashion Planning Bundle Applications.....	3-2
Building the MFP RPAS Domain	3-3
Batch Designs.....	3-4
Building a Domain.....	3-4
Patching a Domain.....	3-5
Creating Users and User Groups	3-5
Loading and Extracting Data	3-6

4 Fashion Planning Bundle and ODI Integration

Overview of the Fashion Planning Bundle	4-1
Integration Interface Data Flow Description.....	4-2
From Size Profile Optimization to Assortment Planning	4-2
From Assortment Planning to Item Planning and an Allocation Application	4-2
From Item Planning to Merchandise Financial Planning	4-2
From Merchandise Financial Planning to Item Planning	4-2
From Merchandise Financial Planning to Assortment Planning.....	4-3
Execution Applications	4-4
Optimization Applications	4-4
Planning Applications.....	4-4
ODI Enabled Integration	4-5
Measure Data Integration	4-5
MFPCost to IP Package	4-6
Scripts Used By the MFPCost to IP Package.....	4-6
Data Mapping for MFPCost to IP Package	4-6
IP to MFPCost Package	4-7
Scripts Used By the IP to MFPCost Package.....	4-7
Data Mapping for IP to MFPCost Package	4-7
MFPRetail to IP Package	4-8
Scripts Used By the MFPRetail to IP Package.....	4-8
Data Mapping for MFPRetail to IP Package	4-8
IP to MFPRetail Package	4-9
Scripts Used By the IP to MFPRetail Package.....	4-9
Data Mapping for IP to MFPRetail Package	4-9
MFPCost to AP Package.....	4-10
Scripts Used By the MFPCost to AP Package.....	4-10
Data Mapping for MFPCost to AP Package.....	4-10
MFPRetail to AP Package	4-11
Scripts Used By the MFPRetail to AP Package.....	4-11
Data Mapping for MFPRetail to AP Package	4-11
SizeOpt to AP Package	4-11
Scripts Used By the SizeOpt to AP Package	4-11

Data Mapping for SizeOpt to AP Package	4-11
AP to IP Package	4-12
Scripts Used By the AP to IP Package.....	4-12
Data Mapping for AP to IP Package	4-12
Hierarchy Integration with ODI.....	4-13
Adding New Products: PROD Hierarchy Integration.....	4-13
Adding New Stores: LOC Hierarchy Integration.....	4-15
5 Script Integration	
Integration Scripts.....	5-1
Batch Designs.....	5-2
Load Actuals.....	5-2
Export to IP	5-4
Export to AP	5-7
6 MFP - RMS Integration	
Overview of the MFP Data Flow.....	6-1
Integration Interface Data Flow Description.....	6-2
From RMS to MFP	6-2
From a Data Warehouse Application to MFP.....	6-2
From MFP to a Markdown Optimization Engine	6-2
RMS to MFP Transformation	6-3
Script Installation	6-3
Environment Setup	6-3
RMS Integration Script Details.....	6-5
Markdown Optimization Integration	6-6
7 Batch Processing	
Batch Structure Overview.....	7-1
Batch Scheduling.....	7-1
Daily Batch Scripts	7-2
Weekly Batch Scripts	7-2
Unscheduled Administration Script.....	7-3
Batch Environment Scripts	7-3
Batch Designs.....	7-4
Calendar Hierarchy Load	7-5
Product Hierarchy Load.....	7-6
Location Hierarchy Load	7-7
Load On Order Data	7-8
Load Actuals Data.....	7-9
Load IP Data	7-12
Process Actuals Data.....	7-14
Run Batch Forecast.....	7-15
Refresh Workbooks.....	7-16
Auto Build Workbooks.....	7-17
Load Mapping Measures	7-18

List of Figures

1-1	Architecture of the RPAS Template Applications.....	1-2
1-2	Business Process Workflow	1-3
4-1	Conceptual Overview.....	4-1
4-2	Overview of Fashion Planning Bundle Integration	4-3
6-1	Data Flow for the Integration of MFP with Other Applications.....	6-1

List of Tables

3-1	Build Scripts.....	3-3
3-2	Building a Domain Error Information	3-4
3-3	Patching a Domain Error Information.....	3-5
4-1	MFP Cost to IP: CP Approved Data.....	4-6
4-2	MFP Cost to IP: OP Approved Data.....	4-6
4-3	IP to MFPCost: CP Approved Data.....	4-7
4-4	IP to MFPCost: OP Approved Data.....	4-7
4-5	MFPRetail to IP: CP Approved Data.....	4-8
4-6	MFPRetail to IP: OP Approved Data	4-8
4-7	IP to MFPRetail: CP Approved Data.....	4-9
4-8	IP to MFPRetail: OP Approved Data	4-10
4-9	MFPCost to AP Data.....	4-10
4-10	MFPRetail to AP Data	4-11
4-11	SizeOpt to AP Data	4-11
4-12	AP to IP Data	4-12
4-13	Prod.csv File Dimension Positions/Labels	4-13
4-14	Loc.csv File Dimension Positions/Labels	4-15
5-1	Integration Script	5-2
5-2	Load Actuals Error Information	5-4
5-3	Export to IP Error Information.....	5-6
5-4	Export to AP Error Information.....	5-7
7-1	Directories Used by Batch Scripts.....	7-1
7-2	Daily Batch Scripts.....	7-2
7-3	Weekly Batch Scripts	7-2
7-4	Unscheduled Administration Script	7-3
7-5	Calendar Hierarchy Load Error Information.....	7-5
7-6	Product Hierarchy Load Error Information.....	7-6
7-7	Location Hierarchy Load Error Information	7-7
7-8	Load Actuals Error Information	7-8
7-9	Load Actuals Data Error Information.....	7-11
7-10	Load Actuals Data Error Information.....	7-13
7-11	Process Actuals Error Information	7-14
7-12	Run Batch Forecast Error Information.....	7-15
7-13	Refresh Workbooks Error Information.....	7-16
7-14	Auto Build Workbooks Error Information.....	7-17
7-15	Load Actuals Error Information	7-18

Send Us Your Comments

Oracle Retail Merchandise Financial Planning Operations Guide, Release 13.1.2.9

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

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

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

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

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Applications Release Online Documentation CD available on My Oracle Support and <http://www.oracle.com>. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: 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

Oracle Retail Operations Guides are designed so that you can view and understand the application's 'behind-the-scenes' processing, including such information as the following:

- Key system administration configuration settings
- Technical architecture
- Functional integration dataflow across the enterprise
- Batch processing

Audience

Anyone who has an interest in better understanding the inner workings of Merchandise Financial Planning can find valuable information in this guide. There are three audiences in general for whom this guide is written:

- System analysts and system operation personnel:
 - who are looking for information about Merchandise Financial Planning processes internally or in relation to the systems across the enterprise.
 - who operate Merchandise Financial Planning on a regular basis.
- Integrators and implementation staff who have the overall responsibility for implementing Merchandise Financial Planning into their enterprise.
- Business analysts who are looking for information about processes and interfaces to validate the support for business scenarios within Merchandise Financial Planning and other systems across the enterprise.

Related Documentation

For more information, see the following documents in the Oracle Retail Merchandise Financial Planning Release 13.1.2.9 documentation set:

- *Oracle Retail Merchandise Financial Planning Release Notes*
- *Oracle Retail Merchandise Financial Planning Retail Installation Guide*
- *Oracle Retail Merchandise Financial Planning Cost Installation Guide*

For more information about the Fashion Planning Bundle applications see the following documentation sets:

- Oracle Retail Item Planning documentation
- Oracle Retail Item Planning Configured for COE documentation
- Oracle Retail Clearance Optimization Engine documentation
- Oracle Retail Assortment Planning documentation
- Oracle Retail Size Profile Optimization documentation

For more information about RPAS and the ODI Enabled Integration, see the following documentation sets:

- Oracle Retail ODI Enabled Integration documentation
- Oracle Data Integrator documentation
- Oracle Retail Predictive Application Server documentation

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, 13.1) or a later patch release (for example, 13.1.2). 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.
<i>italic</i>	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.

Introduction

Oracle Retail Merchandise Financial Planning (MFP) provides strategic and financial product planning functions. These functions support industry planning standards for pre-season and in-season processes. For a more detailed overview of the functionality within MFP, see the *Oracle Retail Merchandise Financial Planning Cost User Guide* and *Oracle Retail Merchandise Financial Planning Retail User Guide*.

Contents of this Guide

This implementation guide addresses the following topics:

- [Chapter 1, "Introduction"](#). Overview of the MFP 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 MFP RPAS domain.
- [Chapter 4, "Fashion Planning Bundle and ODI Integration"](#). Overview of the Fashion Planning Bundle data flow and the ODI enabled integration for the bundle.
- [Chapter 5, "Script Integration"](#). Overview of the script integration used for the Fashion Planning Bundle.
- [Chapter 6, "MFP - RMS Integration"](#). Overview of RMS integration as well as integration with data warehouses and markdown optimization applications.
- [Chapter 7, "Batch Processing"](#). Explanation of batch scheduling and batch designs.

Overview of the Fashion Planning Bundle

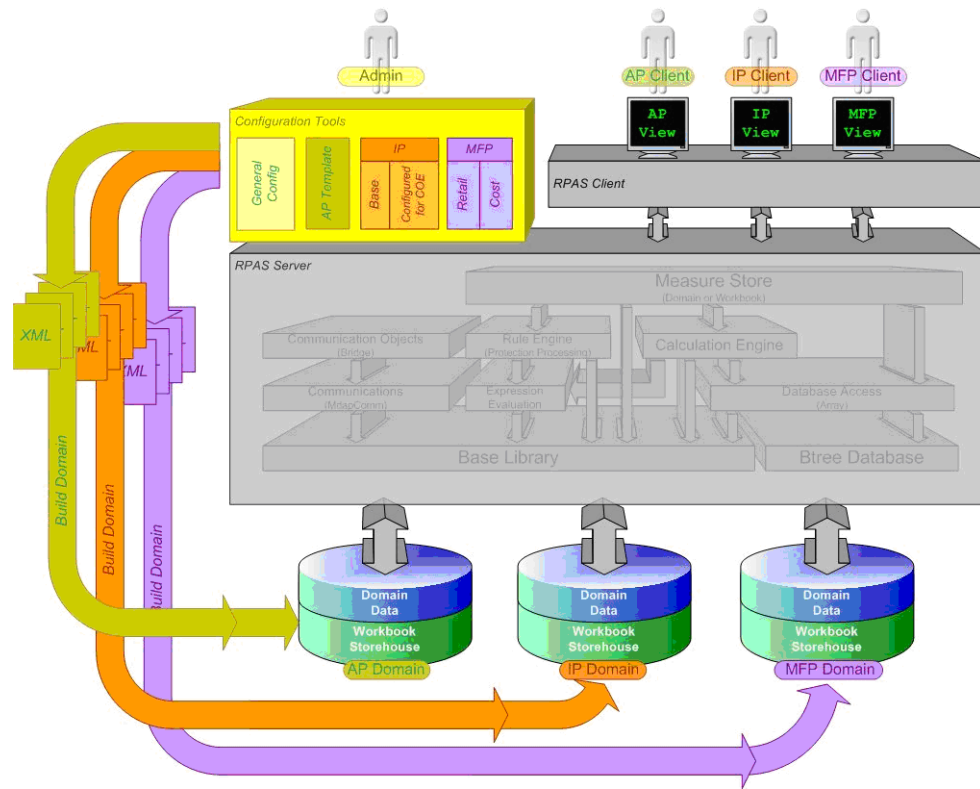
The Fashion Planning Bundle is the integration of Item Planning (IP), Clearance Optimization Engine (COE), Assortment Planning (AP), Merchandise Financial Planning (MFP), and Size Profile Optimization (SPO) as a full-suite planning solution for fashion retailers.

