Oracle® Retail Predictive Application Server Cloud Edition

Administration Guide

Release 17.1

E98038-10

May 2019

ORACLE®

Oracle Retail Predictive Application Server Cloud Edition Administration Guide, Release 17.1

E98038-10

Value-Added Reseller (VAR) Language

Oracle Retail VAR Applications

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

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

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

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

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

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR

Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

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

Contents

4 Internationalization

5 Online Administration Tools

6 Informational Utilities

7 Data Administration

13   Additional Configuration

14   Performance Diagnostic Tool

A Appendix: RPASCE Client Logging Configuration

Send Us Your Comments

Oracle Retail Predictive Application Server Cloud Edition Administration Guide, Release 17.1

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

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

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

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

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

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

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

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

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

Preface

Oracle Retail Administration 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

This document is intended for the users and administrators of Oracle Retail Predictive Application Server. This may include merchandisers, buyers, and business analysts.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For more information, see the following documents in the Oracle Retail Predictive Application Server Cloud Edition documentation set:

- *Oracle Retail Predictive Application Server Cloud Edition Configuration Tools User Guide*
- *Oracle Retail Predictive Application Server Cloud Edition Release Notes*
- *Oracle Retail Predictive Application Server Cloud Edition Security Guide*
- *Oracle Retail Predictive Application Server Cloud Edition User Guide*

Customer Support

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

https://support.oracle.com

When contacting Customer Support, please provide the following:

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

Improved Process for Oracle Retail Documentation Corrections

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

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

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

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

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

Oracle Retail Documentation on the Oracle Technology Network

Oracle Retail product documentation is available on the following web site:

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

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

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introduction

After the RPAS Cloud Edition (RPASCE) Server and Client have been installed and set up, you must complete several online administration activities before users can begin using RPASCE and RPASCE applications. The following are the primary activities for online administration.

- User Maintenance
- System Administration
- Translation Administration
- Online Administration Tools
  - Informational Utilities
  - Data Management
  - Operational Utilities

Before you start any of these activities, you should understand the basics of RPASCE, domains, segments, workbooks, views, hierarchies, and measures.

Basic RPASCE Concepts

RPASCE is a configurable platform with proven scalability for developing multidimensional forecasting-based and planning-based solutions. This platform provides capabilities such as a multidimensional database structure, batch and online processing, a configurable slice-and-dice user interface, a sophisticated configurable calculation engine, user security, and utility functions such as importing and exporting, all on a highly scalable technical environment.

This section introduces you to the following RPASCE concepts:

- Multidimensionality
- Dimensions
- Measures
- Domains, Segments, and Workbooks
- Measure Data

Multidimensionality

In RPASCE, information is stored and represented based on the multidimensional framework. In a multidimensional database system, data is presented as a

multidimensional array, where each individual data value is contained within a cell accessible by multiple indexes.

Multidimensional database systems are a complementary technology to entity relational systems and achieve performance levels above the relational database systems. Applications that run on RPASCE identify data through dimensional relationships. Dimensions are qualities of an item (such as a product, location, or time) or components of a dimension that define the structure and roll up within the dimension.

Dimensions

Dimensions describe the top-to-bottom relationship between the levels or positions of the dimensions in RPASCE. They reflect the dimensions set up at your business and used by the merchandising solutions.

RPASCE supports many alternative dimensions that provide different roll ups and help analyze the data from a different perspective.

Measures

Measures represent the events or measurements that are recorded, while the positions in the dimensions provide a context for the measurement. Measures are defined based on the business rules set in the application. The dimensionality of a measure is configured through the definition of its base intersection, which is the collection of levels (one per appropriate dimension) defining the lowest level at which the information is stored for the measure.

Measure names are completely configurable and typically named using a convention that identifies each component and the meaning of the measure.

Domains, Segments, and Workbooks

RPASCE stores information in a persistent multidimensional data cache that is optimized for large volumes and dimensional or time series data access requirements, typically required by multidimensional solutions. This central repository is called a domain. The domain also includes central definitions of metadata for the solution and provides a single update point.

When you use an RPASCE solution, you interact with the solution through a personal data repository called a workbook. A workbook contains the subset of the data (and metadata) from the domain and its scope is constrained by the access rights available to a user. Workbooks are stored on the RPASCE server and can be built using an online wizard process or scheduled to be built in a batch process automatically. Workbooks are made up of one or more views. These views display the dimension and measure data of the domain.

Each workbook is built from a segment, which is a collection of position selections from multiple dimensions. A segment defines the dimensionality and range of a workbook and is created by the planner through a step-by-step wizard process.

Although the data and metadata in the workbook are copied from the domain, the data remains independent of the domain.

Domains can be built in one of two ways:

- Simple domain: This is the traditional, standalone domain that has no visibility to other domains.

- Global domain: This is a domain environment that contains two or more local domains (or subdomains) and a master domain that has visibility to all local domains that are part of that environment.

A global domain is a type of domain structure that provides users with the ability to view data from multiple domains and to administer common activities of an RPASCE domain and solution.

Using a global domain environment provides two primary functional benefits. The first feature allows users to have a global view of data in workbooks. Users can build workbooks with data from local domains, refresh global workbook data from local domains, save global workbooks, and commit the data from global workbooks to the individual local domains.

Local domains are typically organized, or partitioned, along organizational structures that reflect user roles and responsibilities. Most users only work within the local domains that contain their area of responsibilities, and they may not need to be know about the global domain environment. For performance and user contention reasons, global domain usage must be limited to relatively infrequent processes that require data from multiple local domains.

The other primary feature of global domain is centralized configuration and administration. Most of the mechanisms that are required to build and administer a domain have been centralized, and they must only be run in the master domain, which either propagates data to the local domains or stores the data centrally so that the local domains reference it in the master domain.

> **Note:** For a global domain environment to function properly, all local domains must be structurally identical.

Measure Data

In a global domain environment, measure data can be physically stored in two different ways:

- Across local domains

- In the master domain

Measure data that is stored in local domains is split across domains based on a predetermined level of a given hierarchy. This level is defined during the configuration process, and it is referred to as the partition level.

The base intersection of a measure (for example, what dimensions a measure contains) determines whether data is stored in the local domains or in the master domain. The data is stored in the master domain if the base intersection of a measure is above the partition level or if it does not contain the hierarchy on which the global domain environment is partitioned. This type of measure is referred to as a global domain measure or a higher base intersection measure.

Consider a global domain environment where the partition level is based on the Department dimension in the Product hierarchy. Data for measures that have a base intersection in the Product hierarchy at or below department are stored in the local domain based on the department to which the underlying position in the Product hierarchy belongs. Other hierarchies are irrelevant for this discussion.

However, measures that have a higher base intersection in the Product hierarchy than Department (for example, Division) or measures that do not contain the Product hierarchy (such as a measure based at Store-Week) cannot be split across the local

domains. These measures reside in the master domain and are accessed from there when these measures are required in workbooks.

All measures are registered in the master domain, and they are automatically registered in all local domains. RPASCE automatically determines where the measure must be stored by comparing the base intersection of the measure against the designated partition-level of the global domain environment. The physical locations of the measure data are invisible to the user after the measure has been registered.

RPASCE Client

The RPASCE Client is the web-based client for the RPASCE platform developed using the latest Oracle JavaScript Extension Toolkit (Oracle JET). It delivers an enhanced user experience for the RPASCE platform.

Planning is one of the most important and complex processes in a retail business. It typically involves a set of activities that must be followed as part of a workflow. The RPASCE Client includes an Activity Task Flow feature that provides a robust workflow that makes each planning activity easier to track and maintain.

Administrative Workbooks and Wizards

Using the administration workbooks, designated employees manage other employees' use of RPASCE. System administrators use the administration workbooks to perform the following tasks:

- Set up and maintain users and user groups. Note that authentication is handled externally through SSO or WebLogic/LDAP. The users in the external authentication system must exist in RPASCE so that user access can be managed.

- Manage user access to specific workbook templates and individual measures.

- Edit the contents of translation tables to support multiple-language use of the application.

Online Administration Tools

RPASCE Online Administration Tools (OAT) allow the application administrator to manage back-end operations on a domain from the RPASCE Client. OAT consists of two parts. The first part is a set of activities in the task flow that allows authorized users to launch and manage back-end processes from the client. The second part is a task status dashboard for an administrator to monitor the status of all the administration tasks after they have been submitted.

2

User Maintenance

User administration is the process by which administrators add or delete authorized system users, create or delete user groups, and edit user profiles. These tasks are performed through completion wizards on the User Administration tab.

The following procedures are discussed in this chapter:

- Adding a User

- Adding a User Group

- Deleting a User

- Deleting a User Group

- Editing a User

These procedures can be performed through the RPASCE Client by accessing the User Administration task in the Administration activity in the taskflow (Figure 2–1).

The usermgr online administration tool task is also described in this chapter. This task allows you to manage users and users groups through online administration wizards. For more information, see Managing Users Using User Manager.

Once users and user groups are set up, access permissions to workbook templates and measures within workbooks can be assigned through Security Administration. Security Administration also supports the modification of the label, the default workbook template, and the administration status associated with individual users.

> **Note:** Users must be set up in the directory for the external authorization provider as well. The users must also exist here to define access to and within each RPAS solution.

User Administration Activity

To access the User Administration workbooks:

1. In the taskflow, select the User Administration activity. The activity appears with all the subtasks.

Figure 2–1   User Administration
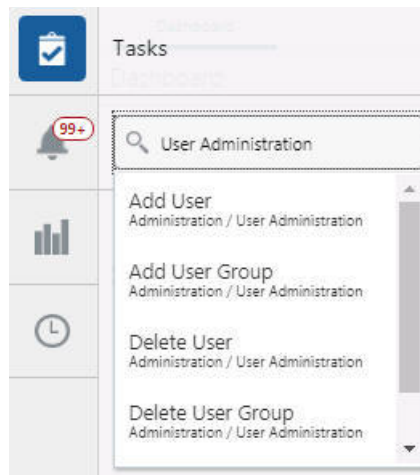
2. From the drop-down list, select the User Administration task you want to work on.

Adding a User

To add a user:

1. Under User Administration, click **Add User**. A Workbook Wizard window appears.

2. In the Workbook Wizard window, enter the relevant information in the following fields:

    ▪   **User name:** Type the user name that the user uses for logging on.

    > **Note:**   Each user name must begin with a letter and contain only alphanumeric characters and underscores. It cannot have spaces. User names are case sensitive.

    ▪   **User label:** Type a label that describes the user (for example, the user's full name). This identifying label appears in various locations throughout the application.

    ▪   **Default group:** Select the user group to which the user belongs.

    ▪   **Other groups:** If a user belongs to more than one group, select the additional groups from the list in the Other groups field.

    > **Note:**   Passwords are not used by the RPASCE Client. The password is defined and managed by your external authentication provider.

3. If the user requires Administration status, check the **Administrator** box.

> **Note:** Administrative users have special privileges and the read-only status may not apply to them.
>
> If you are not sure whether a user should be granted this ability, you can modify the Administration status later in the Users view (in the Security Administration workbook).

> **Note:** Granting users Administration status gives them access to all workbook templates, but it does not automatically give them access to all workbooks.

4. Select the **Lock user account** check box to temporarily disable the user's account.

5. Select the **Inherit group default rights** check box so that the user can inherit the default rights set for the user group via the Group Workbook Template Default Rights and Group Measure Default Rights worksheets. It the Inherit group default rights option is not selected and the Administrator check box is not selected, then the user is assigned Denied access rights to all templates and measures. If the Inherit group default rights option is not selected but the Administrator check box is selected, then the user is assigned Full Access rights to all templates and Read/Write rights to all measures.

6. Click **Finish** to add the new user to the database.

Workbook template and measure access rights can now be assigned to the user. To do this, access the Security Administration workbook. For more information, see Accessing Security Administration.

Adding a User Group

User groups provide an intermediate level of security to workbooks that were created and saved by specific users. When new users are assigned to the system, they must be assigned to existing user groups. User groups should consist of individuals with similar job functions or responsibilities. In the Oracle Retail Predictive Planning Suite, the user group corresponds to the user's planning role.

To add a user group:

1. Under User Administration, click **Add User Group**. A Workbook Wizard window appears.

2. In the Workbook Wizard window, enter relevant information in the following fields:

   ■ In the **Group Name** field, type a name for the group.

   > **Note:** Each group name must begin with a letter and contain only alphanumeric characters and underscores. It cannot have spaces. User group names are case sensitive.

   ■ In the **Group Label** field, type a descriptive label for the group. This label is displayed when referring to the group throughout RPASCE.

3. Click **Finish** to add the user group to the database.

Deleting a User

If a user profile is no longer needed, it must be deleted from the system in order to maintain system security. All the segments and the associated workbooks will be deleted.

> **Caution:**   Before you can delete a user, make sure that the user has logged off and has closed all the workbooks. If any of the user's workbooks are open, the user will not be deleted from the system.

To delete a user:

1. Under User Administration, click **Delete User**. A Workbook Wizard window appears.

2. The Workbook Wizard window displays the user names and labels for all users. Select the name of the user to delete.

3. Click **Finish** to delete the user from the system.

Deleting a User Group

If a user group no longer exists, the group must be deleted from the system as soon as possible to maintain system security.

> **Caution:**   Before you can delete a user group, all users must be removed from the group. For each user in the group, you must either delete the user or change the default user group assignment for the user.

To delete a user group:

1. Under User Administration, click **Delete User Group**. A Workbook Wizard window appears.

2. In the Workbook Wizard window, select the user group you want to delete.

3. Click **Finish** to delete the user group from the system.

Editing a User

To edit a user's profile:

1. Under User Administration, click **Edit User**. A Workbook Wizard window appears.

2. The Workbook Wizard window displays the user names and labels for all users. Select the user you want to edit and click **Next**.

3. Make the necessary changes to the user's profile. You can change anything except the user name. For more information on the fields, see Adding a User.

4. Click **Finish** to save the changes.

> **Note:** If the administrator privilege is withdrawn from a user who is currently an administrator and if that user was in the middle of any tasks that requires the administrator privileges, then those tasks will fail.
>
> If the lock user account option is set on a user who is currently logged-in to the domain, then that user's active sessions will expire as soon as the edit user task finishes executing.

Managing Users Using User Manager

Use the User Manager online administration task to add users, remove users and groups, and edit user and group information.

To use User Manager:

1. Under Online Administration Tools, click **Submit a New Admin**. A Wizard window appears.

Figure 2–2   Submit New Admin Task
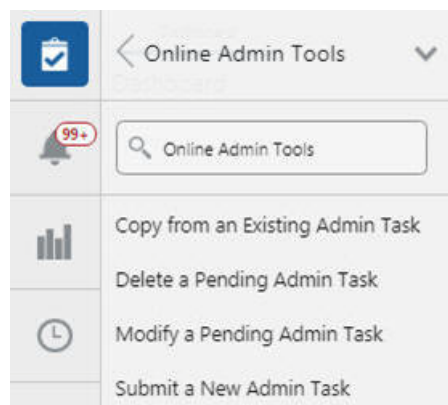
2. The Workbook Wizard window displays the Task Group. Select **System Admin Tasks** and click **Next**.

Figure 2–3   System Admin Tasks
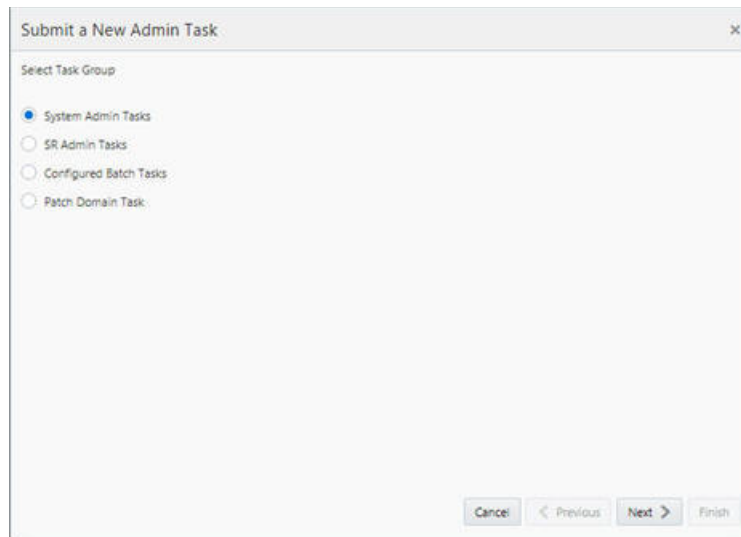
3. Select **User Manager** and click **Next**.