For details on the integration, see [Chapter 4](#) and [Chapter 5](#).

Architecture of the Template Applications

Figure 1-1 shows the architecture of the Oracle Retail Predictive Application Server (RPAS) template applications.

Figure 1-1 Architecture of the RPAS Template Applications



This diagram describes the RPAS template applications. In the truest sense, these templates 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. The templates are pre-defined means by which 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 pre-defined attributes:

- Measures
- Special expressions
- Rules
- Workbook layout

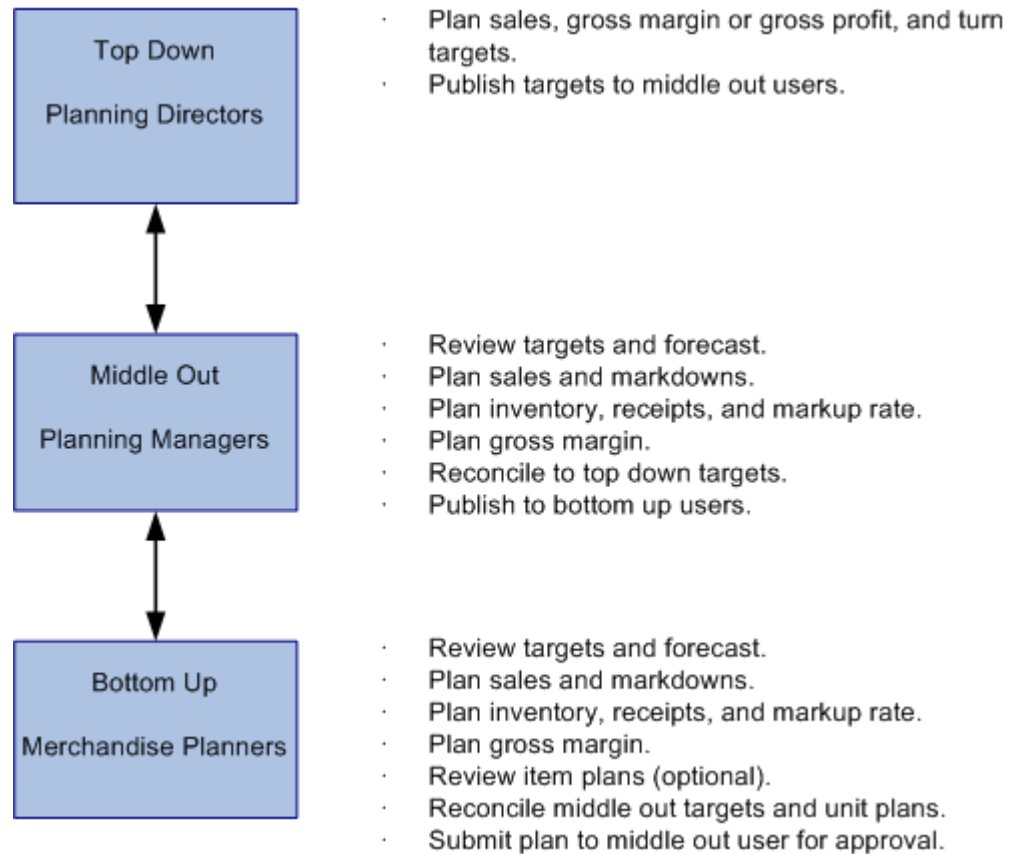
A client requests to use one of the templates via the Configuration Tools. A number of XML files are then output which define the domain to be created. These XML files are used to build the specified domain incorporating all the attributes mentioned above that have been defined specifically.

Once the domain has been created, the end-user is able to access the domain data through the RPAS client. Based on the template used, workbooks, measures, rules, and so on are available to the end user.

Business Process Flow

Figure 1–2 shows a typical workflow for MFP.

Figure 1–2 Business Process Workflow



Key Features of MFP

MFP provides the following features:

- Setting and passing targets
- Creating a financial plan in a structured method
- Plan reconciliation
- Plan approval
- Plan maintenance
- Replanning
- Plan monitoring

Skills Needed for Implementation

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

Applications

The implementer should understand interface requirements of the integrated applications and data sources for the master data, demand, and inventory history. For the Fashion Planning Bundle, the implementer needs this knowledge for the following applications:

- Oracle Retail Predictive Application Server
- Oracle Retail Assortment Planning
- Oracle Retail Item Planning
- Oracle Retail Clearance Optimization Engine
- Oracle Retail Merchandise Financial Planning
- Oracle Retail Size Profile Optimization
- Oracle Retail Merchandising System (RMS)
- Oracle Retail Markdown Optimization

Technical Concepts

The implementer should understand the following technical concepts:

- UNIX system administration, shell scripts, and job scheduling
- Performance constraints based on the retailer's infrastructure
- Technical architecture for MFP
- Retailer's hierarchical (SKU/store/day) data
- MFP batch processes
- How to set up an RPAS domain
- A basic understanding of RPAS configuration and how to use the RPAS Configuration Tools
- Understanding of how RPAS rule language works
- Understanding of measures and dimension constructs

Implementation Considerations

The following information needs to be considered before configuring MFP:

- [Historical Data](#)
- [Sizing Impacts](#)
- [Partitioning](#)
- [Formatting](#)
- [Plug-ins](#)
- [Patch Considerations](#)
- [Batch Scheduling](#)
- [Security](#)
- [Internationalization](#)

Historical Data

It is recommended that you have at least two years of historical sales and inventory data for creating MFP targets. Less data can be used, but the more data that is available, the more statistical significance can be given to the MFP targets.

It is also important to have two years of history to produce an optimal forecast. Both MFP Cost and MFP Retail use the embedded forecast.

Sizing Impacts

The following factors can affect size requirements:

- **Style-Color**—number of style-colors. A style-color is the combination of the style information, such as fleece jacket, and color, such as, navy. Although the lowest level of planning within MFP is SubClass, data is loaded at the style-color level.
- **Store**—number of physical, Web, and other distinct retail outlets. Although store is not the lowest level of planning, data may be loaded at that level.
- **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.

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 MFP domain is defined as a global domain. For performance reasons, a single domain is not recommended. There should be an even distribution of users across a set of local domains. For example, men's merchandise could be in a domain, women's merchandise in a domain, and children's merchandise in a domain. When a user is committing data in the men's merchandise domain, this will not affect the users in the women's or children's 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?

Note: The partitioning level in the MFP configuration is Group. It is recommended that this not be changed.

Formatting

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

- In the configuration, 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. Formatting can only be changed by using the RPAS Configuration Tools. For more information, see the *Oracle Retail Predictive Application Server Configuration Tools User Guide*.
- To set up worksheet sizes and placements, exception value formatting, gridlines, and other formatting, once the domain is built, the implementer or a user can instantiate a workbook of the template to set up specific formatting and use the Format menu. The implementer can then save the format to the template so that it is available to all users for any newly created workbooks. For information on how to use the Format menu, see the *Oracle Retail Predictive Application Server User Guide*.

Plug-ins

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

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 MFP RPAS 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 straightforward process.
- Changes to the configuration
These types of changes can be more complex. If a retailer has customizations in the configuration, the customizations must be redone on the new configuration before the patch is installed.

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. Batch scripts are used for importing and exporting data and for generating targets. The retailer needs to 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 re-create targets needs to be determined.

- You have to 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 coming from other Fashion Bundle application be loaded daily.
- Product availability and seasonal changes can be reasons for recalculating the targets. It can also be triggered by the addition of new products and availability of substantial new sales and inventory history.

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 is not able to edit any values or commit the workbook. The read-only workbook can be refreshed.

When users save a workbook, they assign one of three access permissions to the workbook:

- 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*.

Internationalization

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 may include the following, among others:

- 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
- Demo data
- Training Materials

The RPAS 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 RPAS server and client, as well as strings from the solution extensions.

Translations of the solution templates, including MFP, are not released. Since the labels for measures, hierarchies, dimensions, and workbook templates, are typically changed by the retailer at implementation time, languages for the templates are not released. All templates have the ability to support multibyte characters.

Build Scripts

This chapter describes the scripts that must be executed to build the MFP RPAS domain.

Installation Dependencies

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

- For information on installing RPAS, see the *Oracle Retail Predictive Application Server Installation Guide*.
- For information on installing MFP, see the *Oracle Retail Merchandise Financial Planning Retail Installation Guide* and the *Oracle Retail Merchandise Financial Planning Cost 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 MFP_HOME. The UNIX user performing the installation needs to set up an environmental variable called MFP_HOME in the user's profile:

```
export MFP_HOME=<full path name to MFP home>
```

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

MFP Installation

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

The MFP installer performs the following functions:

- Downloads the configuration and batch scripts into the `$MFP_HOME/config` and `$MFP_HOME/bin` directories
- Downloads a set of sample hierarchy and data files into the `$MFP_HOME/input` directory
- Builds a sample domain at:
 - For MFP Cost, `$MFP_HOME/domain/mfpcst`
 - For MFP Retail, `$MFP_HOME/domain/mfpRTL`

Custom Domain Build

To do a custom build of a domain:

1. Change to the configuration directory: `cd $MFP_HOME/config`
2. Update the `globaldomainconfig.xml` file with the correct domain paths. In addition, the groups that were part of the sample installation are replaced with the position information for the groups that are part of the retailer's hierarchy.
3. Change to the bin directory: `cd $MFP_HOME/bin`
4. Update the `localdomainlist.cfg` file with the correct paths for the local domains.
5. If needed, update the default environment variables in `environment.ksh`.
6. Execute the `build_mfp.ksh` script: `./build_mfp.ksh`
7. It is expected that the first time `build_mfp.ksh` is executed, an error occurs when it tries to remove the old log file because a log file does not yet exist.

Handling Common Hierarchy Files in the Fashion Planning Bundle Applications

The following hierarchy files contain the superset of all the dimensions along the product, location, and calendar hierarchies:

- `prod.hdr.csv.dat`
- `loc.hdr.csv.dat`
- `clnd.hdr.csv.dat`

These common hierarchy files are shared among AP, IP, MFP, and SPO.

Note: The common hierarchy is not available for IP Configured for COE.

Each `hdr.csv.dat` (`hdr`) hierarchy file contains a header line that lists all the dimensions for which position information is contained in the file. The RPAS build process handles these `hdr` files, so that every application extracts the position information relevant to itself and ignores dimensions not configured in the application.

The filterHier utility is run on the hdr files to convert them into standard hierarchy files that are then passed to loadHier. The build process, which uses rpaInstall, can differentiate between standard and hdr hierarchy files. There is no need for the implementer to make any changes in the domain build process.

If using hdr files, the implementer needs to run filterHier before running loadHier. The filterHier utility converts the hdr files into standard hierarchy files that can be processed by loadHier. Note that there is no need to run filterHier if the standard hierarchy files are already available.

Note: The hdr files must reside outside the domain input directory before running filterHier. By default, the filterHier utility puts the newly created filtered hierarchy files into the input folder of the domain.

See the *Oracle Retail Predictive Application Server Administration Guide* for details on the RPAS utilities.

Building the MFP RPAS Domain

The following scripts are not part of a normal batch schedule but are executed only to build or patch a domain.

[Table 3–1](#) lists the build scripts. These scripts are located in the \$MFP_HOME/bin directory.

Table 3–1 Build Scripts

Description	Script Name	Dependency
Build a domain	build_mfp.ksh	None
Patch a domain when the configuration changes do not affect formatting	patch_mfp_keepformats.ksh	None
Patch a domain when the configuration changes require updates to formatting	patch_mfp_deleteformats.ksh	None

Batch Designs

This section contains detailed information on the following build scripts:

- [Building a Domain](#)
- [Patching a Domain](#)

Building a Domain

Script

build_mfp.ksh

Usage

build_mfp.ksh

Error Information

Table 3–2 Building a Domain Error Information

Task Name	Error Code	Abort Required?	Description of Error
build_mfp	-1	yes	Error during domain build.
build_mfp	-2	yes	Error when loading users.
build_mfp	-3	yes	Error when loading measures.
build_mfp	-4	yes	Error in the master domain calculations.
build_mfp	-5	yes	localdomainlist.cfg file is not found.
build_mfp	-6	yes	Error in local domain calculations.
build_mfp	-7	yes	Error when disabling commit later and insert measure functionality.
build_mfp	-8	yes	Error when running copyDomain to set local domain paths to relative paths.

Notes

- The script uses the Configuration Tools `rpasInstall` utility to build a domain. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script also uses the following RPAS utilities: `usermgr`, `mace`, `loadmeasure`, `copyDomain`, and `domainprop`. See the *Oracle Retail Predictive Application Server Administration Guide* for details on these utilities.
- All hierarchy and measure files need to be placed in `$MFP_HOME/input` before the execution of this script.
- The commit later and insert measure functionality is disabled.

Patching a Domain

When changes have been made to MFP that require changes to the domain, the following scripts are used to apply the patch to the domain.

Script

```
patch_mfp_keepformats.ksh
or
patch_mfp_deleteformats.ksh
```

Usage

```
patch_mfp_keepformats.ksh
patch_mfp_deleteformats.ksh
```

Error Information

Table 3–3 Patching a Domain Error Information

Task Name	Error Code	Abort Required?	Description of Error
patch_mfp_deleteformats	-10	yes	Missing configuration file mfp_subdomainlist.cfg.
patch_mfp_deleteformats	-11	yes	Error during domain patch.
patch_mfp_keepformats	-20	yes	Error during domain patch.

Notes

- Prior to RPAS 13.1.1, formats were not transferrable across major versions, and users had to delete formats before patching and then reapply the formats after the patch was installed. RPAS 13.1.1 included an upgrade to the save format process that allows RPAS applications to keep the formats when updating versions. Because this update is not backwards compatible, the script you use depends on the version of RPAS you are upgrading from:
 - If you are upgrading from an RPAS version prior to 13.1.1, use patch_ap_deleteformats. This script removes all existed saved formats (template, group, and user). Once you have installed the patch, reapply your formats.
 - If you are upgrading from RPAS 13.1.1 or later, use patch_ap_keepformats. The formats are automatically converted.
- The scripts utilize the Configuration Tools rpassInstall utility to build a domain. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.

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*.

Loading and Extracting Data

Data is loaded into MFP using the standard RPAS approach. See the *Oracle Retail Predictive Application Server Administration Guide* for details on formatting the load data files and on the utilities that enable administrators to load data into RPAS. If you are using the ODI enabled integration, see [Chapter 4](#) for more information on data integration among Fashion Planning Bundle applications. If you are using script integration for the bundle, see [Chapter 5](#). For information about RMS integration, see [Chapter 6](#). For information on batch scripts, see [Chapter 7](#).

Fashion Planning Bundle and ODI Integration

This chapter describes the overall flow of data among the Fashion Planning Bundle applications, the integration among the Fashion Planning Bundle applications using Oracle Data Integrator (ODI), and the integration of hierarchies using ODI.

For information about script integration for the Fashion Planning Bundle, see [Chapter 5](#). For information about RMS integration, see [Chapter 6](#).

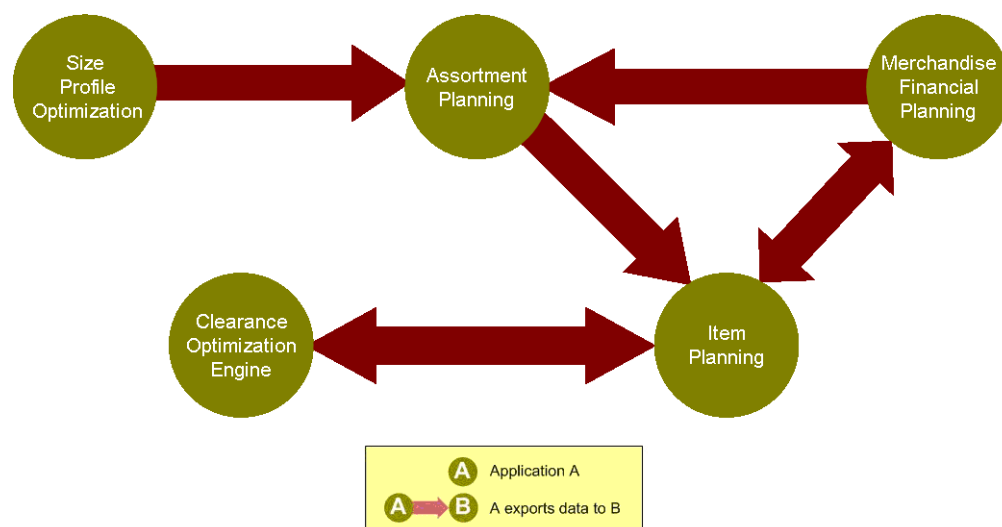
Overview of the Fashion Planning Bundle

This section describes the integration between solutions within the Fashion Planning Bundle. It does not describe external integrations.

The Fashion Planning Bundle is a full-suite planning solution for fashion retailers that integrates the following: Item Planning (IP), Clearance Optimization Engine (COE), Assortment Planning (AP), Merchandise Financial Planning (MFP), and Size Profile Optimization (SPO).

[Figure 4-1](#) shows the conceptual overview of the integration of these products.

Figure 4-1 Conceptual Overview



This solution supports data sharing among these applications. Note that the data sharing functionality is not dependent on the presence of all these applications. The defined data sharing between any of the applications works for the entire suite as well as for a subset of the applications.

Integration Interface Data Flow Description

These descriptions explain each of the data flows in [Figure 4–1](#).

From Size Profile Optimization to Assortment Planning

The following data is imported from SPO to AP:

- Store-level size profiles (current and archived versions) with corresponding escalation levels
- Store-clustered size profiles
- Pre-pack details: labels, descriptions, and configurations

The size profiles are by AP to obtain more detailed buy plans at style-color-size or pre-pack per store per receipt week. This enables the product to be tailored to the consumer selling patterns at each specific location.

SPO can group stores based on historical data into clusters, and these clusters can be used as a basis for creating Pre-Packs.

For more information on SPO, see the Oracle Retail Size Profile Optimization documentation.

From Assortment Planning to Item Planning and an Allocation Application

The following data is exported from AP to IP and an allocation application:

- Buy plan

AP sends the buy plan to IP and the allocation application. The allocation application can use the sales plan or receipt plan created by AP to determine what inventory is allocated to stores. IP uses the pre-buying period assortment plan to track the performance of items during the buying period and to provide a framework to respond accordingly.

From Item Planning to Merchandise Financial Planning

The following data is exported from IP to MFP:

- Item plan

Approved data is exported from IP and loaded into MFP at the subclass level. The aggregation takes place within IP when exporting. Only the plans that have been approved since the last export in IP are imported into MFP.

MFP users can review and reconcile their merchandise financial plans (which are planned at the subclass level) to the approved item plans from IP.

From Merchandise Financial Planning to Item Planning

The following data is exported from MFP to IP:

- MFP current and original plans

The current and original plans are used by IP to help with the creation of item plans.

From Merchandise Financial Planning to Assortment Planning

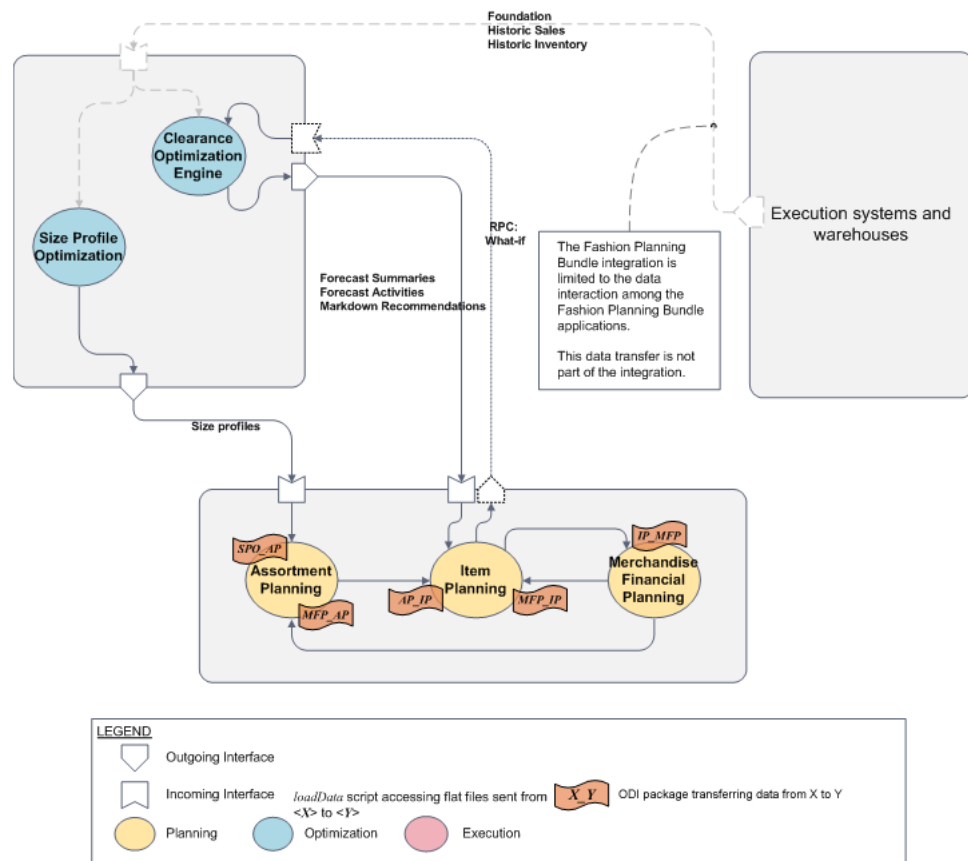
The following data is exported from MFP to AP:

- MFP current plans

MFP current plans are critical inputs into the AP process. They define the financial goals for that buying period. AP uses these plans as targets to determine the number of options it should carry, the number of weeks it should carry them, and the amount of markdown it can afford in order to meet the targets for sales and gross margin.