Figure 2–4   User Manager
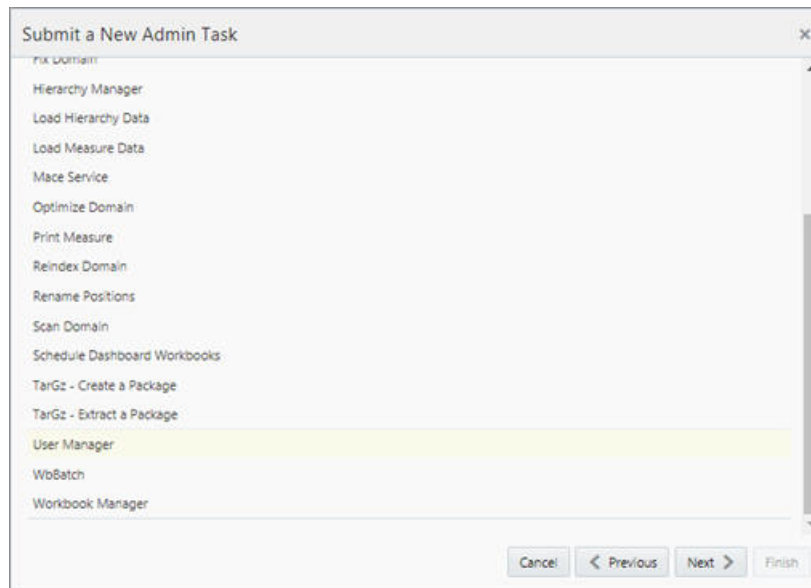
4. Choose any option of your choice, for example, **Add a new user**. Provide a label to the task and click **Next**.

Figure 2–5   Add a New User

5. In the next window provide the user label and click **Next**. To add a new user, choose the option **Run ASAP** and click **Next** and **Finish**.

Figure 2–6   Enter User Label

6. To add a user, select the option **Run ASAP** and click **Next**.

Figure 2–7   Run ASAP

7.   Verify the results and click **Finish**.

Figure 2–8   Verify and Confirm the Results

8.   Once the notification is available, you can view the status/log file under Online Administration Dashboard by clicking on the task label.

Figure 2–9   Notifications
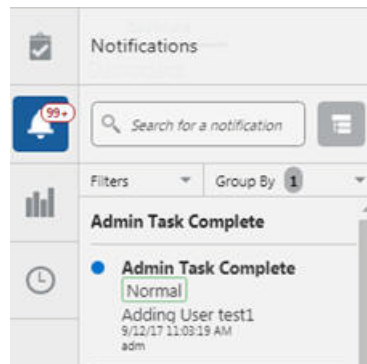
9. View the Online Administration Dashboard by navigating to the Administration task flow and clicking **Task Status Dashboard Sub Task**.

Figure 2–10   Task Status Dashboard
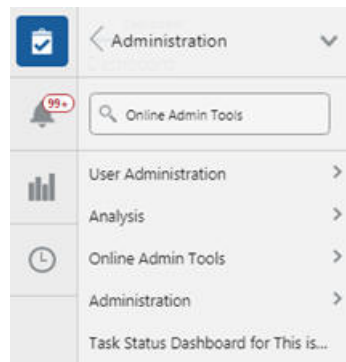
10. You see the task status.

Figure 2–11   Task Status
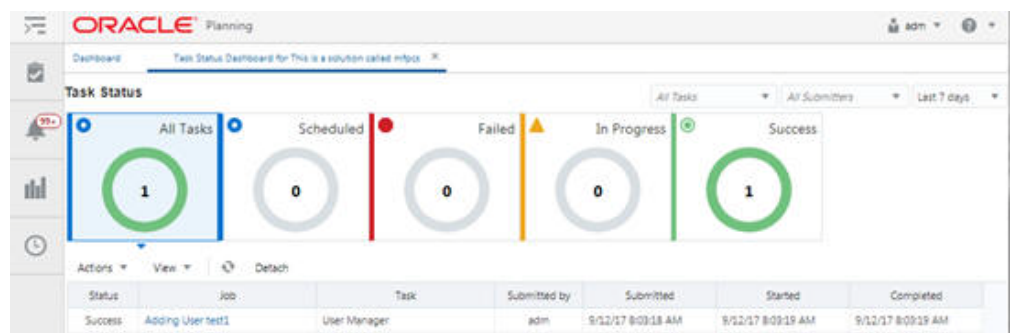
Table 2–1 provides descriptions of the arguments used by the User Manager online administration task.

Table 2–1   User Manager Online Administration Task Arguments

Argument	Description
Add a new user	Adds a user with a specified name.
	This argument has a switch and requires a Label and User Group in the next wizard page. Type a label that describes the user (for example, the user's full name). This identifying label appears in various locations throughout the application. Select a user group to which the user belongs.
	Check box Administrative user specifies that the user being added to the domain has administrative rights.
Add a user group	Adds a group with a specified name.
	Provide a label to the group in the next wizard page.
Remove a user	Removes the user with the specified name from the domain.
Remove a user group	Removes a group with this group name.
Remove all users with this label	Removes all users with this label.
List all user groups and users	List all users registered to the specific domain.
Lock a user	Locks the specified user. This prevents the user from logging into the domain.
Unlock a user	Unlocks the specified user. This allows the user to log into the domain
Print detailed information about a user	Prints the specified information about a user.
Print detailed information about a group	Prints the specified information about a group.

> **Note:**   The arguments userName and groupName must conform to the standards for position names in RPASCE. They must contain only the characters a-z, A-z, 0-9, _, &, $ or %. User and group names cannot begin with an underscore. Any supplied upper case letters will be converted to lower case by the application. User and group names cannot be an empty string.

3

System Administration

This chapter describes the following system administration workbooks and wizards:

- Hierarchy Maintenance Workbook
- Security Administration Workbook
- Measure Analysis Workbook
- Format/Users/Security Export Task
- Formats/Users/Security Import Task
- List Logged-In Users
- Apply Format
- Schedule Dashboard Workbooks

These workbooks and wizards are found in the Administration task in the taskflow.

> **Note:** The Translation Administration workbook is described in Internationalization.

Hierarchy Maintenance Workbook

RPASCE can be used to set up and maintain user-named and user-defined dimensions within hierarchies. Hierarchy Maintenance is the means by which custom-created dimensions within a hierarchy can be established and maintained through the application interface in order to meet individual business needs.

The configurator can define the dimensions and hierarchical structures specific to the customer's organization. For example, the system can be built to recognize that SKUs roll up into styles, styles roll up into product classes, and so on, within the product hierarchy. Occasionally, you might want to group products according to some ad hoc personal design to suit a particular business need. You can group arbitrary items in a hierarchy to use in functions such as forecasting, replenishment, and measure analysis. These user-defined groupings act as normal dimensional levels. In other words, they allow the user to roll data up from lower levels of aggregation along the hierarchical paths that the user defines.

For example, suppose experience has shown that the accuracy of forecasts for your top 50 products (A products) reflects the relative accuracy of all forecasts. Therefore, you would like to group elements within a user-defined dimension as the top 50 products by designating them "A Products." Then, when you select products in a wizard or look at data in a view, you can change the rollup to your user-defined dimension to see your top 50 products grouped together.

> **Note:** Your collection of 50 products may comprise elements from a wide range of product classes or departments, and your grouping scheme may have little to do with the normal dimensional relationships of these items in the product hierarchy.
>
> The group of items you designate as "A Products" may change over time as consumer preferences change. From this example, you can see that user-defined dimensions can be used to create any ad hoc groupings to provide additional support in analyzing, selecting, or summarizing data in demand forecasting. The Hierarchy Maintenance interface allows you to change the nature of the groupings as required.
>
> The number and names of user-definable dimensions are set by your company when an RPASCE-based solution is initially installed. The positions within each dimension and their associated labels can be altered and maintained through the hierarchy maintenance process.

Remember that any hierarchy in RPASCE can have user-defined dimensions within it as long as they are set up by your company at the time of installation. The examples in this section refer to the product hierarchy, but other hierarchies can be maintained in the same way.

Hierarchy Maintenance Example

Suppose you want to designate SKUs in your product hierarchy as either A, B, or C products so that you can group these items together when you view information such as forecasting, replenishment, or measure analysis reports.

To do this, you must maintain a user-defined dimension that allows you to map the SKUs to the various positions of your classification scheme (A, B, or C). The user-defined dimension used in the following example is named "Product Status." To maintain this user-defined dimension, use the Hierarchy Maintenance Wizard.

Hierarchy Maintenance Wizard

To access the Hierarchy Maintenance Wizard:

1. On the main page, click **Tasks**, then select **Administration**. The Administration group appears

Figure 3–1   Tasks
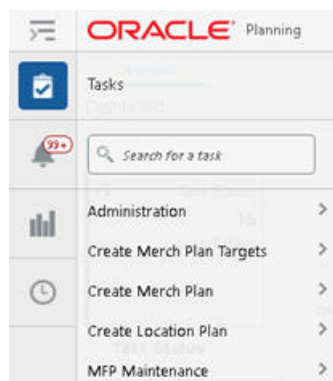
2. In the Administration group, select **System Administration**.

Figure 3–2   System Administration
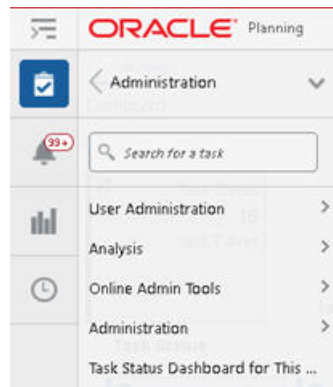
3. In the System Administration group, select **Hierarchy Maintenance**. A plan dialog box appears.

Figure 3–3   Hierarchy Maintenance
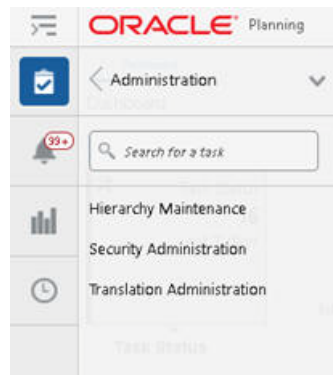
4. Click **Create New Plan** to create a new segment.

Figure 3–4   Create New Plan
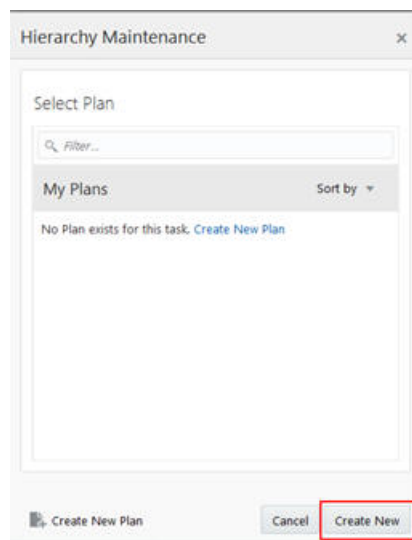
5. Provide a label for the segment.

Figure 3–5   Label

6. A wizard page appears to select the custom dimension to maintain. Select a custom dimension and click **Next**.

7. Select the positions to be included into the segment and click **Finish**.

8. The RPASCE server processes the request asynchronously. Upon completion, a notification will be sent to the user interface and a link to open the segment will be available in the Recent Plans box on the right.

Figure 3–6   Recent Plans
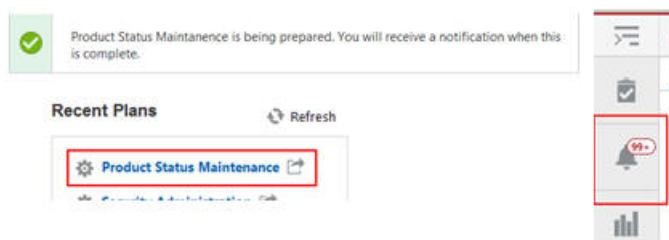
9. Click the link to open the segment.

10. In the position assignment field for the custom dimension, assign a value to each product or location position in the workbook. Enter any text string in a cell. Each unique string is treated as a separate user-defined position within the custom dimension.

11. After you make your selection, for the changes to take effect, click **Calculate** and then **Commit**.

12. To close the segment, click the **X** icon next to the segment label on the top of the page. The segment is saved automatically.
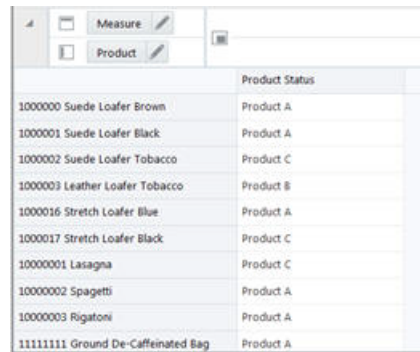
Hierarchy Maintenance View

The Hierarchy Maintenance view displays the position assignment fields for the selected custom dimension. Edit the cells associated with the custom dimension as required.

Returning to the example dimension Product Status: You want to classify each selected SKU in your segment as either a Product A, a Product B, or a Product C. This example provides only three positions, or values, in the Product Status dimension; however, you can enter any character string in an individual SKU's Product Status cell. This new string is treated as a separate user-defined grouping. If this is the first time a particular SKU has been mapped to the Product Status dimension, the label assigned to that SKU will not yet be defined. The Product Status field is automatically filled with "Unassigned."

Assign labels to each product with regard to the Product Status dimension. In the following example, products that were previously Unassigned are now designated as Product A, B, or C.

Figure 3–7   Hierarchy Maintenance View
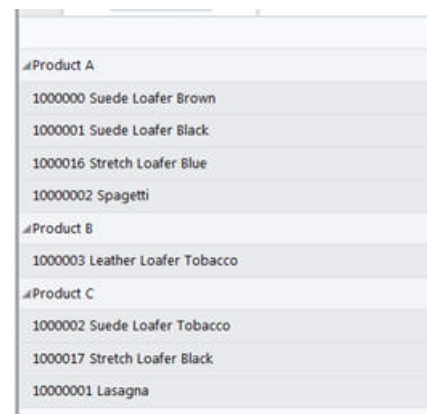
> **Note:**   The RPASCE system is case sensitive when a new position label is entered in the Hierarchy Maintenance segment. After the segment is committed, the typing of the group name is not case sensitive. For example, Product B can later be entered as product b after the Product B group label has been committed.

After making the Product A, Product B, or Product C designations for the selected SKUs, you must calculate and commit the segment for any changes to take effect.

For this example, labels have now been assigned to the various positions within the Product Status dimension, and selected products in the product hierarchy have been classified with regard to the custom dimension. Demand Forecasting treats Product Status, a user-defined dimension, as a normal dimensional level within the product hierarchy.

Figure 3–8 displays the results when you access a quick menu in a wizard and change the rollup to the Product Status dimension. The products shown here are classified according to the position values (Product A, Product A, or Product C) that were assigned while maintaining the Product Status dimension.

Figure 3–8   Product Status Dimension Results

Maintaining a User-Defined Dimension Within a Hierarchy

Use this procedure to assign product or location items to custom-defined positions within a specialized dimension. Custom-created dimensions are distinct from those in the standard hierarchical roll-ups configured in the system implementation. You can use these dimensions in the same way that you use normal Demand Forecasting levels, aggregating data along these new hierarchical paths.

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Hierarchy Maintenance**.