Figure 4–2 shows the applications and data flow that are part of the Fashion Planning Bundle.

Figure 4–2 Overview of Fashion Planning Bundle Integration



The applications shown in the diagram are in the following categories:

- Execution Applications
- Optimization Applications
- Planning Applications

Execution Applications

Note: Full data transfers from the execution applications are not part of the Fashion Planning Bundle integration. They are included in the diagram to illustrate that historical data needs to be obtained from applications outside the applications included in the Fashion Planning Bundle.

The execution applications provide the foundation and historical data that is input to the optimization applications.

- A data warehouse application provides the initial load of sales and inventory data.
- A merchandising application provides the daily and weekly data updates, or deltas, of the sales and inventory data. It provides data updates of master information such as styles, prices, stores, and so on.

Optimization Applications

The optimization applications take the foundation and historical data and produce size and markdown recommendations that are used by the planning applications.

- SPO creates profiles of the optimal size distribution by both merchandise category and store. This gives insight into consumer demand patterns by size. These size profiles feed into AP.

For more information on SPO, see the Oracle Retail Size Profile Optimization documentation. For more information on AP, see the Oracle Retail Assortment Planning documentation.

- COE provides markdown recommendations and forecasts that enable retailers to make informed markdown decisions. What-if data feeds back into COE from IP. COE is only available with IP configured for COE.

For more information on COE, see the Oracle Retail Clearance Optimization Engine documentation. For more information on IP, see the Oracle Retail Item Planning documentation.

Planning Applications

The planning applications take data from the optimization applications and create plans for the retailer.

- AP creates buy plans for the retailer based on size profiles from SPO and the subclass plans from MFP. The buy plans feeds into IP.
- IP takes the buy plan from AP and the subclass plan from MFP to create an item plan. When configured for COE, markdown data from COE is also used. IP output feeds back into MFP. What-if data feeds back into COE to create new markdown plans.
- MFP takes data from IP which is consulted when creating the subclass plan in MFP. Based on financial goals, the subclass plans determine how much merchandise should be available in a store in order to meet the goals. Subclass plans are fed into AP and IP.

ODI Enabled Integration

Oracle Retail ODI Enabled Integration for the Fashion Planning Bundle is a set of packages that allows users to easily execute data transfers between Fashion Planning Bundle applications.

Oracle Retail ODI Enabled Integration leverages Oracle Data Integrator (ODI) to store information about data interfaces among Fashion Planning Bundle 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.

The Fashion Planning Bundle applications supported by the ODI enabled integration include Merchandise Financial Planning Retail (MFP Retail), Merchandise Financial Planning Cost (MFP Cost), Item Planning (IP), Item Planning Configured for COE (IP COE), Assortment Planning (AP), and Size Profile Optimization (SPO).

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

Measure Data Integration

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

- [MFPCost to IP Package](#)
- [IP to MFPCost Package](#)
- [MFPRetail to IP Package](#)
- [IP to MFPRetail Package](#)
- [MFPCost to AP Package](#)
- [MFPRetail to AP Package](#)
- [SizeOpt to AP Package](#)
- [AP to IP Package](#)

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

`$RPAS_HOME/scripts/integration/ODI`

MFP Cost to IP Package

The following information is about the MFP Cost to IP integration.

Scripts Used By the MFP Cost to IP Package

- mfp_to_ip_init.ksh
- mfp_to_ip_finalize.ksh

Data Mapping for MFP Cost to IP Package

The CP Approved data is sent when mowpappcpnewb is set to TRUE.

Table 4-1 MFP Cost to IP: CP Approved Data

MFP Cost Formula	IP Target Measure
bucpbopc	bucpbopc
bucpbopu	bucpbopu
bucpeopc	bucpeopc
bucpeopu	bucpeopu
bucprecc	bucprecc
bucprecu	bucprecu
bucpgmpv	bucpgmpv
bucpslsc	bucpslsc
bucpslsr	bucpslsr
bucpslsu	bucpslsu

The OP Approved data is sent when mowpappopnewb is set to TRUE.

Table 4-2 MFP Cost to IP: OP Approved Data

MFP Cost Formula	IP Target Measure
buopbopc	buopbopc
buopbopu	buopbopu
buopeopc	buopeopc
buopeopu	buopeopu
buoprecc	buoprecc
buoprecu	buoprecu
buopgmpv	buopgmpv
buopslsc	buopslsc
buopslsr	buopslsr
buopslsu	buopslsu

IP to MFPCost Package

The following information is about the IP to MFPCost integration.

Scripts Used By the IP to MFPCost Package

- ip_to_mfp_init.ksh
- ip_to_mfp_finalize.ksh

Data Mapping for IP to MFPCost Package

The CP Approved data is sent when ipcpappnewb is set to TRUE.

Table 4–3 IP to MFPCost: CP Approved Data

IP Formula	MFP Cost Target Measure
ipcpbopc	ipcpbopc
ipcpbopu	ipcpbopu
ipcpeopc	ipcpeopc
ipcpeopu	ipcpeopu
ipcpgmpv	ipcpgmpv
ipcprecc	ipcprecc
ipcprecu	ipcprecu
ipcpslsr	ipcpslsr
ipcpslsu	ipcpslsu
ipcpslsc	ipcpslsc

The OP Approved data is sent when ipopappnewb is set to TRUE.

Table 4–4 IP to MFPCost: OP Approved Data

IP Formula	Target MFP Cost Measure
ipopbopc	ipopbopc
ipopbopu	ipopbopu
ipopeopc	ipopeopc
ipopeopu	ipopeopu
ipopgmpv	ipopgmpv
ipoprecc	ipoprecc
ipoprecu	ipoprecu
ipopslsr	ipopslsr
ipopslsu	ipopslsu
ipopslsc	ipopslsc

MFPRetail to IP Package

The following information is about the MFPRetail to IP integration.

Scripts Used By the MFPRetail to IP Package

- mfp_to_ip_init.ksh
- mfp_to_ip_finalize.ksh

Data Mapping for MFPRetail to IP Package

The CP Approved data is sent when mowpappcpnewb is set to TRUE.

Table 4–5 MFPRetail to IP: CP Approved Data

MFP Retail Formula	IP Target Measure
bucpbopc	bucpbopc
bucpbopu	bucpbopu
bucpeopc	bucpeopc
bucpeopu	bucpeopu
bucprecc	bucprecc
bucprecu	bucprecu
bucpbopr	bucpbopr
bucpicogsc	bucpcogsc
bucpeopr	bucpeopr
bucpigmpv	bucpgmpv
bucpmkdpromor + bucpmkdpermr + bucpmkdclrr	bucpmkdr
bucprecr	bucprecr
bucpslsregr + bucpslsclrr + bucpslspror	bucpslsr
bucpslsregu + bucpslsclru + bucpslsprou	bucpslsu

The OP Approved data is sent when mowpappopnewb is set to TRUE.

Table 4–6 MFPRetail to IP: OP Approved Data

MFP Retail Formula	IP Target Measure
buopbopc	buopbopc
buopbopu	buopbopu
buopeopc	buopeopc
buopeopu	buopeopu
buoprecc	buoprecc
buoprecu	buoprecu
buopbopr	buopbopr

Table 4–6 (Cont.) MFPRetail to IP: OP Approved Data

MFP Retail Formula	IP Target Measure
buopicogsc	buopcogsc
buocpeopr	buocpeopr
buopigmpv	buopgmpv
buopmkdpromor + buopmkdpermr + buopmkdclrr	buopmkdr
buoprecr	buoprecr
buopslsregr + buopslsclrr + buopslspror	buopslsr
buopslsregu + buopslsclru + buopslsprou	buopslsu

IP to MFPRetail Package

The following information is about the IP to MFPRetail integration.

Scripts Used By the IP to MFPRetail Package

- ip_to_mfp_init.ksh
- ip_to_mfp_finalize.ksh

Data Mapping for IP to MFPRetail Package

The CP Approved data is sent when ipcpappnewb is set to TRUE.

Table 4–7 IP to MFPRetail: CP Approved Data

IP Formula	MFP Retail Target Measure
ipcpbopc	ipcpbopc
ipcpbopu	ipcpbopu
ipcpeopc	ipcpeopc
ipcpeopu	ipcpeopu
ipcpgmpv	ipcpgmpv
ipcprecc	ipcprecc
ipcprecu	ipcprecu
ipcpslsr	ipcpslsr
ipcpslsu	ipcpslsu
ipcpbopr	ipcpbopr
ipcpeopr	ipcpeopr
ipcprecr	ipcprecr

The OP Approved data is sent when ipopappnewb is set to TRUE.

Table 4–8 IP to MFPRetail: OP Approved Data

IP Formula	MFP Retail Target Measure
ipopbopc	ipopbopc
ipopbopu	ipopbopu
ipopeopc	ipopeopc
ipopeopu	ipopeopu
ipopgmpv	ipopgmpv
ipoprecc	ipoprecc
ipoprecu	ipoprecu
ipopslsr	ipopslsr
ipopslsu	ipopslsu
ipopbopr	ipopbopr
ipopeopr	ipopeopr
ipoprecr	ipoprecr

MFPCost to AP Package

The following information is about the MFPCost to AP integration.

Scripts Used By the MFPCost to AP Package

- mfp_to_ap_init.ksh
- mfp_to_ap_finalize.ksh

Data Mapping for MFPCost to AP Package

Data is sent when mowpappcpnewb is set to TRUE.

Table 4–9 MFPCost to AP Data

MFP Cost Formula	AP Target Measure
bucpgmpv	mfcpposgmr
bucprecc	mfcprecc
bucprecu	mfcprecu
bucpslsu	mfcpslsu
bucpslsr	mfcpslsr

MFPRetail to AP Package

The following information is about the MFPRetail to AP integration.

Scripts Used By the MFPRetail to AP Package

- mfp_to_ap_init.ksh
- mfp_to_ap_finalize.ksh

Data Mapping for MFPRetail to AP Package

Data is sent when mowpappcpnewb is set to TRUE.

Table 4–10 MFPRetail to AP Data

MFP Retail Formula	AP Target Measure
bucprecc	mfcprecc
bucprecu	mfcprecu
bucpagmpv	mfcpposgmr
bucprecr	mfcprecr
bucpslsregr + bucpslsclrr + bucpslspror	mfcpslsr
bucpslsregu + bucpslsclru + bucpslsprou	mfcpslsu

SizeOpt to AP Package

The following information is about the SizeOpt (also known as SPO) to AP integration.

Scripts Used By the SizeOpt to AP Package

- sizeopt_to_ap_init.ksh
- sizeopt_to_ap_finalize.ksh

Data Mapping for SizeOpt to AP Package

Data is sent when populated (not NA).

Table 4–11 SizeOpt to AP Data

SizeOpt Formula	AP Target Measure
sku2sizexlxcg	sztyclssszmapb
esprofxxlaxg	sztyarchsp1up
esprofxxlbcg	sztyarchsp2up
esprofxxlccg	sztyarchsp3up
esprofxxldcg	sztyarachsp4up
exportprofxxlxcg	sztysizeprflup
exportescxxlxcg	szwpescleveltx
sku2atcdxxlxcg	szwpssncodeb

AP to IP Package

The following information is about the AP to IP integration.

Scripts Used By the AP to IP Package

- ap_to_ip_init.ksh
- ap_to_ip_finalize.ksh

Data Mapping for AP to IP Package

Data is sent when populated (not NA).

Table 4–12 AP to IP Data

AP Formula	IP Target Measure
bpcpwksclru	bpcpwksclru
bpcpwksregu	bpcpwksregu
bpcsfpcstu	bpcsfpcstu
bpcsfprtlu	bpcsfprtlu
bpcpexitweekdt	bpcpexitweekdt
bpcpstartweekdt	bpcpstartweekdt
bpcsbopc	bpcsbopc
bpcsbopr	bpcsbopr
bpcsbopu	bpcsbopu
bpcseopc	bpcseopc
bpcseopr	bpcseopr
bpcseopu	bpcseopu
bpcsmkdclrr	bpcsmkdclrr
bpcsmkdpermr	bpcsmkdpermr
bpcsmkdpromor	bpcsmkdpromor
bpcslrprcu	bpcslrprcu
bpcspromopr cr	bpcspromopr cr
bpcspempr cr	bpcspempr cr
bpcsrcc	bpcsrcc
bpcsr cr	bpcsr cr
bpcsr cu	bpcsr cu
bpcsslclrc	bpcsslclrc
bpcsslclrr	bpcsslclrr
bpcsslclru	bpcsslclru
bpcpprepacku	bpcpprepacku
bpcsstrcntu	bpcsstrcntu

Hierarchy Integration with ODI

If all of the Fashion Planning Bundle applications are on one server or their domains are accessible from one server (for example, through NFS mounts), you only need to run each hierarchy integration package once. However, if you are integrating applications across different servers, the listed integration steps must be repeated with different contexts and agents once for each server is integrated.

Adding New Products: PROD Hierarchy Integration

Sometimes it is required to add new products/SKUs to the product (PROD) hierarchies. But each application's PROD hierarchy typically contains different dimensions. Therefore, the Fashion Planning Bundle provides an integration package named Broadcast PROD Hierarchy that adds new products while maintaining the hierarchy synchronization between all applications.

To add new products, create a prod.csv file (in CSV format) that defines the following information for each new product:

Table 4–13 Prod.csv File Dimension Positions/Labels

Dimension Position / Label
SKU
SKU_LABEL
SKUP
SKUP_LABEL
SKUG
SKUG_LABEL
CLR
CLR_LABEL
SKP1
SKP1_LABEL
SKP2
SKP2_LABEL
VDRC
VDRC_LABEL
VNDR
VNDR_LABEL
CLGP
CLGP_LABEL
SCLS
SCLS_LABEL
CLASS
CLASS_LABEL
ITGP
ITGP_LABEL

Table 4–13 (Cont.) Prod.csv File Dimension Positions/Labels

Dimension Position / Label
PL1
PL1_LABEL
PL2
PL2_LABEL
DEPT
DEPT LABEL
PGRP
PGRP_LABEL
DVSN
DVSN_LABEL
CMPP
CMPP_LABEL

The dimensions in [Table 4–13](#) are the union or superset of the PROD hierarchies for all Fashion Planning Bundle applications. An example file named prod.csv is provided in the \$RPAS_HOME/doc directory. You can edit a copy of the file using a spreadsheet program or text editor.

Once the new products and their attributes have been added to prod.csv, perform the following steps to add the products to all applications:

1. Place the updated prod.csv file in the \$RPAS_HOME directory.
2. In the Projects pane in Designer, open the **Interfaces** folder, then the **Packages** folder.
3. Double-click **Broadcast PROD Hierarchy**.
4. Select the context/agents appropriate to the application domain locations.

Note that the value of the FPB_APPS environment variable for the agent.sh process determines which domains to update. See the *Oracle Retail ODI Enabled Integration Implementation Guide* for an example of the FPB_APPS setup.

For more information about each application's PROD hierarchy, see that application's user guide.

Adding New Stores: LOC Hierarchy Integration

The procedure for adding new stores to the location (LOC) hierarchy is similar to that of adding new products. The Broadcast LOC Hierarchy package adds new stores, while maintaining the hierarchy synchronization between applications.

To add new stores, create a loc.csv file (in CSV format) that defines the following information for each new store:

Table 4–14 *Loc.csv File Dimension Positions/Labels*

Dimension Position/Label
STR
STR_LABEL
DSTR
DSTR_LABEL
RGN
RGN_LABEL
AREA
AREA_LABEL
CHNL
CHNL_LABEL
CHN
CHN_BAEL
COMP
COMP_LABEL

The dimensions in [Table 4–14](#) are the union or superset of the LOC hierarchies for all Fashion Planning Bundle applications. An example file named loc.csv is provided under the \$RPAS_HOME/doc directory. You can edit a copy of the file using a spreadsheet program or text editor.

Once the new stores and their attributes have been added to loc.csv, perform the following steps to add the stores to all applications:

1. Place the updated loc.csv file in the \$RPAS_HOME directory.
2. In the Projects pane in Designer, open the **Interfaces** folder, then the **Packages** folder.
3. Double-click **Broadcast LOC Hierarchy**.
4. Select the context/agents appropriate to the application domain locations.

Note that the value of the FPB_APPS environment variable for the agent.sh process determines which domains to update. See the *Oracle Retail ODI Enabled Integration Implementation Guide* for an example of the FPB_APPS setup.

For more information about each application's LOC hierarchy, see that application's user guide.

Script Integration

This chapter describes the script integration of the Fashion Planning Bundle applications.

For information about ODI enabled integration for the Fashion Planning Bundle, see [Chapter 4](#). For information about RMS integration, see [Chapter 6](#).

Integration Scripts

These scripts can be used for moving data between applications. The following rules apply to integration scripts:

- The exportMeasure utility is used to export data in CSV (comma-separated values) format. This maintains the consistency of start and width attributes across different applications.
- Data exported from a source application is placed in the destination domain input directory.
- Export scripts must run before load scripts. They should be run in the batch window.
- Measure names should not be hard-coded. The names are included in a text file named `<source application>_<destination application>_transferList.txt`. For example, `MFP_AP_transferList.txt` includes the measure names exported from MFP to AP.

Each transfer file includes two columns that correspond to the following data:

- Measure name
- File name

When the transfer file is used for a load script, it contains the name of the measure to be loaded and the name of the file that contains the data.

For an export script, the transfer file contains the name of the measure in the source application and the name of the ovr file that will be created in the input directory of the destination application. The file names correspond to the measure names in the destination application.

- The scripts have a command line argument to set the maximum number of processes that need to be run in parallel. Setting this argument can help speed up the performance of independent tasks on local domains. The default is 1.
- Do not hard-code domain paths. The paths are entered as command-line arguments.

Table 5–1 lists the integration scripts for MFP. These scripts are located in `$MFP_HOME/bin`.

Table 5–1 Integration Script

Application	Script Name	Arguments
MFP	loadactuals.ksh	measurelist, maxprocesses
IP	exportToIP.ksh	maxprocesses, destination, measurelist
AP	exportToAP.ksh	maxprocesses, destination, measurelist

Batch Designs

This section contains detailed information on the following integration scripts:

- [Load Actuals](#)
- [Export to IP](#)
- [Export to AP](#)

Load Actuals

Script

loadActuals.ksh

Usage

loadActuals.ksh <measurelist> <maxprocesses>

Argument	Description	Notes
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the <code>\$MFP_HOME/bin</code> directory. The file name should follow the convention of <code>IP_MFP_transferList.txt</code> .
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.

Control File

`IP_MFP_transfer_list.txt` contains the following list of measures that can be loaded as part of this script:

MFP Cost:

- `ipcpbopc`
- `ipcpbopu`
- `ipcpeopc`
- `ipcpeopu`
- `ipcpgmpv`
- `ipcprecc`
- `ipcprecu`
- `ipcplsc`
- `ipcplsr`
- `ipcplsu`

- ipopbopc
- ipopbopu
- ipopeopc
- ipopeopu
- ipopgmpv
- ipoprecc
- ipoprecu
- ipopslsc
- ipopslsr
- ipopslsu

MFP Retail:

- ipcpbopc
- ipcpbopu
- ipcpeopc
- ipcpeopu
- ipcpgmpv
- ipcprecc
- ipcprecu
- ipcpslsr
- ipcpslsu
- ipcpbopr
- ipcpeopr
- ipcprecr
- ipopbopc
- ipopbopu
- ipopeopc
- ipopeopu
- ipopgmpv
- ipoprecc
- ipoprecu
- ipopslsr
- ipopslsu
- ipopbopr
- ipopeopr
- ipoprecr

Example

```
loadActuals.ksh IP_MFP_transfer_list.txt 1
```

Error Information**Table 5–2 Load Actuals Error Information**

Task Name	Error Code	Abort Required?	Description of Error
loadactuals	40	yes	Arguments are missing.
loadactuals	41	yes	Domain path does not exist.
loadactuals	42	yes	Data file does not exist.
loadactuals	43	no	All measure input files are empty or missing.
loadactuals	44	no	localdomainlist.cfg file is missing.
loadactuals	45	yes	Errors occurred during the load of one or more measures.

Notes

- This script uses the RPAS loadMeasure utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script ignores any missing or empty measure load files.
- All measure files are placed into the domain's input folder.
- If there were no errors during the loading of all measures, the input file is archived into the domain's input/processed directory. A date stamp is appended to the end of the file name.
- The script does not produce an error when records are rejected from the loaded files. These rejected records are logged in the log output of the load process.

Export to IP**Script**

```
exportToIP.ksh
```

Usage

```
exportToIP.ksh <maxprocesses> <destination> <measurelist>
```

Argument	Description	Notes
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.
destination	Sets the path to the input folder of the destination domain where the exported files are placed.	The export script creates the file containing the exported measures in the input folder of the destination domain.
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the \$MFP_HOME/bin directory. The file name should follow the convention of MFP_IP_transferList.txt.

Control File

MFP_IP_transfer_list.txt contains the following list of measures that can be exported as part of this script:

MFP Cost:

- buopbopc buopbopc
- buopeopc buopeopc
- buopgmpv buopgmpv
- buoprecc buoprecc
- buopslsc buopslsc
- buopslsnetr buopslsnetr
- buopslsr buopslsr
- buopbopu buopbopu
- buopeopu buopeopu
- buoprecu buoprecu
- buopslsu buopslsu
- bucpbopu bucpbopu
- bucppeopu bucppeopu
- bucprecu bucprecu
- bucpslsu bucpslsu
- bucpbopc bucpbopc
- bucppeopc bucppeopc
- bucpgmpv bucpgmpv
- bucprecc bucprecc
- bucpslsc bucpslsc
- bucpslsnetr bucpslsnetr
- bucpslsr bucpslsr

MFP Retail:

- bucpbopc bucpbopc
- bucpbopr bucpbopr
- bucppeopc bucppeopc
- bucppeopr bucppeopr
- bucpgmpv bucpgmpv
- bucpmkdr bucpmkdr
- bucprecc bucprecc
- bucpreocr bucpreocr
- bucpslsr buopbopc
- buopbopc buopbopc
- buopbopr buopbopr

- buopeopc buopeopc
- buopeopr buopeopr
- buopgmpv buopgmpv
- buopmkdr buopmkdr
- buoprecc buoprecc
- buoprecr buoprecr
- buopslsr buopslsr
- bucpbopu bucpbopu
- bucpeopu bucpeopu
- bucprecu bucprecu
- bucpslsu bucpslsu
- buopbopu buopbopu
- buopeopu buopeopu
- buoprecu buoprecu
- buopslsu buopslsu

Error Information

Table 5–3 Export to IP Error Information

Task Name	Error Code	Abort Required?	Description of Error
export	200	yes	Argument missing.
export	210	yes	Domain does not exist.
export	220	yes	Missing \$EXPORTLISTFILE.
export	230	no	Errors found in the export log file.