Figure 3–9   Hierarchy Maintenance Task
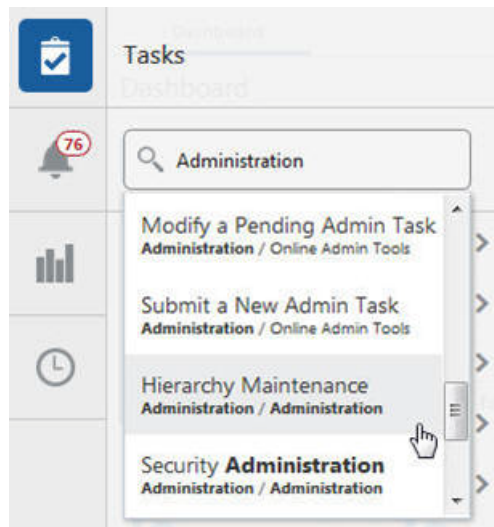
3. The segment dialog box opens. Select **Create New Plan** and enter the label of the segment. A Hierarchy Maintenance Wizard window appears.

Figure 3–10   Plan Label
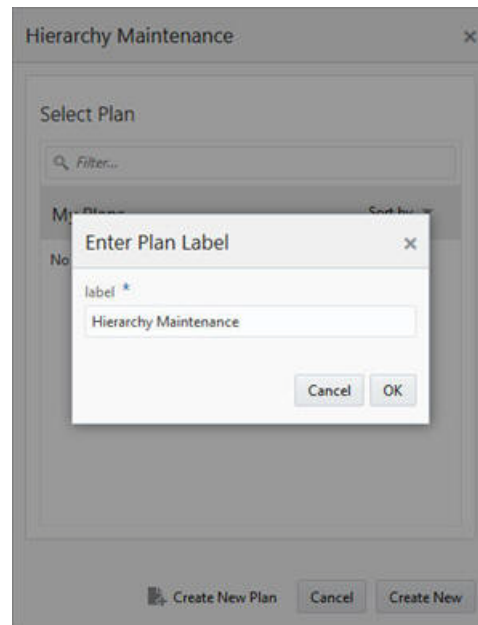
4. Select the hierarchy to specify a user-defined dimension (for example, Product or Location). Only the hierarchies that have been set up to contain user-defined dimensions appear here.

5. Click **Next**.

6. Select the user-defined dimension to be updated. The number and names of available custom dimensions are set at installation.

7. Click **Next**.

8. From the Available Items in the selection wizard, select the items to be mapped to positions within the custom dimension.

9. Use the Add, Add All, Remove, and Remove All buttons or the drag and drop action to move your selection to the Selected Items section.

10. After all items to appear in your workbook have been selected, click **Finish**. The Hierarchy Maintenance segment is created with the given label and submitted for building workbook. After the notification is provided, open the workbook.

11. In the position assignment field for the custom dimension, assign a value to each product or location position in the workbook. Enter any text string in a cell. Each unique string is treated as a separate user-defined position within the custom dimension.

12. After you make your selection, for the changes to take effect, click **the Commit** (F8) icon on the bottom right corner. Commit notification will be sent once the commit is successful.

13. To close the workbook, click **X** on the top-right corner of the tab.

14. The Hierarchy Maintenance workbook is displayed. In the position assignment field for the custom dimension, assign a value to each product or location position in the workbook. Enter any text string in a cell. Each unique string is treated as a separate user-defined position within the custom dimension.

Security Administration Workbook

The security model in RPASCE includes workbook templates, workbooks, measures, and positions. The levels of security are defined as measure level, position level, and workbook level.

Security Overview

This section provides the basic information on the security model in RPASCE.

User and Group Security

In the RPASCE Client, user authentication is handled externally through SSO or WebLogic/LDAP. The users in the external authentication system must be added to the RPASCE domain so that user access can be managed.

User accounts can be marked as locked by the domain administrator. This prevents the user from logging on with the RPASCE Client. The account remains locked until the administrator re-enables the account. Account lockouts can be set or cleared by the domain administrator by using the Manage Users OAT task or the Edit User activity in the User Administration workbook.

User groups provide an intermediate level of security to positions, measures, and workbooks. If access to an item is denied to a user group, none of the users in that group have the permission to access that item. User groups must consist of individuals with similar job functions or responsibilities. In the Oracle Retail Predictive Planning Suite, a user group generally corresponds to the user's planning role.

When new users are added to the system, they must be assigned to existing user groups, including one default group and, optionally, multiple other groups.

Measure Level Security

Measures have access rights; these are read-write, read-only, or denied. Measures that are read-write or read-only may be selected in the extra measures and insert measure dialog boxes. RPASCE ensures that read-only measures are not editable by the user, and the presence of read-only measures does not affect the ability to commit a workbook.

Measure security can be specified and changed through the Security Administration workbook. The Measure Rights view allows read-only, deny, or read/write access to a measure to be specified for each user.

A workbook template can override the security of a measure, but it can only narrow the security of that measure. For example, a measure can have read-write access for a user and a template can specify that all users have read-only access to the measure when a workbook is built. However, if the measure security is read-only, the template cannot expand the security of that measure to read-write. Measures that are explicitly made read-only by a workbook template are not expanded to read-write access by RPASCE.

> **Note:** Refer to the *Oracle Retail Predictive Application Server Cloud Edition User Guide* for more information on the Measure Analysis workbook.

Position Level Security

Position level security allows access control for dimensions on a position-by-position basis. This capability is completely optional. If position level security is not explicitly defined and configured, all users in a domain have access to all positions in all hierarchies. After the position level security is defined, access to a position can be granted or denied for individual users, users in a group, or for all users.

Position level security can be defined at levels (dimensions) at or above base (such as class in the product hierarchy) in any hierarchy other than calendar. As positions are added at a level/dimension lower in the hierarchy than where the position level security is maintained, access to those positions is automatically granted if a user has access to the parent position. For example, if security is maintained at the subclass level, users are automatically granted access to all the SKUs in a given subclass if they have access to that subclass. This includes those that were added after security was established.

Exactly one dimension in each hierarchy can be defined as the security dimension for the hierarchy. If a security dimension is defined for the hierarchy, all dimensions in the hierarchy have position level security enabled, but position security is set at or above the designated dimension. For example, if the class dimension is designated as the security dimension, an administrator can maintain access to positions in the class dimension or at any level above class.

To specify the security dimension for a hierarchy, use the hierarchyMgr utility.

After a security dimension is defined for a hierarchy, all users in the domain default to having access to all positions in any dimension in the hierarchy. Additionally, users automatically have access to newly added positions to a domain. Views in the Security Administration workbook are used to control position access for individual users, user groups, or all users (referred to as world or default access). Three views are provided in this workbook for each hierarchy with a defined security dimension. The default view controls access to positions for all users (for instance, Prod Security Default); one view controls access to positions by user group (for instance, Prod Security Group); and the last view controls access to positions by individual users (for instance, Prod Security User).

Access must be granted at all levels for a user to have access to a position. This means that a position must have a value of true at the levels default/world, group, and user. Table 3–1 demonstrates how access is granted or denied based on all combinations of settings.

In the table, security is set by Position. Denied = False and Granted = True. Based on the settings for User, User Group, and World, the user is either granted or denied access, as shown in the Resulting Access column.

Table 3–1   Grant or Denial of Access by a Combination of Settings

User	User Group	World	Resulting Access
Denied	Denied	Denied	Denied
Denied	Denied	Granted	Denied
Denied	Granted	Denied	Denied
Granted	Denied	Denied	Denied
Denied	Granted	Granted	Denied
Granted	Denied	Granted	Denied
Granted	Granted	Denied	Denied

Table 3–1   (Cont.) Grant or Denial of Access by a Combination of Settings

User	User Group	World	Resulting Access
Granted	Granted	Granted	Granted

Position level security is used when a user selects positions in the wizard process before building a workbook. Only positions to which a user has access are available for selection in the 2-tree, which are then included in the build of the workbook.

Note that position level security, when used for a global domain environment on the same dimension on which it is partitioned, is used to guide the user to the domain or domains that the user has access to. If the user only has access to positions within a single local domain, that user will be guided there on New Workbook. If the user has access to more than one, that user will be asked and can choose based on partition-level positions.

Similarly, Open by default only lists workbooks from those domains, and the user is only shown alert counts from those domains.

Workbook Security

Workbook access is either granted or denied. If users have been granted access to a workbook, they can open, modify, and commit the workbook. No distinction is made between read-write-commit, read-write, and read-only access.

Workbook access is automatically granted to the user that built it.

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

Users with administrator status automatically have access to all workbook templates.

Another aspect of workbook security is the ability to set limits for the number of workbooks that a user can have saved at any given time. Limits can be set for a user per template, for a user group per template, or for a template for all users. The limits are evaluated in the above order, which means that a limit defined at user-template overrides any values defined at group-template or template. If the above limits are not defined, the default value is one billion.

The limits are checked when the workbook build process is initiated. When the limit is reached, an error message displays informing the user that the workbook build process cannot complete because the limit has been reached. The message also lets the user know what that limit is.

Administrative users have full access to all workbook templates, regardless of the access rights that other administrative users may assign to them in the Security workbook. The administrative user can build the Security workbook to change the access right back, so the nominal assignment does not matter for administrative users.

Non-administrative users do not have access to the Security template and User Administration template groups even if the administrator inadvertently assigns them access rights.

Security Administration Workbook

The Security Administration Workbook is only available to system administrators. After users and user groups are created, the administrator may set up and maintain access permissions to workbook templates and measures within those workbook templates. This workbook allows the administrator to determine which templates individual users can access, as well as the measures that users can access while manipulating workbooks in the system. The user can also specify and restrict the measures that are available to be added to a given workbook template. Setting access permissions in this way provides a high degree of measure security because users can be restricted to viewing and editing only certain relevant measures.

All administrative users have full access to all workbook templates, regardless of the access rights that they have been assigned in the Security workbook by other administrative users. The administrative user can build the Security workbook to change the access right back, so the nominal assignment does not matter for administrative users.

The Security Administration workbook has the following views:

- Workbook Template Rights View

- Workbook Template Measure Rights View

- Measure Rights View

- Dimension Modification Rights View

- Position Level Security View

- Workbook Template Limits Views

- Max Domain Session Limit View

- Max User Session Limit view

- Group Workbook Template Default Rights View

- Group Measure Default Rights View

Security Template Administration also allows the administrator to modify the label, Admin status, and default workbook template associated with each user. The administrator can also access this workbook template to modify the labels associated with user groups, workbook templates, and workbook template groups. Using this workbook, the administrator can:

- Assign and modify the access rights of each user to all workbook templates. User/template permissions are set in the Workbook Template Rights view.

- Determine which optional measures are to be accessible through individual workbook templates. Template/measure permissions are set in the Workbook Template Measure Rights view.

- Assign and restrict user access to individual measures. User/measure permissions are established in the Measure Rights view.

Workbook Template Rights View

The Workbook Template Rights view is used for setting and maintaining the access permissions of each user to specific workbook templates.

The view contains a drop-down list for each available workbook template and user combination. To grant a user access rights to a workbook template, select one of the following options from the drop-down list for that workbook template:

- Denied

- Read-Only

- Full Access

After a user's profile is changed, those changes must be committed to the database in order for them to take effect.

The Read-Only permission on a template applies only to actual workbooks created by the template. For templates that do not generate a workbook, but only run through a wizard process for other purposes, the Read-Only permission for a user on that template does not prevent the user from running through the wizard. This applies to standard RPASCE templates, such as Add User and Delete User, but it may also apply to various application-specific templates.

Workbook Template Measure Rights View

The Workbook Template Measure Rights view allows administrators to determine which registered measures are available for optional inclusion in newly built workbooks.

When a measure is initially registered as a public measure, all templates default to having access to that measure. This means that it is possible for this measure to be added to a workbook template, even if it is not one of the standard measures displayed when a workbook of that type is built. Some new workbook wizards include a dialog box that prompts users to select any additional measures to be included in the workbook build. By default, all newly registered measures are included on this list of available additional measures. The other method of inserting new measures into a workbook is through the Insert Measure command.

The Workbook Template Measure Rights view is used to modify template/measure permissions, which allow only certain templates to optionally include specified measures in new workbook builds.

This view contains a check box for each available workbook template and registered measure combination.

Measure Rights View

The Measure Rights view allows the administrator to restrict user access to individual measures on a user-by-measure basis. User/measure permissions are initially determined by the system by integrating the current user/template and template/measure settings and applying the following rule:

"A user cannot have access to any measure that is not available in at least one template to which the user has access."

Permissions can be made even more restrictive on a user-by-measure basis by using the Measure Rights view to deny users access to measures that they would normally be permitted to edit.

The view contains a drop-down list for each available user and registered measure combination. Three security options are available: Denied, Read-Only, and Read/Write. Denied prevents the user from viewing data. Read-Only allows the user to view the data. Read/Write allows the user to edit data values. However, a commit rule must be configured for a measure for data to be committed to the RPASCE data store.

A measure has the security rights it had when it was inserted in the workbook. The change in measure security rights is only reflected in new workbooks when that measure is inserted.

> **Note:** If a measure that has dependent measures is inserted into a view, those dependent measures will also be inserted into the view. If the dependent measures have denied measure access, they are still inserted into the view, but they are hidden by default. These measures can be made visible in the Show/Hide dialog box.
>
> The Measure Rights view contains only public measures, that is, measures that can be optionally included in a view, depending on choices made in a new workbook wizard. Measures that are registered as private measures do not appear in this view. If there are no public measures available to be displayed in this view, the view will not be built.

Dimension Modification Rights View

The Dimension Modification Rights view allows the administrator to determine which dimensions, if any, a user can modify. The view contains a check box for each available user and dimension combination. A check mark in the cell indicates that the user is permitted to modify the specified dimension.

After changes are made to a user's dimension modification rights, they must be committed before they take effect.

Position Level Security Views

The position level security views are used to grant or deny access to positions for individual users, user groups, or all users. Position-level security is set for a specific dimension of a hierarchy (other than calendar).

For each hierarchy/dimension that has position-level security enabled (normally just a single hierarchy/dimension), there are three views, one each for user, user group, and world/all users.

After changes are made to position level security, they must be committed before they take effect.

Workbook Template Limits Views

The Workbook Template Limit views are used to limit the number of workbooks that the user can have saved. Limits can be set for a user per template, for a user group per template, or for a template for all users. The limits are evaluated in the above order, which means that a limit defined at user-template overrides any values defined at group-template or template. If the above limits are not defined, the default value is one billion, but it is not displayed in the workbook.

The limits are checked when the user begins the segment build process. If the limit has been reached, an error message appears that informs the user that the workbook build process cannot complete because the limit has been reached.

Max Domain Session Limit View

The Max Domain Session Limit view is used to limit the number of user sessions that can be attached to a single domain by all users of that domain. The limit is set at the domain level.

This limit is checked during user login. If the limit has been reached, an error message appears to inform the user that the login has failed because this limit has been reached.

Max User Session Limit View

The Max User Session Limit view is used to limit the number of concurrent user sessions that can be attached to a single domain by the same user at the same time. The limit is set per user so that the administrator can control the maximum number of concurrent sessions that are allowed for an individual user.

This limit is checked during user login. If the limit has been reached, an error message appears to inform the user that the login has failed because this limit has been reached.

Group Workbook Template Default Rights View

The Group Workbook Template Default Rights view is used for setting and maintaining default access permissions for each user group for specific workbook templates. The worksheet contains a drop-down list of available rights for each available workbook template and user group combination.

To grant a user group access rights to a workbook template, select one of the following options from the drop-down list for that workbook template: Denied, Read Only, or Full Access. After you change a user group's profile, you must commit the changes to the database for them to take effect. If the -inheritGroupDefaultRights flag is set to true when the user is created, then the user inherits the default rights for the various workbook templates that are defined here for the user group.

Group Measure Default Rights View

The Group Measure Default Rights worksheet is used by the administrator to define the access that each user group can have for each measure. The worksheet contains a drop-down list of available rights for each available user group and registered measure combination. Three security options are available: Denied, Read-Only, or Read/Write. After you change a user group's profile, you must commit the changes to the database for them to take effect. If the -inheritGroupDefaultRights flag is set to true when the user is created, then the user inherits the default rights for the various measures that are defined here for the user group.

Using the Security Administration Workbook

> **Note:** These tasks are performed through the Security Administration workbook. This workbook is only available to system administrators.