Notes

- This optional script is used only if exports to IP are required.
- The script uses the RPAS exportData utility to export measure data from the domain. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.

Export to AP

Script

exportToAP.ksh

Usage

exportToAP.ksh <maxprocesses> <destination> <measurelist>

Argument	Description	Notes
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.
destination	Sets the path to the input folder of the destination domain where the exported files are placed.	The export script creates the file containing the exported measures in the input folder of the destination domain.
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the \$MFP_HOME/bin directory. The file name should follow the convention of MFP_AP_transferList.txt.

Control File

MFP_AP_transfer_list.txt contains the following list of measures that can be exported as part of this script:

MFP Cost:

- bucpmpv mfcposgmr
- bucpbcc mfcprecc
- bucpslr mfcpslr
- bucpccu mfcpreccu
- bucpssu mfcpslsu

MFP Retail:

- bucpmpv mfcposgmr
- bucpbcc mfcprecc
- bucpslr mfcpslr
- bucpccu mfcpreccu
- bucpssu mfcpslsu
- bucpccr mfcpreccr

Error Information

Table 5-4 Export to AP Error Information

Task Name	Error Code	Abort Required?	Description of Error
export	200	yes	Argument missing.
export	210	yes	Domain does not exist.
export	220	yes	Missing \$EXPORTLISTFILE.
export	230	no	Errors found in the export log file.

Notes

- This optional script is used only if exports to AP are required.
- The script uses the RPAS exportData utility to export measure data from the domain. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.

MFP - RMS Integration

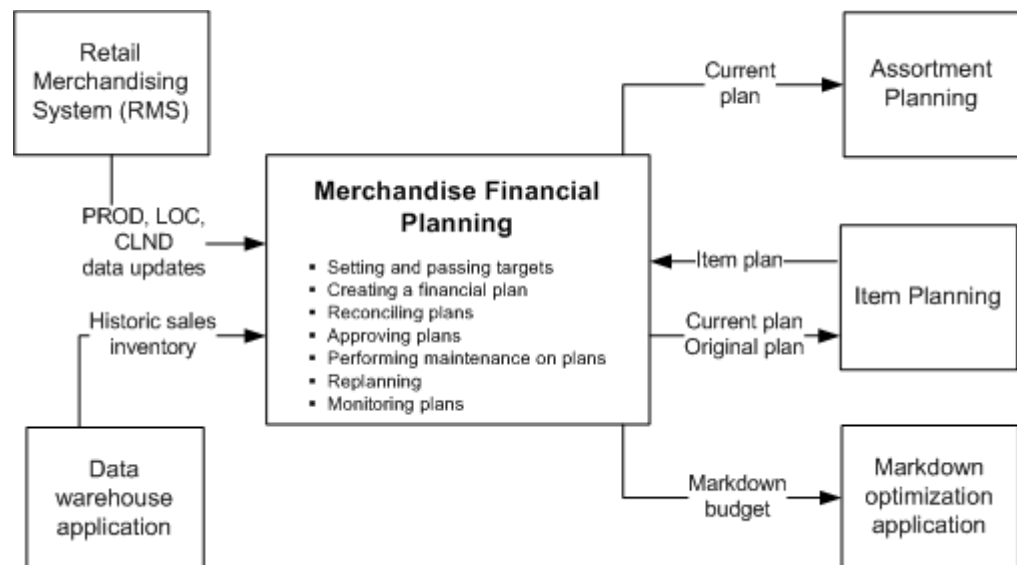
This chapter describes the overall MFP data flow and integration with RMS, markdown optimization applications, and data warehouses.

For information about ODI enabled integration for the Fashion Planning Bundle, see [Chapter 4](#). For information about script integration for the bundle, see [Chapter 5](#).

Overview of the MFP Data Flow

[Figure 6–1](#) shows the integration of MFP with other applications and the flow of data between the applications.

Figure 6–1 Data Flow for the Integration of MFP with Other Applications



Integration Interface Data Flow Description

These descriptions explain some of the data flows in [Figure 6-1](#). For information about the data flows among MFP, IP, and AP, see [Chapter 4](#) and [Chapter 5](#).

From RMS to MFP

Note: The integration between RMS and MFP includes only hierarchy, on order, and inventory data. All other data required by MFP are not part of the RMS/MFP integration.

Data for the following hierarchies is imported into MFP from RMS:

- Product (PROD) hierarchy
- Location (LOC) hierarchy
- Calendar (CLND) hierarchy

Note: Non-stockholding company stores and non-inventory items which are non-merchandise items, consignment, concession and deposit returns are sent from RMS to MFP but are not utilized in MFP.

Hierarchies are the structures used by an organization to define the relationships that exist between measures of data, products, locations, time, and other dimensions. These dimensions are represented within the Fashion Planning Bundle applications as hierarchies that correspond to an organization's structure, including all roll-ups.

The Product hierarchy provides the parent-child merchandise level relationships that are available within an application. The Location hierarchy provides the parent-child-location level relationships that are available within an application. Application data is presented at an intersection level of the Product, Location, and Calendar hierarchies.

In addition to the hierarchy files, MFP receives on order and inventory from RMS. These files are based at the week/style-color/store level and then aggregated to the planning levels in the MFP domain.

For more information about the RMS integration, see [RMS to MFP Transformation](#). For additional details on the RMS/MFP integration from the perspective of RMS, see the RPAS/MFP - RMS Integration chapter of the *RMS Operations Guide - Volume 1*.

From a Data Warehouse Application to MFP

The following data is imported into MFP from a data warehouse application:

- Historic sales
- Inventory

From MFP to a Markdown Optimization Engine

MFP Retail sends the current plan markdown budget to a markdown optimization application such as Oracle Retail Markdown Optimization.

For more information, see [Markdown Optimization Integration](#).

RMS to MFP Transformation

This section details the process that is required to transform the extracted RMS files to create load ready files for the MFP solution.

Script Installation

1. The RETLforRpas directory and MFPIntegration directory should be installed under the same directory. Both directory structures should overlap because MFP uses the schema and environment files of the RETLforRpas release. The RETLforRpas is part of the RMS release while the MFPIntegration directory is part of the MFP release.
2. On some flavors of Unix, you may not have the `nawk` executable. You may only have the `awk` executable. In that case a soft link is needed to be created with the name `nawk` which will point to the `awk` executable.
3. RETL should be installed in the machine.

Environment Setup

The following environment variables need to be set in the environment:

- RMSE_RPAS_HOME points to the RMS release of RPAS integration scripts directory


```
export RMSE_RPAS_HOME=$MFP_HOME
```
- The DATA_DIR is the input and output directory. The RMS input files has to be copied to the DATA_DIR directory. After the MFP transformation, the output files also will be stored in the same location. Once the processing is done, the processed input files will be transfer to the processed directory in the DATA_DIR.
- RMSE_RPAS_HOME points to the RMS release of RPAS integration scripts directory


```
export RMSE_RPAS_HOME=$MFP_HOME
```
- As the RMS environment scripts use the RDF_HOME variable internally, this needs to be set.


```
export RDF_HOME=$MFP_HOME
```
- The RMS Schema directory


```
export RMS_RPAS_SCHEMA_DIR=$RMSE_RPAS_HOME/rfx/schema/
```
- The MFP Schema directory


```
export MFP_SCHEMA_DIR=$MFP_HOME/rfx/schema/
```
- The intermediate data files are stored in the MFP_TEMP_DIR, If the input files are very huge, it is advisable to use a special Temp directory, rather than using the default OS temp directory. By default /tmp is taken as MFP_TEMP_DIR if this variable is not set.


```
export MFP_TEMP_DIR=$MFP_HOME/data
```

In addition to the variable described above, the following optional environment variables can be set:

- **ADD_AT_SIGN_TO_WH_DESC:** This variable is used by the `mfp_lochier.ksh`. In the warehouse records if @ sign needs to be prefixed with warehouse name description field then this should be set to True. By default this feature is disabled. If this feature is required the following command should be executed in the environment.

```
export ADD_AT_SIGN_TO_WH_DESC=True
```

- **PROCESS_WAREHOUSE_DATA:** This variable is used by the `mfp_lochier.ksh`. By default this variable is set to True. If you do not want the `mfp_lochier.ksh` to process the warehouse data then this variable must be set as False.

```
export PROCESS_WAREHOUSE_DATA=False
```

- **ENABLE_INVENTORY_RETAIL**
ENABLE_ON_ORDER_RETAIL

These two variables are used by the `mfp_inventory.ksh` and `mfp_on_order.ksh`. The setting of these variables is dependent on the type of MFP being used. If you are using MFP Retail, you need to set these values to true to get the retail version of the inventory and on order files. By default both these variables are set to True. Use the following command in environment to disable them if you are instead using MFP Cost.

```
export ENABLE_INVENTORY_RETAIL=False
export
ENABLE_ON_ORDER_RETAIL=False
```

- **ENABLE_INVENTORY_COST**
ENABLE_ON_ORDER_COST

These two variables are used by the `mfp_inventory.ksh` and `mfp_on_order.ksh`. The setting of these variables is dependent on the type of MFP being used. If you are using MFP Cost, you need to set these values to true to get the cost version of the inventory and on order files. By default both these variables are set to True. Use the following command in environment to disable them if you are instead using MFP Retail.

```
export ENABLE_INVENTORY_COST=False
export
ENABLE_ON_ORDER_COST=False
```

RMS Integration Script Details

This section outlines the scripts that will be run to transform the RMS files to load into MFP. These scripts reside in the \$MFP_HOME/rfx/src directory. As previously stated, the input files and output files are in the \$DATA_DIR.