Accessing Security Administration

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

Figure 3–11   Security Administration Task
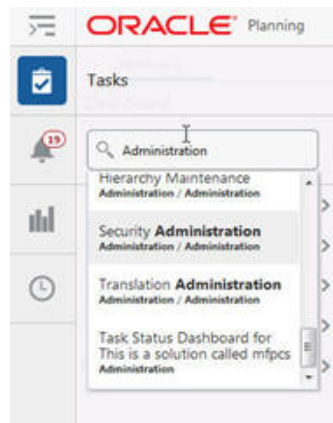
3. A new segment is created and a notification is issued. If the segment already exists, the existing workbook associated with the segment is re-opened.

Figure 3–12   New Segment Notification

You can also open the segment from recent plans.

Figure 3–13   Last Opened Recent Plan

Setting or Modifying Users' Access to Workbook Templates

The Workbook Template Rights view enables you to set one of the following access privileges to each workbook template for a user:

■   Denied

■   Read-Only

■   Full Access

To set or modify a user's access to workbook templates:

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Workbook Template Rights view, for each workbook template, select one of the access privileges for the users you want.

5. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is done.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Setting Measure Availability for Workbook Templates

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Workbook Template Measure Rights view, select each registered measure that must be available for inclusion in the associated workbook template. For measures that must not be included in the associated template, make sure there is no check mark.

5. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is done.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Assigning or Restricting User Access to Measures

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Measure Rights view, for each measure that a user should have access to, select Read-Only or Read/Write from the drop-down list in the cell. For measures to which the user should not have access, make sure Denied is selected.

> **Note:** The drop-down list appears when you double-click inside the cell.

5. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is completed.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Changing a User's Ability to Modify Dimensions

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Dimension Modification Rights view, select each dimension for which the user needs modification rights. For dimensions that the user should not be able to modify, make sure there is no check mark.

5. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is completed.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Setting or Modifying Access to Positions (if Position Level Security has been Enabled)

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Position Level Security view, select the view for which security must be set or modified: User, User Group, or World.

5. By default, the dimension (level) at which position level security is enabled appears. To manage security at a level above the designated level (only levels above are possible), right-click and select **Rollup** to view the available dimensions.

6. To grant access to a position, select the check box in the relevant cell.

   > **Note:** A user must have access at the User, User Group, and World levels to have access to a position.

7. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is completed.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Limiting the Number of Workbooks that a User Can Save

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Workbook Limit Per User/Template view, for a workbook template, set an appropriate value as the limit for a user.

5. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is completed.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Limiting the Number of Workbooks Per Template

1. In the taskflow, search for Administration activity. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created and a notification is issued. If the segment already exists the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Workbook Limit Per Template view, for a workbook template, set an appropriate value as the limit.

5. After you make your selection, for the changes to take effect, click the **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is completed.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Limiting the Number of Workbooks that a Group Can Save

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Security Administration**.

3. A new segment is created, and a notification is issued. If the segment already exists, the existing workbook associated to the segment is re-opened. You can also open the segment from recent plans.

4. On the Workbook Limit Per Group/Template view, for a workbook template, set an appropriate value as the limit for the user group.

5. After you make your selection, for the changes to take effect, click **Commit** (F8), located under the bottom right corner. A notification is issued once the commit is completed.

The data modified is auto saved.

To close the workbook, click **X** on the top-right corner of the tab.

Measure Analysis Workbook

The Measure Analysis workbook template allows the user to view data associated with any registered measure in the RPASCE applications, such as actual sales data for specified product/location/calendar combinations. The user may also use the Measure Analysis workbook to edit values for writable measures; however, commit capability is only allowed to administrative users.

Although a common use of the Measure Analysis workbook is to view actual sales data, the workbook is not restricted to presenting sales data alone. The user can view any data loaded into the RPASCE master database, such as selling prices, shipments, and orders. The Measure Analysis Wizard provides a list of all stored measures that have an insertable measure property set to true. The user simply chooses the measures to be displayed in the new workbook.

> **Note:** Due to its dynamic nature, formatting settings cannot be saved in the Measure Analysis workbook.

Measure Analysis View

The Measure Analysis view allows the user to view the chosen measure data for the positions selected from the measure's associated hierarchies. Each Measure Analysis view is displayed at a different dimensional intersection, depending on the measure selections made in the wizard. This dimensional intersection is shown in the view title bar.

Figure 3–14   Measure Analysis View Example

Figure 3–14 shows a Measure Analysis view that displays Weekly Sales data for several items in a particular store. The location/product/calendar dimensional intersection of this view, as shown in the title bar, is STR (Store), ITEM, WEEK. The Weekly Sales measure, because it is registered as a read/write measure, can be edited in this view. However, only an administrative user can commit overwrites to writable measures in this workbook.
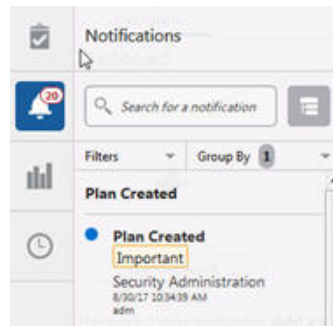
Accessing Measure Analysis

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Measure Analysis**.

Figure 3–15   Measure Analysis Task
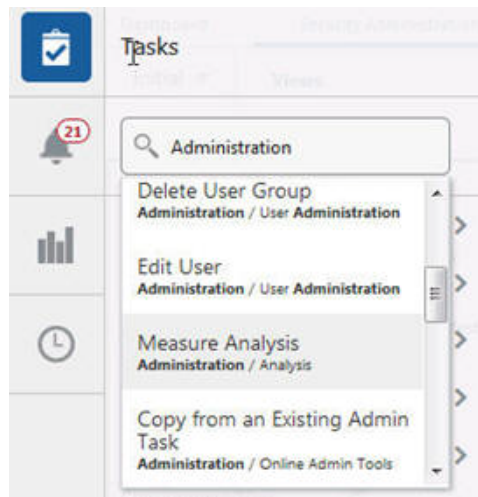
3. The segment dialog box opens. Select **Create New Plan** and enter the label of the segment.

Figure 3–16   Plan Label
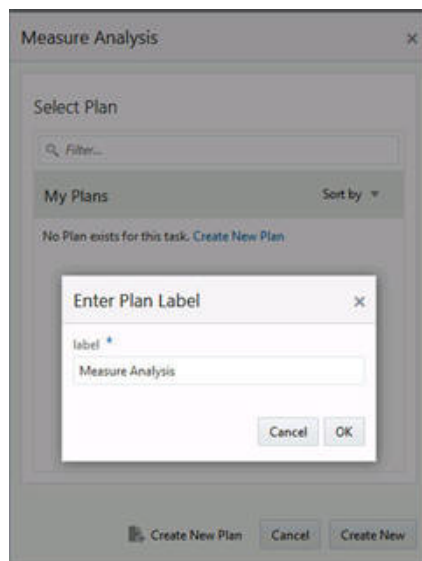
4. A Measure Analysis Wizard window appears. Select the respective positions and click **Finish**.

Figure 3–17   Plan Created
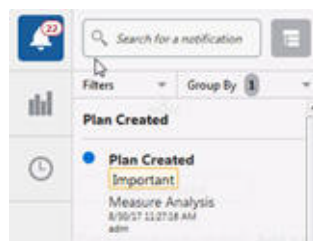
5. A segment is created, and the user will receive a notification once the segment is built and can open it from recent plans.

Figure 3–18   Recent Plans

Reviewing and Editing Sales or Other Registered Measure Data

1. In the taskflow, select **Administration**. The Administration activity appears with all the subtasks.

2. Click **Measure Analysis**.

3. The segment dialog box opens. Select **Create New Plan**, and enter the label of the segment. A Measure Analysis Wizard window appears.

> **Note:**   To open an existing Measure Analysis workbook, users can open the segment dialog box from step 3, and then go to step 8.

4. In the Workbook Wizard window, select the measures you want to include in the new workbook. Use **Ctrl-Click** or **Shift-Click** to select multiple measures.

5. Click **Next**. Based on each hierarchy specified in the base intersection of the selected measures, a hierarchy wizard appears.

6. In the hierarchy wizard, select the relevant positions and then click **Next**. Repeat this step for each hierarchy wizard.

7. Click **Finish**. A segment is created, and the user will receive a notification once the segment is built and can open it from recent plans.

8. On the Measure Analysis View(s), view the stored data associated with the measures and hierarchy positions selected in the wizard. Make any changes as required. As an administrator, you can commit the changes.

Format/Users/Security Export Task

If the customer wants to migrate the security and format data, use the Online Administration Task Formats/Users/Security.

The OAT task extracts the following information from the domain:

■ All user and group information

■ Workbook template formats at the template, group, and user levels for the master domain only

■ The following security settings:

  – Dimension Modification Rights

  – Max User Session Limit

  – Group Measure Default Rights

  – Group Workbook Template Default Rights

- Max Domain Session Limit
- Measure Rights
- Workbook Template Limits
- Workbook Template Measure Rights
- Workbook Template Rights
- Position Level Security (all position security related measures)

All exported data is saved in an archive file named FormatSecurityExport.tar.gz and placed in the Outgoing FTP location.

The option is: Verbose mode. Output more detail on the operations.

Formats/Users/Security Import Task

The task Formats/Users/Security - Import in OAT can be used to import user, group, security, and workbook format data. The data archive file FormatSecurityExport.tar.gz created using the Formats/Users/Security - Export Task earlier must be first copied to the Incoming FTP location.

If changes have been made to the styles and formats when the domain configuration is updated and the workbook formats are imported on the template level using the OAT task, the domain must be patched afterwards using the same configuration for creating the domain. This is necessary in order to merge the changes in the configuration into the styles and formats in the domain that has been overwritten during the importing process.

The options are:

- Verbose mode. Output more detail on the operations.
- Import everything, including formats, user/group, and security data.
- Import essential components.
    - Import user/group data. All users and groups that exist in the importing set will be updated or added to the target domain. Users and groups that exist only in the target domain will not be changed.
    - Import security data. Position level security is completely reset if the secure dimension is different from the import.
    - When importing security data, replace (RPL) other than override (OVR). If the target domain has extra user than the import set, RPL mode will wipe clean all security measures for those users. OVR mode will only overlay the import security on top of current security.
    - user,group,template. Import user, group, or template level formats to the target domain. It can be a combination of any of the three levels. The format saved at the user/group/template level can only be imported at the level at which it was originally saved. For example, it is not possible to save the format at the user level and import that format at the group level in this step.

List Logged-In Users

The List Logged-In Users task can be used by the administrator to list all the users logged into the domain. This OAT task also lists the number of active sessions per user at a given point.

Apply Format

In RPAS, the data is split into local domains based on the partition dimension. If the user is building the workbooks with more than one position from the partition dimension, the workbook resides in the master domain. Any changes done to the formats are also saved in the master domain workbook.

Users making format changes in a workbook can save it to the template level (which will be available in future releases) from the UI. In this way, other users who are building the same template in another local domain can also use them. In order for the format changes to reflect all the local domain workbooks, when building the workbook the Apply Format OAT task must be executed.

The Apply Format OAT task in the System Admin task group can be used by the administrator to copy the template-level formats to all the subdomains.

> **Note:** In order for this process to work, when the workbook is being built, the positions from multiple local domains must be included so that a master workbook is created first. Once the workbook is created, the format changes are saved to the template level (in future releases) and then the Apply Format OAT task is executed to copy it to every local domain. If a local domain only workbook is built and the formats saved, this task will not work.

Schedule Dashboard Workbooks

The Dashboard is a feature in RPASCE that is loaded when a user first logs into the system. The dashboard gives you access to key statistics. You can use the dashboard to analyze the health of the business. The dashboard must be refreshed periodically, as new products and locations are added. This typically happens weekly, but depends on your administrator's settings. The measure data of existing products and locations in the dashboard can be refreshed at any time to view updated changes.

The dashboard is supported by a workbook template that must be configured based on the customer business. It also requires the configuration of a json file using the Config Tools and uploaded during the building of the domain. See *Oracle Retail Predictive Application Server Cloud Edition Configuration Tools User Guide* for more information.

The dashboard workspace can be built using Schedule Dashboard Workbooks, which can be accessed from the System Administration Tasks Online Admin task group.

Figure 3–19   Schedule Dashboard Workbooks

The following options are available:

- Task Label. Enter the name of the build task in order to review the status of a build.

- RPAS_TODAY. Use if needed to build the dashboard as of a date other than the current one. Optional.

- Range dimension. The Calendar dimension against which the dashboard is pre-ranged.

- Pre-Season Range. The number of periods of the Range dimension that must be included before RPAS_TODAY.

- Post-Season Range. The number of periods of the Range dimension that must be included after RPAS_TODAY.

- Delete and Recreate Dashboard Segment. Use only if the dashboard template styles are patched. The use of this option results in the lose of all dashboard formats.

- Users. Select who can access the dashboard: All Users or User Groups.

4

Internationalization

Internationalization is the process of creating software that can be translated into different languages more easily. Changes to the code are not specific to any particular market. This section describes configuration settings and features of the software that ensure that the base application can handle multiple languages. Oracle Retail applications have been internationalized to support multiple languages.

This chapter contains the following sections:

- Translation
- Translation Administration
- Taskflow Components Translation
- Position Label Translation
- Translation of Online Administration Tools
- Troubleshooting

Translation

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

- Graphical user interface (GUI)
- Error messages
- Taskflow components

The following components are not translated:

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

The user interface has been translated into the following languages:

Table 4–1   Supported Languages with Language Identifiers

Language	Language Identifier	Windows Language ID
Chinese (Simplified)	CHINESE_SIMPLIFIED	2052
Chinese (Traditional)	CHINESE_TRADITIONAL	1028
Croatian	CROATIAN	26
Dutch	DUTCH	19
English	ENGLISH	9
French	FRENCH	12
German	GERMAN	7
Greek	GREEK	8
Hungarian	HUNGARIAN	14
Italian	ITALIAN	16
Japanese	JAPANESE	17
Korean	KOREAN	18
Polish	POLISH	21
Portuguese (Brazilian)	PORTUGUESE	22
Russian	RUSSIAN	25
Spanish	SPANISH	10
Swedish	SWEDISH	29
Turkish	TURKISH	31

Translation Administration

> **Note:**   For information on the translation of position labels, see
> Position Label Translation.

Every product, location, and calendar position can be presented in multiple languages, as can messages presented through the client. However, before translated strings can be viewed in the client, the following processes must be followed to set up the environment to support multiple languages.

1. Build the domains with the Multi-Language setting enabled in the Configuration Tool properties.

2. Change the Web browser locale settings to reflect the relevant language settings in the application user interface using the following steps:

    a. Launch Internet Explorer.

    b. In the Tools menu, click **Internet Options**. The Internet Options window appears.

    c. In the Internet Options window, on the General tab, click **Languages**. The Language Preference window appears.

    d. In the Language Preference window, click **Add** in the Language Preference area to add the relevant language.

  e. Select the language you added, and use the **Move Up** or **Move Down** buttons to set up the order of preference.

  f. On the Language Preference window, click **OK**.

  g. On the Internet Options window, click **OK**.

3. Log on to the RPASCE Client as an administrative user, and build the Translation Administration workbook using the steps below.

  a. Click **Tasks**.

Figure 4–1 Tasks
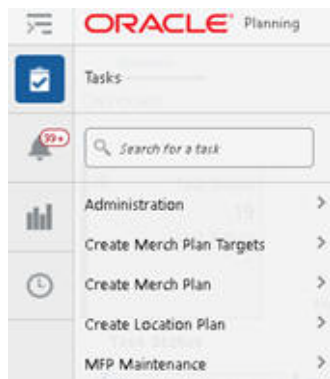
  b. Select **Administration**.

Figure 4–2 Administration
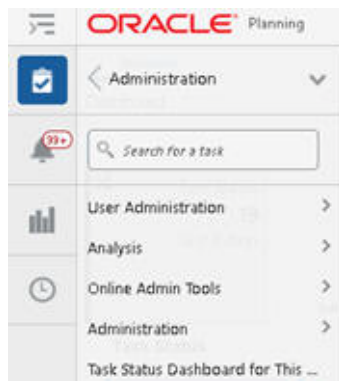
  c. Select **Administration** again.

  d. Select **Translation Administration**.

Figure 4–3   Translation Administration
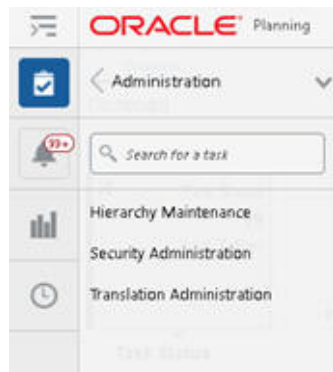
4. See the Translation Administration Workbook for the available translation options. Note that unless the workbook is committed, any new translation strings entered in the workbook will not be reflected in the domain.

Translation Administration Workbook

The Translation Administration workbook contains views for translating text used in measure labels, workbook template names, template group names, user group labels, and general areas (for instance, wizard instructions, and error messages).

Note: RPASCE and solution-specific messages to the user *must* not be modified. If changes are made to these messages, they may be overwritten when patching occurs.

Note: In addition to the following views, there are other views in this workbook for translation of dimension position labels. See Position Label Translation for more information.

Hierarchy Labels View

The Hierarchy Labels view allows the user to view and edit the translations of hierarchy labels. Translations are supported for each of the system's allowable alternative languages.

Dimension Labels View

The Dimension Labels view allows the user to view and edit the translations of dimension labels. Translations are supported for each of the system's allowable alternative languages.

Workbook Template Group Labels View

The Template Group Translations view allows the user to view and edit the translations of template group names. Translations are supported for each of the system's allowable alternative languages. Translations in this view affect the labels on the tabs that appear in the File - New dialog (for example (in English), Administration, Analysis, and Predict).

Workbook Template Labels View

The Template Translation view allows the user to view and edit the translations of workbook template names. Translations are supported for each of the system's allowable alternative languages.

Measure Labels, Measure Descriptions, and Measure Picklists View

The Measure Translations view allows the user to view and edit the translations of measure labels, measure descriptions and measure picklists. Translations are supported for each of the system's allowable alternative languages.

Measure picklist translations are saved in the domain in the translation measure r_measpicklist. As with other RPASCE translations in the Translation admin workbook, it is possible to load the underlying translation measure instead of entering the translations in the Translation administration workbook and committing it.

The following example illustrates loading translations for the picklist measure tdaccesstx using loadmeasure. If you are going to enter the translations in the translation administration workbook instead, just ignore the first two columns in the measure data file and pay attention to the values in the value field that is the third and final column of the measure load file below. Create a r_measpicklist.csv.ovr file with the translations for the example picklist measure tdaccesstx. It is a csv file, so the fields are comma separated. Each row is in the format below.

<picklistMeasureName>, <languageId> , "value1(value1Label), value2(value2Label), valueN(valueNLabel)"

The example contents for r_measpicklist.csv.ovr with translations for a picklist measure tdaccesstx are shown in Figure 4–4.

Figure 4–4   Example Contents

```
tdaccesstx,english,"0(Denied),1(Read Only),2(Full Access)"
tdaccesstx,french,"0(Refusé),1(Lecture seule),2(Accès complet)"
tdaccesstx,chinese_simplified,0(已拒绝)、1(只读)、2(读/写)
tdaccesstx,chinese_traditional,"0(拒絕),1(唯讀),2(讀取/寫入)"
tdaccesstx,croatian,"0(Odbijeno),1(Samo za čitanje),2(Čitanje/pisanje)"
tdaccesstx,dutch,"0(Geweigerd),1(Alleen-lezen),2(Lezen/Schrijven)"
```

r_measpicklist will contain default translations for many internal RPASCE picklist measures as well, so it is important to use the ovr format when adding new picklist measure translations and not some other data file format that might wipe out all other picklist measure translations.

User Group Labels View

The User Group Translations view allows the user to view and edit the translations of user group labels. Translations are supported for each of the system's allowable alternative languages. The list of user groups includes the Administration, Default, and Internal user groups, plus any other user group names set up by the system administrator. For products in the Oracle Retail Predictive Planning Suite, the list of user groups also includes the various planning roles.

Rule Group Labels View

The RGRP Labels view allows the user to view and edit the translations of rule group labels displayed to users in the RPASCE Client. Translations are supported for each of the system's allowable alternative languages.

Taskflow Components Translation

UI side translations are done using a properties file used in the UI named MultiSolutionBundle.properties. The default file is in English and is located in the config/MultiSolution/resources directory. It contains the strings for labels, descriptions, instructions for Activity Groups, Activities, Tasks, and Steps. This file also contains domain information for a multi-solution installation. Additional files can be created that contain strings for translated languages. These files are in the format of MultiSolutionBundle_XX.properties, where XX is the language code. RPASCE supports base translations for the languages listed in Table 4–1.

> **Note:** Decimal digit and separator formatting cannot be configured in the RPASCE Client. It relies on the user's browser locale for this, where the choice of the decimal and digit grouping separator characters is derived from the language specified in the user's browser session.

Custom Labels for Hierarchies and Dimensions

The dimensions or hierarchies in RPASCE can have different labels in different templates. For example, the user may find it more meaningful to label the PROD hierarchy as "Stock" instead of "Product" in a particular template.The procedure to make this work is detailed as follows. The same process can be used to translate the labels for workbook-only dimensions.

1. Create a message ID and load the ID to the msgs hierarchy using the loadHier utility.

   - For dimension labels use [dimension name]_[template name] as the message ID.

   - For hierarchy labels use [hierarchy name]_[template name] as the message ID.

2. After the message IDs are loaded into the msgs hierarchy, the user can load the r_msglabel measure or use the Translation admin workbook to create the translation for these message IDs.

Position Label Translation

To enable translation of position labels for the desired dimensions, they must be marked translatable in the configuration. Building or patching the domain with this configuration builds the necessary infrastructure in the domain to manage translations for those dimensions. However, label translations must be separately loaded.

Position label translations are loaded in dimension-specific translation measures for every language used by the users. If translated labels are not loaded using these measures, workbooks show position names wherever a label has to be shown. Note that for a translatable dimension, RPASCE never uses or shows the position labels from the hierarchy load file but always refers to the labels in dimension-specific translation measures. This implies that if a domain were patched to make a dimension translatable but the translation measures were not loaded, RPASCE users would see position names instead of position labels from the load file.

Dimension-specific position translation measures are named as r_<dim name>label, where <dim name> must be replaced with the name of the translated dimension. For example, if the sku dimension were to be translated, load the r_skulabel measure with

translations. These measures must be loaded after loading the hierarchy because RPASCE can only load translations for already loaded positions.

The position label translation measure load files have three columns. The first column has the position names, the second column has the language identifier, and the third column has the translation for the language specified in that row.

For example, a translation measure file for the sku dimension is named r_skulabel.csv.ovr and has the content formatted, as shown in the following example. Note that the same file contains labels in four languages.

```
10006782,ENGLISH,White Nike Running Shoe size 11
10006782,CHINESE_SIMPLIF,????????11
10006782,FRENCH,Taille blanche 11 de chaussure de course de Nike
10006782,ITALIAN,Formato bianco 11 del pattino corrente di Nike
10004523,ENGLISH,Black leather shoe size 8
10004523,CHINESE_SIMPLIF,?????8
10004523,FRENCH,Taille noire 8 de chaussure en cuir
10004523,ITALIAN,Formato nero 8 del pattino di cuoio
```

> **Note:** For a list of language identifiers, see Table 4–1.

Alternatively, you can manually enter or alter translated labels using the Translation administration. In this workbook, a worksheet is available for each dimension that has translations enabled. You can manually enter translated strings for the language of interest. After they are committed, these translations are available for every new workbook.

It is possible that, because of error that may have occurred when translation files were prepared, translated labels for some positions may not be loaded. In a situation where RPASCE is unable to look up the label for the locale of the machine on which the RPASCE client is being run, RPASCE looks for a non-empty label string for the English language. If it fails to find a non-empty label string for the English language, it uses or shows the loaded position name of the position.

Translation of Online Administration Tools

The following are translatable in OAT.

- Administration task list label
- Administration task label
- Administration task description
- Argument branches label
- Argument label

Troubleshooting

The following measures are used internally by RPASCE for storing the translated labels coming in from the translation administration workbook. During the domain build, the data files for these measures in different languages can be placed in the input folder of the domain and loaded.

If you see any untranslated text in the UI and want to verify that the translated strings are loaded correctly in the domain, follow Table 4–2 to find the translation measure that holds the text and print its contents.

Table 4–2   Translation Measures

Text Type	Measure
Pick list measure labels	r_measpicklist
Dimension labels	r_dimlabel
User group labels	r_grplabel
Hierarchy labels	r_hierlabel
Measure description text	r_measdescripti
Measure labels	r_measlabel
Workbook template group labels	r_wbtglabel
Workbook template labels	r_wbtlabel
Internal messages and generic strings	r_msglabel
Position label	r_<dimensionname>label

5

Online Administration Tools

The RPASCE Online Administration interface allows the application administrator to manage back-end operations on a domain from the RPASCE Client. The Online Administration interface consists of two parts. The first part is a set of activities in the taskflow that authorized users access in order to launch and manage back-end processes from the Client. The second part is a dashboard that an administrator uses to monitor the status of all the administration tasks after they have been submitted.

This chapter contains the following sections:

- Online Administration Tools Templates
- System Administration Tasks
- Service Request Administration Tasks
- Administration Dashboard

Online Administration Tools Templates

Several wizard-only workbook templates are available that the RPASCE administrator can use to manage RPASCE administration tasks on the RPASCE server. These templates are located in the workbook template group called Online Administration Tools, which is available from the task flow menu under Administration.

Use the following menu to access the administration tools.

Figure 5–1   Online Administration Tools
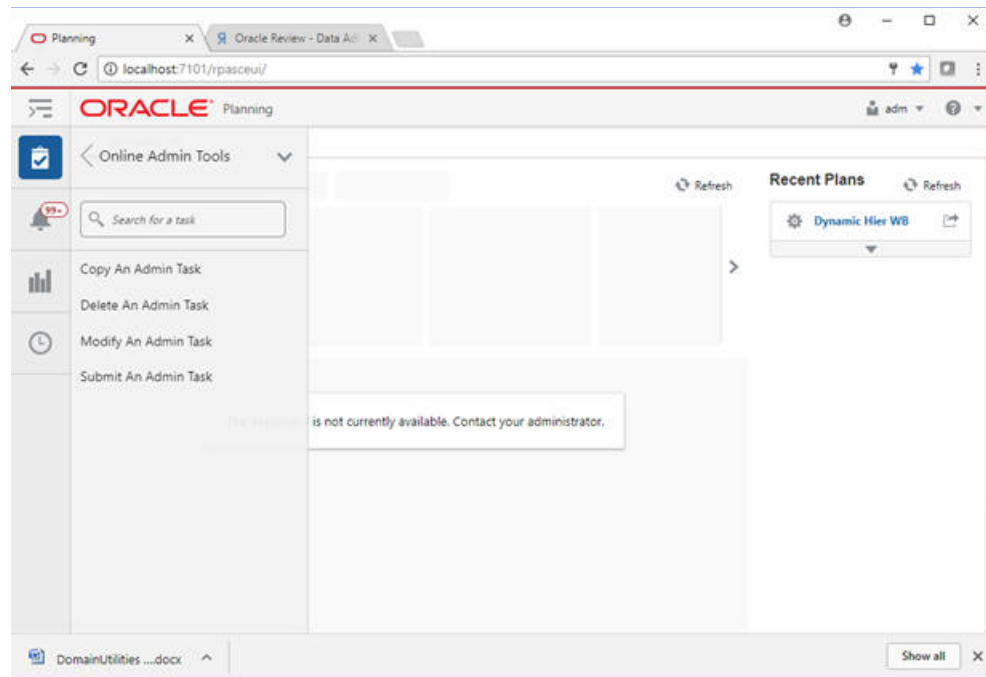
Copy an Administration Task

This template allows the administrator to copy a scheduled or completed administration task. The first wizard page provides a drop-down list for the user to select a task. Subsequent wizard pages are similar to those in the Submit an Administration Task template, but with the argument selections populated with the values of the existing task.

A new label for the task must be entered.

Delete an Administration Task

This template allows the user to delete an active task that has been scheduled but not yet started. The list only included scheduled tasks.

Modify an Administration Task

This template allows the user to modify a scheduled administration task. The first wizard page provides a drop-down list for the user to select a task. Subsequent wizard pages are similar to those in the Submit an Administration Task template, but with the argument selections populated with the values of the existing task.

A new label for the task must be entered.

Submit an Administration Task

The Submit an Administration Task template is a multiple-page template in which the contents of each page are dependent upon the selection of the previous page.

The first wizard page prompts the user to choose one of the task lists available in the domain.

The second wizard page is used for selecting one of the pre-defined tasks.

A job label is required that identifies the submitted task. A meaning label is recommended since it is the identifier of the task when its status is displayed.

The uniqueness of the job label is not checked. It is the user's responsibility to ensure that the name is recognizable.

From the third wizard page, the user selects or enters the command line arguments for the selected task.

If the argument list contains argument branches, another wizard page will be generated based on the user's selection of the branches. This is done recursively until all argument branches are exhausted.

After all arguments are specified, the next and final wizard page, which is used for scheduling options, is displayed.

The user sees the Confirm Action page. Verify the options on this page and click **Finish** to submit this task for execution.

System Administration Tasks

Table 5–1 lists the system administration tasks available for the RPASCE platform.

Table 5–1 Standard Administration Tasks for Domain

Administration Task Name	Administration Task Label	Description	Notes
dimensionMgr	Dimension Manager	Dimension manager	
domaininfo	Domain Information Utility	Domain information utility	
domainprop	Domain Property Utility	Domain property utility	
exportHier	Export Hierarchy Data	Export hierarchy data	
exportMeasure	Export Measure Data	Export measure data	
hierarchyMgr	Hierarchy Manager	Hierarchy manager	Only supports the following options: list, specs
listActiveUsers	List Logged-in Users	List the users who are currently logged into the domain.	
loadHier	Load Hierarchy Data	Load hierarchy data	
loadMeasure	Load Measure Data	Load measure data	
mace	MACE Utility	MACE - multi-dimensional array calculation engine	Only supports informational commands: find, check, print.
optimizeDomain	Optimize Domain	Optimize domain to improve performance and free up space	
printMeasure	Print Measure	Print Measure Information	
reindexDomain	Reindex Domain	Reindex domain	
renamePositions	Rename Positions	Rename positions	

Table 5–1   (Cont.) Standard Administration Tasks for Domain

Administration Task Name	Administration Task Label	Description	Notes
scanDomain	Scan Domain Utility	Scan domain utility	
scheduleDash	Schedule Dashboard Workbooks	Schedule dashboard build for any users	
packageOutput	TarGz - Create a Package	Package data files in the output directory after exporting data from the domain	
unpackageInput	TarGz - Extract a Package	Unpack data archive files in the input directory for use in data loading	
Usermgr	User Manager	User and group manager	
wbbatch	Workbook Batch Utility	Workbook batch utility	
wbmgr	Workbook Manager	Workbook manager	

Service Request Administration Tasks

Table 5–2 lists the Service Request (SR) administration tasks available for the RPASCE platform.

Table 5–2   Service Request Administration Tasks by Domain

Administration Task Name	Administration Task Label	Description	Notes
copyDomainTask	Clone Domain	Clone and package the domain for submission with service request.	
copyLogsTask	Package Logs	Package the log files for submission with service request.	
envprofile	RPAS Environment Profile	Gather environment profile about the RPASCE system and domain.	
startOSWatcher	Start Oracle OS Watcher	Start Oracle OS Watcher to gather OS metrics for performance analysis	
stopOSWatcher	Stop Oracle OS Watcher	Stop the running of the Oracle OS Watcher	

Application Administration Tasks

Optionally, one or more sets of required application-specific administration tasks can be set up. These tasks are outside the scope of this document and are covered in the application-specific administration guides.