- Script name: mfp_calhier.ksh
Description: Transforms exported calendar hierarchy data from RMS to MFP-loadable format.
Input data file(s): rmse_rpas_clndmstr.dat
Output file(s): clnd.csv.dat
Params: None
- Script name: mfp_prodhier.ksh
Description: Transforms exported product hierarchy data from RMS to MFP-loadable format.
Input file(s): rmse_rpas_item_master.dat, rmse_rpas_merchhier.dat
Output file(s): prod.csv.dat
Params: None
- Script name: mfp_lochier.ksh
Description: Transforms exported location hierarchy data from RMS to MFP-loadable format.
Input data file(s): rmse_rpas_orghier.dat, rmse_rpas_store.dat and rmse_rpas_wh.dat
Output file(s): loc.csv.dat
Params: None
- Script name: mfp_inventory.ksh
Description: Transforms the extracted Inventory data from RMS to an MFP-loadable format.
Required input data file: rmse_mfp_inventory.W.dat
Optional Input data file: rmse_mfp_inventory.I.dat
Output file(s):
Params: None
- Script name: mfp_on_order.ksh
Description: Transforms the extracted On-order data from RMS into an MFP-loadable format.
Required input data file(s): rmse_mfp_onorder.dat
Output file(s): mfp_inventory_retail.csv.ovr, mfp_inventory_cost.csv.ovr
Params: None

Once the processing is done, the processed input files are transferred to the processed directory in the \$DATA_DIR. The output files need to be transferred to the MFP domains input directory in order to load with the standard load scripts. For more about the standard load scripts, see [Batch Designs](#).

Markdown Optimization Integration

The exportToMDO.ksh script is located in the MFP_HOME/bin folder. It is used to create MFP Retail current plan markdown budget extract for exports to markdown optimization applications such as Oracle Retail Markdown Optimization (MDO).

The script takes three arguments:

- The location for the destination directory.
- The RPAS name for a dimension along the location hierarchy. All location dimensions above or at the lowest stored dimension, channel (CHNL), are valid. You must provide a valid dimension.
- The RPAS name for a dimension along the product hierarchy. All product dimensions above or at the lowest stored dimension, subclass (SCLS), are valid. You must provide a valid dimension.

The script generates the total value of current approved markdowns. This is the sum of the promotional, permanent, and clearance markdowns at the base intersection. This is derived from the product and location dimensions that you supply.

Note: The script assumes the calendar dimension to be month.

The markdowns, along with position names along the dimensions, are written to the ASH_BUDGET_TBL.dat file in the destination directory that you provide. It is a pipe delimited file that contains the following information:

- Product key
- Location key
- Fiscal year
- Fiscal month
- Total markdown

The product key (also known as the merchandise key) and location key are the position names for the product and location. The year is a four digit year. The month is an index of the fiscal month. For instance, if you start your fiscal year on February 1, the calendar month of December 2011 would be 2011 | 11.

Batch Processing

This chapter contains all the scripts that are needed to run and maintain the MFP environment.

Batch Structure Overview

The following directories are used by the batch scripts. These directories are subdirectories of the `$MFP_HOME` directory.

Table 7-1 *Directories Used by Batch Scripts*

Directory Name	Content of the Directory
bin	Batch scripts
config	MFP template configuration
domain	Domains
input	Input files for building the domain
logs	Log files from running any of the batch scripts A system administrator can scan the logs for any errors, exceptions, or failures. If there are none, the batch completed successfully.
temp	Temporary files used by the batch scripts

Batch Scheduling

Scheduling of the batch scripts are in the following categories:

- [Daily Batch Scripts](#)
- [Weekly Batch Scripts](#)
- [Unscheduled Administration Script](#)

The following information is included in the tables for each batch script:

- A short description of the script
- The name of the script
- The directory in the `$MFP_HOME/bin` directory where the batch script is found
- Dependencies on other batch scripts

For a detailed description of each script, see [Batch Designs](#).

Daily Batch Scripts

The daily batch scripts are run every day before executing the weekly batch scripts.

Table 7–2 lists information on the daily batch script. For more information on the Load Actuals and Export to IP scripts, see [Integration Scripts](#) in [Chapter 5](#).

Table 7–2 Daily Batch Scripts

Description	Script Name	Batch Directory	Dependency
Backup	N/A	N/A	N/A
Export to AP	exportToAP.ksh	bin	Backup
Export to IP	exportToIP.ksh	bin	Backup
Load Actuals	loadActuals.ksh	bin	Backup

Weekly Batch Scripts

The daily batch scripts are run before executing the weekly batch scripts.

Table 7–3 lists information on the weekly batch scripts.

Table 7–3 Weekly Batch Scripts

Description	Script Name	Batch Directory	Dependency
Backup	N/A	N/A	N/A
Export to AP	exportToAP.ksh	bin	Backup
Export to IP	exportToIP.ksh	bin	Backup
Calendar Hierarchy Load	loadhier.ksh	loadhier	Backup
Product Hierarchy Load	loadhier.ksh	loadhier	Backup
Location Hierarchy Load	loadhier.ksh	loadhier	Backup
Load on order data	loadActuals.ksh	bin	Calendar Hierarchy Load, Product Hierarchy Load, Location Hierarchy Load
Load Actuals data	loadActuals.ksh	bin	Calendar Hierarchy Load, Product Hierarchy Load, Location Hierarchy Load
Load IP data	loadActuals.ksh	bin	Calendar Hierarchy Load, Product Hierarchy Load, Location Hierarchy Load, and export file ready from IP
Propagate inventory and aggregate data for all planning levels	processactuals.ksh	actualize	Load on order data, Load Actuals data
Generate sales forecast	runforecast.ksh	forecast	Load on order data, Load Actuals data
Refresh existing workbooks Note: This script is optional.	refresh.ksh	workbook	Generate sales forecast

Table 7-3 (Cont.) Weekly Batch Scripts

Description	Script Name	Batch Directory	Dependency
Auto build workbooks placed on queue Note: This script is optional.	autobuild.ksh	workbook	None

Unscheduled Administration Script

The following script is not part of a normal batch schedule. This script is run as a one-off process to perform the specified activity.

[Table 7-4](#) lists information on the unscheduled administration script.

Table 7-4 Unscheduled Administration Script

Description	Script Name	Batch Directory	Dependency
Load mapping	loadActuals.ksh	bin	None

Batch Environment Scripts

These scripts are included in the other batch scripts to control logging and set environment variables.

The first script, `message.ksh`, controls the overall logging. The script writes batch script details to a daily log file. The daily log file is created in the `$MFP_HOME/logs` directory. The format of the file name is `MnthID_Day.log`, for example, `Apr_02.log`.

The second script, `environment.ksh`, is called at the beginning of every batch script. This script sets the following environment variables:

- `export MFP_CONFIGNAME=mfprtl` for MFP Retail and `mfpcst` for MFP Cost
- `export MFP_DOMAINHOME=$MFP_HOME/domain`
- `export MFP_MASTERDOMAIN=$MFP_DOMAINHOME/mfprtl` for MFP Retail and `mfpcst` for MFP Cost
- `export MFP_CONFIGHOME=$MFP_HOME/config`
- `export MFP_EXPORT=$MFP_HOME/export`
- `export MFP_EXPORT_TOAP=$MFP_HOME/exportToAP`
- `export MFP_INPUThOME=$MFP_HOME/input`
- `export MFP_LOG_DIR=$MFP_HOME/logs`
- `export MFP_LIB=$MFP_HOME/bin`
- `export MFP_TEMP=$MFP_HOME/temp`
- `export MFP_BATCH=$MFP_HOME/bin`
- `export LOGLEVEL=all`

The `LOGLEVEL` parameter can be set to any of the RPAS supported logging levels—all, profile, debug, audit, information, warning, error, and none.

Batch Designs

This section contains detailed information on the following batch scripts:

- [Auto Build Workbooks](#)
- [Calendar Hierarchy Load](#)
- [Product Hierarchy Load](#)
- [Load Actuals Data](#)
- [Load IP Data](#)
- [Load On Order Data](#)
- [Load Mapping Measures](#)
- [Location Hierarchy Load](#)
- [Process Actuals Data](#)
- [Refresh Workbooks](#)
- [Run Batch Forecast](#)

Some of the scripts have a command line argument to set the maximum number of processes that need to be run in parallel. Setting this argument can help speed up the performance of independent tasks on local domains. The default is 1.

Calendar Hierarchy Load

Script

loadhier.ksh

Usage

loadhier.ksh <hierarchy> <purgeage> <maxprocesses>

Argument	Description	Notes
hierarchy	Sets the RPAS name of the hierarchy to be loaded.	The default for the calendar hierarchy is clnd.
purgeage	Sets the number of days after updating a hierarchy position before purging the position.	The default is 10000.
maxprocesses	Sets the maximum number of hierarchy load processes to run in parallel.	The default is 1.

Example

loadhier.ksh clnd 10000 10

Error Information

Table 7-5 Calendar Hierarchy Load Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadhier	20	yes	One or more arguments are missing.
loadhier	21	yes	Domain does not exist.
loadhier	22	no	Hierarchy file does not exist.
loadhier	24	yes	Loading the hierarchy produced errors. Further batch processing should not be run until the error has been investigated.

Notes

- Before running this script, a clnd.dat file must exist in the input directory of the domain. This can be created by running the transformation scripts to transform the RMS files. For more information, see [Chapter 6](#).
- This script uses the RPAS loadHier utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The calendar, product, and location hierarchy loads can be run in any order.

Product Hierarchy Load

Script

loadhier.ksh

Usage

loadhier.ksh <hierarchy> <purgeage> <maxprocesses>

Argument	Description	Notes
hierarchy	Sets the RPAS name of the hierarchy to be loaded.	The default for the product hierarchy is prod.
purgeage	Sets the number of days after updating a hierarchy position before purging the position.	The default is 10000.
maxprocesses	Sets the maximum number of hierarchy load processes to run in parallel.	The default is 1.

Example

loadhier.ksh prod 10000 10

Error Information

Table 7–6 Product Hierarchy Load Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadhier	20	yes	One or more arguments are missing.
loadhier	21	yes	Domain does not exist.
loadhier	22	no	Hierarchy file does not exist.
loadhier	24	yes	Loading the hierarchy produced errors. Further batch processing should not be run until the error has been investigated.

Notes

- Before running this script, a prod.dat file must exist in the input directory of the domain. This can be created by running the transformation scripts to transform the RMS files. For more information, see [Chapter 6](#).
- This script uses the RPAS loadHier utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The calendar, product, and location hierarchy loads can be run in any order.

Location Hierarchy Load

Script
loadhier.ksh

Usage
loadhier.ksh <hierarchy> <purgeage> <maxprocesses>

Argument	Description	Notes
hierarchy	Sets the RPAS name of the hierarchy to be loaded.	The default for the product hierarchy is loc.
purgeage	Sets the number of days after updating a hierarchy position before purging the position.	The default is 10000.
maxprocesses	Sets the maximum number of hierarchy load processes to run in parallel.	The default is 1.

Example
loadhier.ksh loc 10000 10

Error Information

Table 7-7 Location Hierarchy Load Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadhier	20	yes	One or more arguments are missing.
loadhier	21	yes	Domain does not exist.
loadhier	22	no	Hierarchy file does not exist.
loadhier	24	yes	Loading the hierarchy produced errors. Further batch processing should not be run until the error has been investigated.

Notes

- Before running this script, a loc.dat file must exist in the input directory of the domain. This can be created by running the transformation scripts to transform the RMS files. For more information, see [Chapter 6](#).
- This script uses the RPAS loadHier utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The calendar, product, and location hierarchy loads can be run in any order.

Load On Order Data

Script

loadActuals.ksh

Usage

loadActuals.ksh <measurelist> <maxprocesses>

Argument	Description	Notes
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the \$MFP_HOME/bin directory. The file name should follow the convention of IP_MFP_transferList.txt.
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.