RPAS_TODAY for Online Administration Tools Tasks

RPAS_TODAY for some Online Administration Tools (OAT) tasks is provided to handle the following user scenario. For example, a customer wants to run one of the time-intensive tasks, the workbook batch task, on Friday night. The coming Monday morning, the customer checks the task status and finds it failed (due to some error in the batch script). After fixing the issue in the batch script, the customer might find there is no way to rerun this batch script with the past timestamp set as RPAS_TODAY because the customer does not have access to the back end to change the system environment value of RPAS_TODAY.

RPAS_TODAY is an environment variable used in RPASCE and also available as a task-specific variable for some tasks in the Online Administration Tools. In an OAT environment, this task-specific RPAS_TODAY will overwrite the RPAS_TODAY environment variable (if it exists) when the task is being executed. After the task finishes, the task-specific RPAS_TODAY no longer exists and the existing RPAS_TODAY environment variable will take effect.

Currently, four online administration tools support the task-specific RPAS_TODAY setting. They are Load Hierarchy Data, Load Measure Data, Schedule Dashboard Workbooks and WbBatch (Figure 5–3).

Figure 5–2   WbBatch Task Example

For these four online administration tasks selected from the Admin Tasks Group, there is one text field in the Task Option wizard page. Here the task, WbBatch, is used as an example (Figure 5–2). Note that RPAS_TODAY for OAT task is an optional input.

Figure 5–3   Online Administration Tools Supporting RPAS_TODAY

After user enters the RPAS_TODAY for a particular OAT task, this RPAS_TODAY value will be set for this task and will appear in the rest of Task Option wizard pages (Figure 5–4), and in Task Confirmation wizard page (Figure 5–5).

Figure 5–4   Task Option Wizard Page

Figure 5–5   Task Confirmation Wizard Page

System Administration Tasks

This section lists all system administration tasks and the operations and arguments specific to them.

Dimension Manager Task

The Dimension Manager task is used for displaying dimension properties.

Name of Dimension to Which the Settings Should Apply

This is the drop-down list of dimensions. Any subsequent operations apply to the dimension(s) selected.

Display Properties of the Dimension

This displays the properties of the selected dimension. These properties are used to indicate that the DPM and translation are enabled for the dimension and that the dimension is image-enabled.

Here is the list of properties that this argument returns:

- Dimension label
- Associated hierarchy of the dimension
- Start and width of the file load
- Label start and width of the file load
- Number of used positions
- Image enabled
- Bit size

- Capacity

- Used

- Reindex threshold

- Translation enabled

- Spreads or aggs (that is, where the dimension fits in the rollup)

- If the dimension is at the base level of an indicated hierarchy

Domain Information Task

The Domain Information task is used to provide miscellaneous details about a domain, including type of domain (simple, master, subdomain, or local) and the upgrade and version history of the domain.

Display Terse Output

This option is used to reduce the amount of output that is displayed.

Display All Basic Information of the Domain

This is a combined display of all the properties that can be selected in the display that shows the various properties of the domain option.

Display Various Properties of the Domain

When the user selects this and clicks **Next**, the window to enter additional arguments for the operation is displayed.

The options are:

- Display version of referenced domain. This displays the RPASCE version of the specified domain.

- Display domain version expected by code. This displays the expected RPASCE version of the domain the utility expects.

- Display integration status of domain.

- Display domain type.

  - Simple: domain is a standard, non-partitioned domain

  - Global: domain is the master of a global domain set

  - Subdomain: domain is one partition of a global domain set

- Display application associated with domain. This displays the application associated with the master domain.

- Display version history of referenced domain. This displays the version history of the domain, specifically, when the domain was upgraded to a new version of RPASCE (patch or release).

- Display all subdomains of a global domain. This displays a list of all the local domains in a global domain environment and indicates which positions at the partition level are in each local domain.

- While displaying subdomains, indicate if paths are relative. This indicates if the paths are relative when the subdomains are listed.

- Display master domain path and partition dimensions for subdomain. It displays the path to the master domain and the partition dimensions for subdomains.

Display File Size Information for the Domain

This displays the file size for the domain.

Display String Stats for the Domain

This displays the number of strings for all given lengths that occur in the domain. The output returns a list of string lengths (for example, 0.758, 121, 28, 369).

Display String Values for the Domain

This displays the number of occurrences of each unique string that are in the domain. The output returns a list of each unique string (for example, 785", 1 copied pasted, 2%, 1% 1 <= passwd <= 2, 2% 1 Window).

Display Array Btree Density

This displays the array btree density.

Display Array Schemas Count

This displays how many arrays in the domain are formatted for each schema. The output returns a list similar the following:

- Count of array schema for the domain <name of domain>
- Array schema
- Unknown schema

Display All Arrays Path to dimRegistery

This displays all the array paths to dimRegistry.

Domain Property Utility Task

The Domain Property Utility task is used to manipulate the properties of a domain, including password properties, locks on user accounts, and information about daemons.

Reset the Domain Properties to a Default Value

This resets most of the domain properties to the default value. It also creates any required domain directories.

Display the Current Value of All Domain Properties

This displays the current value of all domain properties.

Display the Current Value of the Named Property

This displays the current value of the specified property.

Update the Named Property to the Value Provided

This specifies the property to be changed. When the user selects this, enters a value, and clicks **Next**, the window to enter additional arguments for the operation is displayed. When the value of a property is being updated, one of the optional switches (property as numeric or property as Boolean) can be used to indicate the type of property. If a value is not provided, the previous type of the property will be used. If this is a new property, then the type will be string.

The options are:

- Property is numeric.

The property type is numeric.

- Property is Boolean.

The property type is Boolean.

Domain Statistics Generation/Validation Task

This Online Administration Task allows customers to validate the domain data after the migration.

Domain migration from RPAS BFC version 16.0.2 to RPASCE implies domain rebuild and data migration via flat files. A systematic process is needed to facilitate data validation to ensure error-free migration to RPASCE. The goal is to minimize manual data inspection across the original and migrated domains. The scope of the validation does not include workbook data. Workbook data is expected to be committed to the domain prior to the migration. The original and migrated domain must be in an identical state with regard to data loading and calculations.

Assumptions

The following assumptions are made as applied to the original and migrated domains:

- The RPASCE domain must be created from the upgraded configuration associated with the original domain.

- The original configuration must first be upgraded to RPAS BFC 16.0.2 base version and then opened and saved with the RPASCE Configuration tools.

- Data must be exported from the original domain and loaded to the new domain prior to comparison.

- Dimensional data (for example, skus, styles, and so on.) must be the same across the original and new domains to avoid false alarms from the validation tool.

- Original and migrated domains must be in a consistent state with respect to the batch execution. This eliminates false alarms due to the comparison of the calculated measures.

False alarms may be detected if the new domain is built from an altered configuration that has a different hierarchy structure or measure definitions. It is important to keep the hierarchy structure intact and changes to the measures to the minimum.

Validation Process

The validation process includes two stages:

Stage 1: Generate Report Files for Dimension and Measure Data in the Original Domain

The OAT options are:

- Generate domain statistics

  - Generate all domain statistics

  - Generate selected measure statistics

The above two options result in the creation of a measure statistics report file in the outgoing FTP path. The report file includes measure information such as intersection, type, pop-count, average-pop, min-pop, max-pop, status, and measure na-value for all measures in the domain.

The status file flags error situations where the measure data cannot be read or there is no database associated with the measure.

Pop-count is calculated based on the measure NA Value.

For non-numeric types, average, min, and max are reported as 0.

Generate selected measure statistics option can limit the scope of the report to a specific set of measures specified on the command line in a comma-delimited list.

Note that both of the above options result in the creation of the Dimension Statistics file and are placed in the outgoing FTP path. The dimension statistics report file includes a row corresponding to each level of the hierarchy. The row looks like the following:

Hierarchy Name, Dimension Name, Active Position Count

These options result in a separate CSV files for dimension and measure data (dimDataStats.csv and measDataStats.csv). Both report files include headers that describe the fields. The report files are located in the outgoing FTP location.

Stage 2: Validation

The OAT options are:

- Validate domain statistics

    - Validate all measure statistics

    - Generate selected measure statistics.

    - Continue measure validation even if difference in dimension statistics

Dimensional and measure statistics are read from the CSV files from Stage 1. The corresponding dimensions and measures in the new domain are validated against the content of the reports.

A Generate selected measure stats option limits the scope of validation to a selected list of measures.

Output files (dimCompReport.txt and measCompReport.txt) are generated that include the discrepancies between the CSV report and the new domain.

The output files are placed in outgoing FTP directory.

Export Hierarchy Data Task

The Export Hierarchy Data task is used to export all the positions in a hierarchy, including their rollup relations. The utility exports all hierarchy positions, but the file may be specified to include only formal or informal positions. The resulting file can then be used as a .dat file with Load Hierarchy.

Hierarchy Name

This is the drop-down list of available hierarchies for export.

The Path Where .dat File is to be Created

This .dat file can be used in load hierarchy to load the hierarchy into a domain. Only the file name must be specified. The file is created in the output directory of the domain.

Output the .dat File in Fixed-Width Format

This is used to output the .dat file in a fixed-width format instead of comma-separated values (CSV). The width of each column is set to the size set during the domain build. Consult WIDTH and LABEL WIDTH in the Defining Dimension Properties section in the *Oracle Retail Predictive Application Server Cloud Edition Configuration Tools User Guide*.

Only Include Formal Positions in the .dat File

If this is specified, the informal position will be skipped.

Only Include Informal Positions in the .dat File

If this is specified, the formal position will be skipped.

Export All Position Names in Uppercase

If this is specified, the position names will be exported in all uppercase.

Generate Header Line for Dimension Columns and Label Columns

This is used to generate the header line for a CSV export. The line contains fields for the dimension columns and label columns.

Export Positions of User-Defined Dimensions

This can only be used with the Only Include Formal or the Only Include Informal options. The user-defined dimensions can only be exported in CSV format.

File to be Created Containing Informal Positions

This creates a file containing informal positions in a format that can be used to set positions back to an informal status using the Informal Position Manager utility.

Export Measure Data Task

The Export Measure Data task is used to export domain or workbook measure data from RPASCE to either a CVS or a fixed-width file format. A single measure or multiple measures can be exported, based on a specified intersection.

 If the measure's base intersection is not the same as the export intersection, the measure's default aggregation method will be used to aggregate data to an intersection higher than base, or replication will be used to spread the measure data if the data is required at an intersection lower than base. This utility supports the export of data in a user-specified range, which can be a single mask measure, in a range specified in the calendar dimension or in a combination of the two.

Hierarchy Manager Task

The Hierarchy Manager task is used to list all the hierarchies in a domain and their properties. Hierarchy structures are loaded into a domain using the Load Hierarchy utility.

List All Hierarchies in the Domain

All hierarchies in the domain are listed.

Display Properties of the Hierarchy

When the user selects this and clicks **Next**, the window to enter additional arguments for the operation is displayed.

Hierarchy name: Displays the properties of the specified hierarchy.

Load Hierarchy Data Task

The loadHier utility is used to load and refresh a hierarchy. It supports either a comma-separated value (CSV) or fixed-width flat files for loading. The load file must have a .dat file extension when a flat file is loaded and a .csv.dat file extension when a CSV file is loaded. When a fixed-width file is used, the width of fields (number of characters) is specified in a configuration file before a domain is built. The width of fields can be increased after a domain has been built using the Dimension Manager utility or by changing a property in the Configuration Tools and patching the domain. The utility also allows a simple compression method that can skip duplicated values line by line.

Exclusive Message

This is a notification message for a domain-exclusive administrative task. After the task has been started and is in progress in the domain, if the user tries to access domain data, such as committing or refreshing a workbook, this message will be displayed and the operation will be terminated.

Load and Refresh the Specified Hierarchy

When the user selects a hierarchy and clicks **Next**, the window to enter additional arguments is displayed.

The options are:

- Do not clean input files and metadata used after load. This prevents the removal of input files and temporary data files that are generated during the hierarchy load process. Input files remain in the input directory of the domain after the process is completed. This option is often used for debugging or troubleshooting purposes.

- Log successfully loaded input file lines: Logs successfully loaded input file lines into a loaded [HIERNAME].dat file under the processed directory.

- Enforce new hierarchy rollup changes: Enforces new hierarchy roll-up changes so that they dominate existing hierarchy roll-ups in case they conflict with the roll-ups specified in the input file.

- Force NA consistency: Forces NA consistency when the current NA value is different from the originally defined NA value for the measure.

- Load user-defined positions back to the domain: If the data file contains positions for UDD dimensions, this check box indicates that this data must also be loaded. The CSV format with a header line is expected. Data for all user-maintained dimensions must be presented in the data file.

- Purge age: The number of days before the hierarchy data is purged from the domain. For example, the purge age for SKU position 12345 is set to 30 (days). If the next load of the prod.dat (containing SKUs) does not have position 12345 in it, it will compare the current date with the date when that position was last loaded. If that position was last loaded in the past 30 days, it is kept. However, if it has been more than 30 days since the load hierarchy data has loaded that position, then 12345 is purged from the domain.

- Default domain paths to accommodate new positions: Specifies comma-separated default domain paths that are used for accommodating new partitions. Default domains can point to existing or new (non-existing) local domains.

- Column order base on the header line: Specifies the column order base on the header line. Otherwise, the order is based on the start property of the dimension. This option only has meaning when loading a CSV file and has no effect on loading fixed-width files.

Load All Hierarchies

When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The options are:

- Do not clean input files and metadata used after load: Prevents the removal of input files and temporary data files that are generated during the hierarchy load process. Input files remain in the input directory of the domain after the process is completed. This option is often used for debugging or troubleshooting purposes.

- Log successfully loaded input file lines: Logs successfully loaded input file lines into a loaded [HIERNAME].dat file under the processed directory.

- Enforce new hierarchy rollup changes: Enforces new hierarchy roll-up changes so that they dominate existing hierarchy roll-ups in case they conflict with the roll-ups specified in the input file.

- Force NA consistency: Forces NA consistency when the current NA value is different from the originally defined NA value for the measure.

- Load user defined positions back to the domain: If data file contains positions for UDD dimensions, this check box tells load hierarchy data to load this data also. The CSV format with header line is expected. Data for all user-maintained dimensions must be presented in the data file. Any missing user-defined dimensions cause an error. All loaded positions have formal status after running.

- Purge age: The number of days before hierarchy data is purged from the domain. For example, SKU position 12345 has its purge age is set to 30 days. If the next load of the prod.dat (containing SKUs) does not have position 12345 in it, it will compare the current date with the date when that position was last loaded. If that position was last loaded in the past 30 days, it is kept. However, if it has been more than 30 days since the load hierarchy data was loaded that position, then 12345 is purged from the domain.

- Default domain paths to accommodate new partitions: Specifies comma-separated default domain paths that are used for accommodating new partitions. Default domains can point to existing or new (non-existing) local domains.

- Column order base on the header line: Specifies the column order base on the header line. Otherwise, the order is based on the start property of the dimension. This option only has meaning when loading a CSV file and has no effect on loading a fixed-width file.

Load Measure Data Task

The Load Measure Data utility is used to load measure data from text files into the domain. The administrator must specify the measure names and the path to the domain that contains the measures.

The loadmeasure utility supports the use of fixed width and CSV files for loading measure data. RPASCE recommends the use of CSV files to reduce the size of the load file and to reduce disk I/O time.

To load measure data, system administrators must copy or create one or more load files in the input folder of the domain directory. The administrator can then call loadmeasure to load data.

Name of the Measure to be Loaded

If more than one measure is specified, all the measures must be in the same input file. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed. Measure Name is the name of the measure to be loaded.

The options are:

- Do not move input files to processed directory: Input files are not moved to the processed directory. This prevents the input files from being moved to the processed directory. This option is used when a single file is used to load multiple measures but not all measures from the file are loaded at once. This option instructs loadmeasure to leave the load file behind for subsequent loading of unloaded measures. The user can use this option to perform intermediate processing between loads of measures available from the same file.

- Apply staged loads: This applies to any staged loads for the named measure. If the measure is registered to be a stage-only measure, loadmeasure will put the load in a staging area but will not update the measure until loadmeasure is called again with this argument. When this argument is used, loadmeasure applies all loads that have been queued up in the staging area. It clears out the staged loads unless the measure's loadsToKeep property has been set to a non-zero number. In such cases, it does not clear out the latest loadsToKeep loads.

  Only .ovr, .inc, and .clr loads can be staged. The .rpl loads cannot be staged. Additionally, staging is only allowed for base intersection loads. RPASCE cannot stage loads where the load intersection is below the base intersection of the measure.

  This argument must not be used for measures that are not stage-only.

- Purge old measure data even if no new data loaded: This forces the purge routine to run even if no new data is loaded. This purges old measure data. This option can be applied to stage-only measures without having to apply loads. When a measure has the Calendar hierarchy in its base intersection, the setting for the purgeAge measure property defines how and when existing data gets purged to a NA value. If purgeAge has not been set, the data never gets purged. If a purge age of zero or more has been set, data is purged for all dates that are before RPAS_TODAY - purgeAge days. For example, if purgeAge is five, at data load time all data that is more than five days before RPAS_TODAY will be purged.

  This option does not require you to load any new data.

- Defragment the domain after measure data is loaded: Used to shrink the size of a domain after measure data is loaded.

- Split input files in global domain across local domains: This causes the input files in the global domain to be split across the local domains, but does not do any further processing of the input files.

- Load the pre-split input files into the local domains: Used to load the splitting information into the local domains.

- Logging level for record loading issues: Used to specify the log level for any per record logs. Record logs include messages indicating that record line was shorter than expected, a data column was missing, one or more positions were missing,

record could not be parsed, or data conversion errors. Standard log levels (for example, error or warning) are used for this option.

Load All Measure Files Located in the Specified Directory

Only .rpl files can be used with this option, and only the CSV format with a header line is supported. The header line is used to map the columns to dimensions and measures (for example, SKU,STR,DAY,Sales). Enter one measure per input file. The name of the measure is extracted from the file name. For example, sales.csv.rpl corresponds to measure sales. The input data must be at the base intersection of the measure. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

- Do not move input files to processed directory: This prevents input files from being moved into the processed directory.

Mace Informational Commands Task

Mace is used to evaluate rule groups or expressions in order to manipulate measures.

Search for a Specific String

Searches all expressions for the specified printing of all the rules and rule groups that have these expressions.

Validate a Specific Expression

Validates the specified expression string.

Print the Specified Rule

When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Enter the rule list: Prints all the specified rules. The rule list is a comma-separated list of rule names.

Print the Specified Rule Group

When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Enter the group list: Prints all the specified rule groups. The rule group list is a comma-separated list of rule group names.

Print All the Rules and Rule Groups

All rules and rule groups are printed.

Optimize Domain Task

The Optimize Domain utility is used to improve performance and minimize the space required by the domain data.

Because the RPASCE Btree dimension arrays undergo continuous updates and changes as a result of adding and deleting existing positions, over time, measure arrays become full of stale data. This stale data is created when positions are deleted,

but the associated index remains. This results in wasted space and inefficient operations. The measure arrays containing any data for these positions must be updated to reflect these deletions. Deleting positions marks the hierarchy data for the corresponding dimensions as changed, but it does not clean up the associated data from the measure arrays. The Optimize Domain utility cleans this stale data from the measure arrays.

For example, consider a measure called meas1 that exists at intersection sku_str_week. When a user adds a DPM position skudpm1 through an RPASCE client, it is assigned to an integer index in the meas1 data array in addition to being assigned within the dimension's array. When this DPM position is deleted, the position is flagged as inactive, but the integer index cannot be reused until the dimension is reindexed. The measure array also still contains the data associated with the inactive position. This measure array must also be updated in order to remove the stale data.

The same also holds true for formal positions that are purged during a normal hierarchy patch. The measures arrays must be updated for this set of positions because they may be pointing to stale data as well.

In addition, over time, existing arrays in RPASCE databases may become fragmented, resulting in wasted space and a possible degradation in efficiency on array operations. This occurs because the RPASCE Btree array stores data in chunks of disk memory called pages. Deleting data from the array causes empty spaces or holes to develop in these pages. These holes contain no data, yet they still increase the overall size of the array. Over time, these holes become larger and more frequent, and array operations suffer degraded performance as a result. Also, the pages begin to acquire large amounts of unused space, causing inefficient disk usage and a larger than optimal domain size.

The Optimize Domain utility is used to correct these two issues.

Selectively Defrag Domain Databases

Selectively defragments the domain data based on database fragmentation. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

- Return a non-zero value if any of the databases or arrays failed to defrag: The process still runs to completion. Default behavior is to log a warning and continue.

- Force to defrag entire measure arrays set: The entire measure arrays set is defragmented. This option makes the defragmenting process run longer.

Clean Domain Stale Data

When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

- Return a non-zero value if any of the databases or arrays fail to clean:

Print Measure Task

Print Measure is used to print measure information.

List All the Measures

This returns a list of all registered measures in the domain. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Workbook Name: Specifies the workbook associated with the measures to print.

List the Intersection at Which Data is Available

This returns the base intersection of the measure. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Measure Name: Specifies the measure to print.

Print Description of the Measure

This returns the list of measure properties. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Measure Name: This specifies the measure to print.

List the Information of a Measure at its Base Intersection

This lists the number of dimensions, logical size, populated size, and density of a measure at its base intersection. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Measure Name: This specifies the measure to print.

Purge Online Admin Tasks

Purge Online Admin Tasks is used to save or remove online admin tasks that are in the success or failed queues of the online admin dashboard.

Use this task to help improve the online admin dashboard load and refresh time in the UI by purging old tasks from the dashboard.

Operations

- Delete the tasks?

  Use this option to delete the purged tasks selected from the queues.

- Archive and send to outgoing ftp?

  Use this option to save a copy of the purged tasks into the outgoing ftp path logs directory in order to back it up.

  > **Note:** It is the responsibility of the customer to remove these files after they have been fetched from the outgoing ftp path logs directory.

Purge Age

- Remove All

  Use this option to purge all the tasks from selected queues up to the current date.

- Retain last three days

  Use this option to retain tasks from the last three days and purge the remainder of the tasks from the success and failed queues.

- Retain last five days?

Use this option to retain tasks from the last five days and purge the remainder of the tasks from the success and failed queues.

- Retain last seven days?

    Use this option to retain the tasks from the last seven days and purge the remainder of the tasks from the success and failed queues.

Queues

- Tasks success status

    Use this option if you only want to select the tasks in the success queue.

- Tasks failed status

    Use this option if you only want to select the tasks in the failed queue.

- Tasks in success/failed status

    Use this option if you only want to select the tasks in the failed and success queues.

Reindex Domain Task

The Reindex Domain task is used to compress, increase, or decrease the set of physical address spaces (or indexes) of the multidimensional arrays. The process of compressing and defragmenting the physical IDs makes the domain load and run faster. The reindexing options let you reindex the entire domain, dimensions within a hierarchy, or a specified list of dimensions, and also prepend calendar dimension positions.

Reindex All Dimensions of the Specified Hierarchy

This reindexes all dimensions of the specified hierarchy base on reindex conditions checks. When the user selects a hierarchy and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

- Force reindexing: This skips the reindexing conditions check for all dimensions provided.

Reindex the Specified Dimensions

This reindexes the specified dimension. If this is selected, the user must provide a value from the drop-down list of dimensions. When the user selects a hierarchy and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

- Force reindexing: This skips the reindexing conditions check for all dimensions provided.

Reindex All the Dimensions of the Domain

This reindexes all dimensions in the domain. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

- Force reindexing: This forces reindexing at the domain/hierarchy/dimension level. It does not check the reindexing condition.

Create a New DimRegistry by Allocating Space

When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

- Force reindexing: This forces reindexing at the domain/hierarchy/dimension level. It does not check the reindexing condition.

Analyze All the Dimensions of the Domain

This analyses the dimensions of the entire domain and prints the report.

Analyze All the Dimensions of the Specified Hierarchy

This analyses the dimensions of the hierarchy and prints the report. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

The option is:

Hierarchy Name: The name of the hierarchy to be analyzed.

Analyze the Specified Dimensions

This analyses the dimensions of the hierarchy. It checks whether reindexing is necessary for hierarchies or dimensions. This argument returns details about the dimension name, bitsize, number of available and used position IDs, the threshold ratio, and whether reindexing is necessary. If this is selected and the user clicks **Next**, the window to enter additional arguments opens in the workspace.

Print the Current Reindexing Properties

Prints the current reindexing properties.

Rename Positions Task

The Rename Positions task is used to change the name of one or more positions. Positions that are to be renamed must be included in a hierarchy data file. After the hierarchy data files have been updated and placed in the designated location, an administrator must run the Rename Positions utility.

Figure 5–6   Submit New Admin Task

Hierarchy Name

This indicates the hierarchy for which positions are being renamed.

Do Not Apply Changes

This reports which changes are applied, but it does not apply the changes. A dry run still generates a log file.

Input File Format

The Rename Positions OAT task renames positions in a specified hierarchy. It looks for a file named hierName.rn.dat (for example, prod.rn.dat) in the domain's input directory. This file is usually copied to that location through a batch script in the Batch Framework.

The input file is a CSV text file with three columns. Tab-delimited fields are not supported in the input file. This is consistent with other domain utilities in which tab-delimited fields are not supported. One line is required for each rename request.

Each line consists of a dimension name, the old position name, and a new position name. For example:

sku,10000_old,10000_new

The utility ignores any line not incorrectly formatted, empty lines, and lines with a specified dimension that does not exist for the specified hierarchy.

The input file is not case sensitive, and all names are converted to lower case before the positions are renamed. Old position names that are specified but that do not actually exist are ignored. A new position name cannot be an already used name. Lines with invalid position names are ignored and added to the log file. The old position name and new position name must not be prefixed with the name of the dimension.

The width property of the dimension is enforced. The width attribute in the domain must be greater than or equal to the maximum length of the new names in the input file. If the width of the new name is greater than the width attribute of the corresponding dimension, RPAS prints an error in the log file and ignores the record.

Dimensions specified in the input file must belong to the hierarchy that is specified in arguments. Otherwise, the record is ignored and RPAS prints an error in the log file.

If you want to swap position names (a->b, b->a), you must perform two separate operations in this order: (a->tempa, b->tempb) and (tempa->b, tempb->a).

Scan Domain Utility Task

The Scan Domain utility is a domain utility used for detecting data loss and repairing data corruption in an RPASCE database. Data loss occurs when an RPASCE process is abnormally terminated. This can happen when an external mechanism, such as a power failure, causes a sudden termination of an RPASCE process. Data loss can also occur due to an unexpected program breakdown.

Data corruption can occur if an external program modifies the RPASCE database files or an unforeseen defect occurs in the processes using the RPASCE database (an extremely rare event).

The Scan Domain utility can detect both corruption and data loss, but it can only fix corruption. This utility can operate on global, non-partitioned, and local domains. It supports parallelization when repairing databases in a domain.

While the utility is attempting to perform a repair of the databases, it can use the option to enable backing up the original databases. While running in detection mode (check for data loss or check for data corruption), the utility does not change any of the RPASCE databases, and therefore, it does not create such backups. In detection mode, the output can be directed to a file.

Repair Database Corruption

This repairs the database corruption in the specified domain. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

- Back up database files before repairing them: This backs up database files before repairing them. It cannot be used with Check for data loss in the domain or with Check database corruption in the domain.

- Skip folder containing user information while scanning: This skips the folder containing user information while scanning.

Check for Data Loss in the Domain

This checks for data loss in the specified domain. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

- Skip folder containing user information while scanning: This skips the folder containing user information while scanning.

Check Database Corruption in the Domain

This checks for database corruption in the specified domain. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

- Skip folder containing user information while scanning: This skips the folder containing user information while scanning.

Check Both Database Corruption and Data Loss in the Domain

This checks for both data loss and database corruption in the specified domain. When the user selects this and clicks **Next**, the window to enter additional arguments is displayed.

■ Skip folder containing user information while scanning: This skips the folder containing user information while scanning.

Segment Manager Utility

The Segment Manager utility, WbBatch, is used by the administrator to manage the segment and build, commit, refresh, or remove a segment and build, commit, refresh, or remove a segment as well as the batch operation on the workbooks in the build, commit, and refresh queues.

■ Remove a segment

■ Remove the workbook for a segment

■ Build a segment

■ Commit a segment

■ Refresh a segment

■ Add a segment to the batch build queue

■ Add a segment to the batch commit queue

■ Add a segment to the batch refresh queue

■ Remove a segment from the batch build queue

■ Remove a segment from the batch commit queue

■ Remove a segment from the batch refresh queue

■ Start a batch operation on all/build/commit/refresh queues

■ Print a segment in all/build/commit/refresh queues

TarGz Create a Package Task

The TarGz Create a Package task is used to create a tar.gz compressed archive file at the specified location. The task is implemented as a script that uses tar and gzip system utilities to create the archive.

Archive File Name

This is the name of the tar.gz archive. The archive name is used as it is and the script does not add any extensions automatically.

Overwrite Existing Archive

If checked, it overwrites the existing tar.gz archive. If an archive of the same name exists and this option is not checked, then the compression is not performed.

Delete Archived Source Files

If checked, it deletes all the existing source files in the specified directory after the successful completion of the compression. If this option is not checked, then the source files are retained.

Archive Files

A multi-select list box that displays the list of all files that can be archived and compressed.

TarGz Extract a Package Task

The TarGz Extract A Package task is used to extract a tar.gz compressed archive located at a specified location. This task is implemented as a script and uses the gunzip and tar system utilities to extract the archive.

Archive File

This is the name of the tar.gz archive that must be uncompressed.

Delete Extracted Archive

If checked, it deletes the tar.gz archive after successful extraction. If not checked, the tar.gz archive is retained.

Workbook Manager Task

The Workbook Manager task is use to inspect and remove existing workbooks.

List All Workbooks in the Domain

This lists workbooks in the domain.

List Workbooks Belonging to a Specific User

This lists workbooks in the domain belonging to a specific user.

List Workbooks Belonging to a Specific Template

This lists workbooks in the domain belonging to a specific template. This option can also be used to filter based on users.

Print Detailed Information About Workbooks

This prints detailed information about workbooks.

Remove All Workbooks in the Domain

This removes all workbooks in the domain.

Remove Workbooks Belonging to a Specific User

This removes workbooks from the domain belonging to a specific user.

Remove Workbooks Belonging to a Specific Template

This removes workbooks from the domain belonging to a specific template. This option can also be used to filter based on users.

> **Note:** Removing the workbook from the domain will always report success in OAT. If any workbook is open, it skips that workbook deletion and continues to delete the other workbooks. This functionality prevents the task from failing on first failure to delete the workbook. Make sure to always check the log files after the task is completed.

Service Request Administration Tasks

This section lists the tasks for the task group called "SR Admin Tasks." This task group covers all the diagnostic tasks and the operations and arguments specific to them.

These tasks are based on the existing RPASCE utilities. The output from these tasks are used for diagnostic purposes.

These operations are supported as part of diagnostic tasks:

- Clone Domain
- Package Logs

Both the Clone Domain and Package Logs tasks create packages at a predetermined location on the back-end server that is identified by the domain property, r_oat_sr_dest. This path must already exist and sufficient permissions granted to RPASCE Daemon for create/update/delete operations. This path must be outside of the RPASCE application domain and preferably on a separate mount. This path is used exclusively for these two SR operations.

Package Logs Task

The Package Logs task is used to copy the User Sessions and OAT task logs and package them into a zip file. The output package is named as logpackage-[category].gz, where category is either all, usersessions, or oattasks, based on the user selection. Note that the existing package file will be overwritten. The OAT task logs also contain the output from the batch scripts.

Set the following arguments:

- **Task Label**

  This is the label for the task.

- **RPAS_TODAY**

  This should be ignored.

- **Options**

  Depending on the option selected, the User Session logs, OAT tasks log or all logs will be copied and packaged.

Copy User Session Logs

Figure 5–7   Set Arguments for Copy User Session Logs

Set the following arguments:

- **Include logs from this date**

  The user session logs created after this date are packaged.

- **Select users to include their logs**

  Logs of the selected users are packaged.