Control File

LoadOnOrderList.txt: Contains the following list of measures that can be loaded as part of this script:

MFP Cost:

- drtyonordc
- drtyonordu

MFP Retail:

- drtyonordc
- drtyonordu
- drtyonordr

Example

loadActuals.ksh LoadOnOrderList.txt 1

Error Information

Table 7-8 Load Actuals Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadactuals	40	yes	One or more arguments are missing.
loadactuals	41	yes	Domain does not exist.
loadactuals	42	no	Data file does not exist.
loadactuals	43	yes	All measure input files are empty or missing.
loadactuals	44	yes	Configuration file does not exist.
loadactuals	45	yes	Errors occurred during the load of one or more measures.

Notes

- If you are using RMS on order file, ensure that you run the transformation and transfer the load ready file to the input directory of the domain. For more information, see [Chapter 6](#).
- This script uses the RPAS loadmeasure utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.

- The script ignores any missing or empty measure load files.
- All measure files are placed into the domain's input folder.
- If there were no errors during the loading of all measures, the input file is archived into the domain's input/processed directory. A date stamp is appended to the end of the file name.
- The script does not produce an error when records are rejected from the loaded files. These rejected records are logged in the log output of the load process.

Load Actuals Data

Script

loadActuals.ksh

Usage

loadActuals.ksh <measurelist> <maxprocesses>

Argument	Description	Notes
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the \$MFP_HOME/bin directory. The file name should follow the convention of IP_MFP_transferList.txt.
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.

Control File

LoadActualsList.txt: Contains the following list of measures that can be loaded as part of this script:

MFP Cost:

- drtybopclrc
- drtybopclru
- drtybopregc
- drtybopregu
- drtyeopclrc
- drtyeopclru
- drtyeopregc
- drtyeopregu
- drtyrecc
- drtyrecu
- drtyslclrr
- drtyslclru
- drtyslclrc
- drtyslpror
- drtyslsprou
- drtyslsprou
- drtyslproc

- drtyslsregr
- drtyslsregu
- drtyslsregc
- drtydevalc
- buwpvatp
- drtyshrinkc
- drtyshrinku
- drtymiscinc
- drtymiscinu
- drtymiscoutc
- drtymiscoutu

MFP Retail:

- drtybopclrc
- drtybopclru
- drtybopclrr
- drtybopregc
- drtybopregu
- drtybopregr
- drtyeopclrc
- drtyeopclru
- drtyeopclrr
- drtyeopregc
- drtyeopregu
- drtyeopregr
- drtyrecc
- drtyrecu
- drtyrecr
- drtyslsclrr
- drtyslsclru
- drtyslspror
- drtyslsprou
- drtyslsregr
- drtyslsregu
- buwpvatp
- drtyshrinkr
- drtyshrinku
- drtymiscinr

- drtymiscinu
- drtymiscoutr
- drtymiscoutu
- drtymkdr
- drtymkupr

Example

loadActuals.ksh LoadOnOrderList.txt 1

Error Information

Table 7–9 Load Actuals Data Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadactuals	40	yes	One or more arguments are missing.
loadactuals	41	yes	Domain does not exist.
loadactuals	42	no	Data file does not exist.
loadactuals	43	yes	All measure input files are empty or missing.
loadactuals	44	yes	Configuration file does not exist.
loadactuals	45	yes	Errors occurred during the load of one or more measures.

Notes

- If you are using RMS inventory file, ensure that you run the transformation and transfer the load ready file to the input directory of the domain. For more information, see [Chapter 6](#).
- This script uses the RPAS loadmeasure utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script ignores any missing or empty measure load files.
- All measure files are placed into the domain's input folder.
- If there were no errors during the loading of all measures, the input file is archived into the domain's input/processed directory. A date stamp is appended to the end of the file name.
- The script does not produce an error when records are rejected from the loaded files. These rejected records are logged in the log output of the load process.

Load IP Data

Script

loadActuals.ksh

Usage

loadActuals.ksh <measurelist> <maxprocesses>

Argument	Description	Notes
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the \$MFP_HOME/bin directory. The file name should follow the convention of IP_MFP_transferList.txt.
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.

Control File

LoadActualsList.txt: Contains the following list of measures that can be loaded as part of this script.

MFP Cost:

- ipcpbopc ipcpbopc
- ipcpbopu ipcpbopu
- ipcpeopc ipcpeopc
- ipcpeopu ipcpeopu
- ipcpgmpv ipcpgmpv
- ipcprecc ipcprecc
- ipcprecu ipcprecu
- ipcpslsc ipcpslsc
- ipcpslsr ipcpslsr
- ipcpslsu ipcpslsu
- ipopbopc ipopbopc
- ipopbopu ipopbopu
- ipopecopc ipopecopc
- ipopecopu ipopecopu
- ipopgmpv ipopgmpv
- ipoprecc ipoprecc
- ipoprecu ipoprecu
- ipopslsc ipopslsc
- ipopslsr ipopslsr
- ipopslsu ipopslsu

MFP Retail:

- ipcpbopc ipcpbopc
- ipcpbopu ipcpbopu
- ipcpeopc ipcpeopc
- ipcpeopu ipcpeopu
- ipcpgmpv ipcpgmpv
- ipcprecc ipcprecc
- ipcprecu ipcprecu
- ipcpslsr ipcpslsr
- ipcpslsu ipcpslsu
- ipcpbopr ipcpbopr
- ipcpeopr ipcpeopr
- ipcprecr ipcprecr
- ipopbopc ipopbopc
- ipopbopu ipopbopu
- ipopeopc ipopeopc
- ipopeopu ipopeopu
- ipopgmpv ipopgmpv
- ipoprecc ipoprecc
- ipoprecu ipoprecu
- ipopslsr ipopslsr
- ipopslsu ipopslsu
- ipopbopr ipopbopr
- ipopeopr ipopeopr
- ipoprecr ipoprecr

Example

loadActuals.ksh LoadActualsList.txt 1

Error Information

Table 7-10 Load Actuals Data Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadactuals	40	yes	One or more arguments are missing.
loadactuals	41	yes	Domain does not exist.
loadactuals	42	no	Data file does not exist.
loadactuals	43	yes	All measure input files are empty or missing.
loadactuals	44	yes	Configuration file does not exist.
loadactuals	45	yes	Errors occurred during the load of one or more measures.

Notes

- This script uses the RPAS loadmeasure utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script ignores any missing or empty measure load files.
- All measure files are placed into the domain's input folder.
- If there were no errors during the loading of all measures, the input file is archived into the domain's input/processed directory. A date stamp is appended to the end of the file name.
- The script does not produce an error when records are rejected from the loaded files. These rejected records are logged in the log output of the load process.

Process Actuals Data**Script**

processactuals.ksh

Usage

processactuals.ksh

Error Information**Table 7-11 Process Actuals Error Information**

Task Name	Error Code	Abort Required?	Description of Error
processactuals	60	yes	Domain does not exist.
processactuals	61	yes	Errors occurred during the processing of actualized data.

Notes

- This script uses the RPAS mace utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script can run the batch_calc rule group.

Run Batch Forecast

Script

runforecast.ksh

Usage

runforecast.ksh

Error Information

Table 7–12 Run Batch Forecast Error Information

Task Name	Error Code	Abort Required?	Description of Error
runforecast	70	yes	Domain does not exist.
runforecast	71	yes	Errors occurred during forecast creation. Only forecast measures are affected by this.

Notes

- This script uses the RPAS mace utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script can run the batch_fc rule group. This rule group can execute the in-season and preseason embedded forecasts.

Refresh Workbooks

Script

```
refresh.ksh
```

Usage

```
refresh.ksh <maxprocesses>
```

where

maxprocesses is the maximum number of workbook refresh processes to run in parallel.

Argument	Description	Notes
maxprocesses	Sets the maximum number of refresh processes to run in parallel.	The default is 1.

Example

```
refresh.ksh 10
```

Error Information

Table 7–13 Refresh Workbooks Error Information

Task Name	Error Code	Abort Required?	Description of Error
refresh	120	yes	Argument missing.
refresh	121	yes	Domain does not exist.
refresh	122	no	Workbook list is either blank or missing. This may be the result of there being no workbooks available to refresh.
refresh	123	no	Errors occurred during workbook refresh processing.

Notes

- This is an optional script. It enables all workbooks to be kept current with the elapsed settings and prevents stale data from being committed.
- The script uses the RPAS wbbatch utility to refresh workbooks in the auto build queue. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- Workbooks should be committed before running the refresh batch job. If a workbook is not committed, the data in that workbook can be overwritten by data in the domain during the refresh process.
- If running both the auto build and refresh scripts, it is important to run the refresh job first. The refresh updates all MFP workbooks saved in the domain. If the auto build is run first, these workbooks are unnecessarily refreshed.

Auto Build Workbooks

Script

autobuild.ksh

Usage

autobuild.ksh <maxprocesses>

Argument	Description	Notes
maxprocesses	Sets the maximum number of autobuild workbook processes to run in parallel.	The default is 1.

Example

autobuild.ksh 10

Error Information

Table 7–14 Auto Build Workbooks Error Information

Task Name	Error Code	Abort Required?	Description of Error
autobuild	110	yes	Argument missing.
autobuild	111	yes	Domain does not exist.
autobuild	112	no	Errors occurred during auto workbook build processing.

Notes

- This script is an optional script. It should be used only if auto workbooks are utilized.
- The script uses the RPAS wbatch utility to build workbooks in the auto build queue. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- Auto build entries must be added via the domain before running this script. See the *Oracle Retail Predictive Application Server User Guide* for details on this process.
- If you are running both the auto build and refresh scripts, it is important to run the refresh job first. The refresh updates all MFP workbooks saved in the domain. If the auto build is run first, these workbooks are unnecessarily refreshed.

Load Mapping Measures

Script

loadActuals.ksh

Usage

loadActuals.ksh <measurelist> <maxprocesses>

Argument	Description	Notes
measurelist	Sets the location of the file which contains the list of measures to be exported.	By default, these files are provided with the package in the \$MFP_HOME/bin directory. The file name should follow the convention of IP_MFP_transferList.txt.
maxprocesses	Sets the maximum number of export processes to run in parallel.	The default is 1.

Control File

LoadMapList.txt: Contains the following list of measures that can be loaded as part of this script:

- bulylagtx
- buallagtx

Example

loadActuals.ksh LoadMapList.txt 1

Error Information

Table 7–15 Load Actuals Error Information

Task Name	Error Code	Abort Required?	Description of Error
loadactuals	40	yes	One or more arguments are missing.
loadactuals	41	yes	Domain does not exist.
loadactuals	42	no	Data file does not exist.
loadactuals	43	yes	All measure input files are empty or missing.
loadactuals	44	yes	Configuration file does not exist.
loadactuals	45	yes	Errors occurred during the load of one or more measures.

Notes

- This script uses the RPAS loadmeasure utility. See the *Oracle Retail Predictive Application Server Administration Guide* for details on this utility.
- The script ignores any missing or empty measure load files.
- All measure files are placed into the domain's input folder.
- If there were no errors during the loading of all measures, the input file is archived into the domain's input/processed directory. A date stamp is appended to the end of the file name.
- The script does not produce an error when records are rejected from the loaded files. These rejected records are logged in the log output of the load process.