Copy OAT Task Logs

Set the following argument:

- **Select tasks to include their logs**

  Logs of the selected tasks are packaged.

Copy All Logs

Set the following arguments:

- **Include logs from this date**

  The user session logs created after this date are packaged.

- **Select users to include their logs**

  Logs of the selected users are packaged.

- **Select tasks to include their logs**

  Logs of the selected tasks are packaged.

Clone Domain Task

The Clone Domain task provides two main options:

- **Clone only metadata**

Metadata cloning creates a much smaller package as it does not include measure data, workbooks, or other non-essential data. It is best used for diagnosing hierarchy issues. However, this package is not a fully operational clone.

■ **Clone by an intersection**

Clone by an intersection produces a clone based on the intersection provided. The user must input the intersection option exactly as provided by the GRID team. No spaces must be added. This clone contains a subset of the data from the original domain, and it is a fully operational clone. Users have additional options to select the workbooks to be included with the clone or to not include any workbooks at all.

Clone Domain Operation

Set the following arguments:

■ **Exclusive Message**

This is a notification message for a domain-exclusive administrative task. After the task has been started and is in progress in the domain, if the user tries to access domain data, such as committing or refreshing a workbook, this message will be displayed and the operation will be terminated.

■ **Task Label**

Label for the Clone Domain task.

■ **RPAS_TODAY**

This should be ignored.

■ **Options**

– Only metadata - Used for diagnosing hierarchy issues. However, this package is not a fully operational clone.

– Intersection clone - Contains a subset of the data from the original domain. It is a fully operational clone. The option opens another wizard page to include or exclude workbooks.

Set the following arguments:

■ **Exclude all workbooks**

Workbooks are excluded from clone operation.

■ **Include selected workbooks**

The user can select one or more workbooks to be packaged as part of clone operation.

Start Oracle OS Watcher

The Start Oracle OS Watcher task is used to collect OS performance metrics with user-specified parameters. When the task ends, the collected data is packaged in a tar file named as osw_archive_<timestamp>.tar. The file is moved to the location defined by the domain property r_oat_sr_dest. If r_oat_sr_dest is not defined, the tar file is located in $RPAS_HOME/oswbb directory. The collected data, along with other profile data or logs, is used to help Oracle Support to diagnosis performance issues of the RPASCE applications.

Set the following arguments:

Task Label

A short description of the task. This label is displayed on the Task Status Dashboard to identify the task.

Snapshot interval in seconds

Interval at which the hardware snapshots are taken. The recommended interval of 30 seconds is pre-selected.

Duration of data retention (in hours)

This parameter specifies the length of the retention period of the collected data. The recommended value is 24 hours. That is, the OS Watcher task will keep the data from the last 24 hours; data older than 24 hours is deleted.

Duration of OS Watcher run in hours

This parameter specifies the length of run time for the task. It should be long enough to cover the duration of the RPASCE task whose performance is of concern, with the addition of a period of time before and after the concerned task execution.

Note that only one running instance of the OS Watcher task is allowed at any given time. If a task of OS Watcher is already running, any subsequent attempts to start the task will fail. As with any other Online Admin Tasks, the OS Watcher task can be scheduled or run as soon as possible.

Stop Oracle OS Watcher Task

The purpose of this task is to stop the running OS Watcher before its duration ends. When the Stop Oracle OS Watcher task starts, it will immediately stop the running OS Watcher task, and package the collected data in a tar file, the same way the Start Oracle OS Watcher task does at the end of its duration.

Set the following argument:

Task Label

A short description of the task.

Bootstrap Administration Tasks

This section lists the tasks for the Bootstrap task group. This task group covers all the tasks that are required to build a domain from the configuration. It is also possible to rebuild an existing domain using the bootstrap tasks. Once the domain is built, there is a task available to restart the WebLogic server with the taskflow and resource xml files from the new domain.

Applications override the base RPASCE version of the bootstrap task group with application specific bootstrap tasks. For complete details regarding the setup required for using the base bootstrap tasks that is not application specific, refer to the RPASCE Enterprise Edition implementation guide. For application specific bootstrap tasks refer to the application's documentation. This section briefly describes the RPASCE Enterprise Edition bootstrap tasks and their parameters for quick reference.

Start by selecting the Submit a New Admin Task template that is attached to the bootstrap domain. This opens up the screen shown in Figure 5–8. These are the bootstrap tasks that are available.

Figure 5–8   Bootstrap Group Task
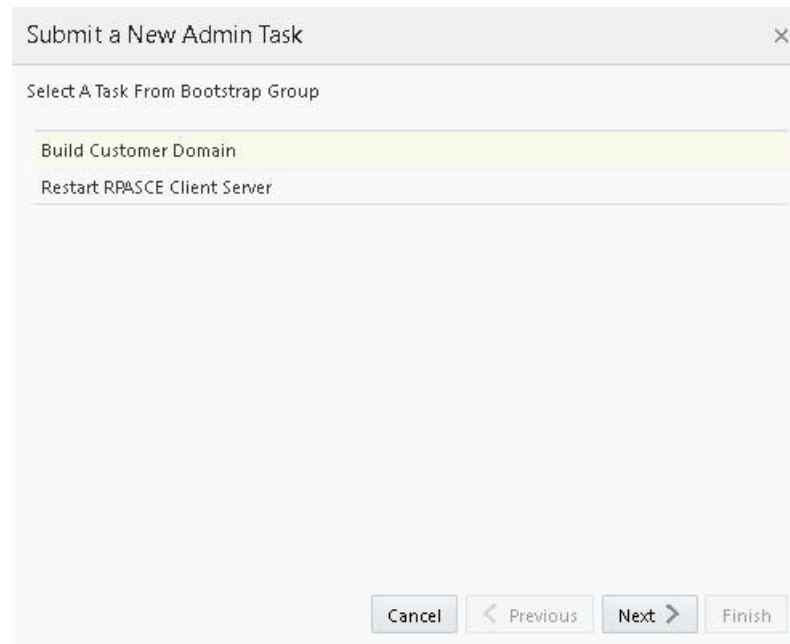
Build Customer Domain Task

This task is used to build a new domain. Prior to using this task make sure the following pre-requisites are met:

- The configuration has been uploaded to the config folder under the incoming FTP site.

- Input files have been uploaded to the input folder under the incoming FTP site.

- Batch control files have been uploaded to the batch_control folder under the incoming FTP site.

- Java special expression jar files if required must be located in the jse folder under the incoming FTP site.

Figure 5–9   Build Customer Domain Task Arguments

Set the following arguments:

Task Label

This is the label for the task; it can be any string. It is used to identify the task after it is submitted.

Config Name

The name of the domain that must be built. This field is a single select drop-down list whose entries are populated by listing the configurations that are present in the config folder under the incoming FTP site. The configurations must be archived with the extension .zip or .tar.gz. In addition, the archive file must be suffixed with _config (for example, configName_config.zip or configName _config.tar.gz).

Partition Dimension

The dimension on which the domain will be partitioned. The domain is constructed with one subdomain for each position in the given dimension. This must be a level of separation that fits with the intended workflow for individual users so that, when possible, most users' daily tasks relate to only one subdomain. (This reduces contention when many users are active in the system.)

Overwrite existing domain

Overwrite in the case where the domain has already been built once. The implementer must rebuild the domain from scratch (perhaps because a non-patchable change has been made to the configuration). This option must be selected. If it is left in the default unselected state, then the domain build process will halt and report an error, rather than overwrite the existing domain.

Input Data File Archive (Optional)

It is possible to upload the input files archived with extension .zip or .tar.gz. This drop-down list is populated with all the tar.gz and .zip files present in the input folder of the incoming FTP site.

JSE jar files (Optional, Comma separated)

If any Java Special Expression .jar files are used by the domain configuration, they must be indicated here with the .jar file name only. If you are including multiple .jar files, provide a comma-separated list of .jar file names. This field is optional and must be left blank if JSE extensions are not used. If listed, the .jar files named here must be present in the incoming FTP area under the jse directory. If the named .jar file(s) are not present, the domain build process will halt with an error.

Batch task group to run after domain build

Once a domain has been built successfully, a named group of batch operations may be specified (typically including measure data loads and mace calculations). This operation sequence must be one batch_type entry in the Batch Exec control file, batch_exec_list.txt (described in Batch Exec service section).

Restart RPASCE Client Server Task

This task must be run after the first time a domain is built in order to enable logging in to the new domain. At the time of RPASCE installation, the RPASCE client's information is taken as input. This task will run the WebLogic restart script in the RPASCE client server, both of which are specified during the RPASCE installation. The relevant screen from the RPASCE installer is shown in Figure 5–10, with some example inputs for reference.

Figure 5–10   RPASCE Client Details
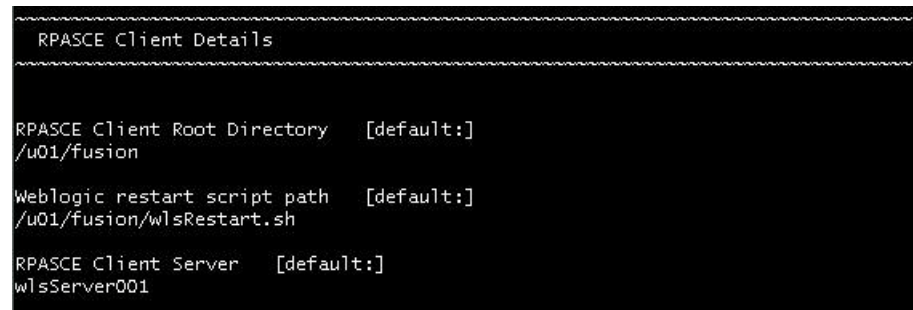
This task has only one argument.

Task Label

This is the label for the task; it can be any string. It is used to identify the task after it is submitted.

Administration Dashboard

The Administration Dashboard provides an interface that can be used to monitor the status of the tasks and download the log file when a task is completed. It allows the user to filter the submitted tasks by type, owner, or time. The status tiles show the percentage of tasks in that status and also act as a filter if the user clicks on them. The job name will become a hyperlink to the log file once the task is completed.

Figure 5–11   Accessing Task Status Dashboard
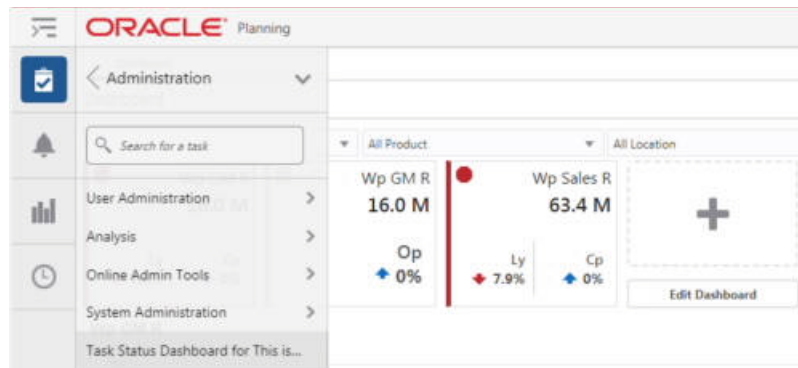
It can be accessed from the menu by clicking **Administration -> Task Status Dashboard**.

Figure 5–12   Status Dashboard
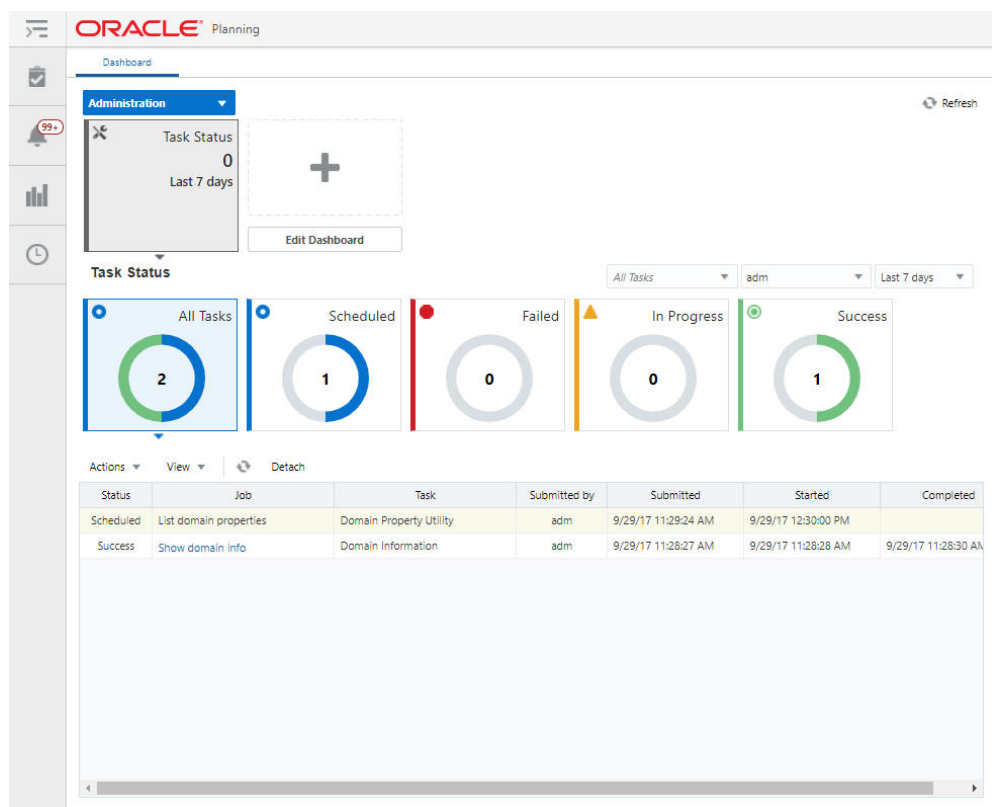
The Status Dashboard displays the details of all or some administration tasks, based on the filters that are applied. Logs are available for each task.

Dashboard Columns

The following data is displayed in the dashboard.

Job Label ("Job")

This represents the task label provided by the user. You can use this field to differentiate between the various jobs that have been submitted or scheduled. For

example, Domain Information is a task in the Online Administration Tool. You can schedule this task to run daily and assign the task label Domain Information Daily to it. At the same time, you can schedule the same task to run weekly and assign the task label Domain Information Weekly to it.

Admin Task Name ("Task")

This represents the name of the task. The name is unique for the task type. All labels are pre-defined and cannot be changed by the user.

Submitter ("Submitted by")

This represents the login name of the user who submitted the task.

Start Time ("Started")

This represents the time when the submitted started. The submission time and the start time can differ for scheduled tasks.

Completion Time ("Completed")

This represents the time when the task was completed or ended. If the status of a task is either In Progress or Scheduled, then this field is left blank.

Submission Time ("Submitted")

This represents the time when the task was submitted.

Status ("Status")

This represents the status of the task. The possible status values are Success, Failure, In Progress, Scheduled, and Pending. If the task fails, a message link is also provided that shows the failure message when you hover over the status field.

Sorting

The task list can be sorted by any column, with the exception of the Output column, by using the Sort Ascending and Sort Descending arrows that are visible when you hover over the column header.

Filtering

The task list can be filtered by type, owner, or time. The status tiles are also filters that when clicked will cause the task list to only show tasks of that status.

Logs and Messages

You can access the log for the particular task by clicking on the link provided in the Job label field of the Status Dashboard. You are prompted to open or download the log file. If the task fails, you will see a failure message when you hover over the failed status.

6

Informational Utilities

Many RPASCE utilities can be used to find information about the different components of a domain or domain data. The utilities are submitted to the back end for execution through the RPASCE Administration Tool on the User Interface (UI). The following utilities are used solely for retrieving information and are not used to make any changes to a domain or to data in a domain.

- domaininfo is used to retrieve domain information
- printMeasure is used to print data from measures

Retrieving Domain Information

The domaininfo administration task is used to provide miscellaneous details about a domain, such as the upgrade and version history of the domain, the associated application, and so on.

Submitting a domaininfo Task

To submit a domaininfo task, complete the following steps:

1. From Task Flow, select **Admin Tasks for Application Domain** under **Tasks**.

Figure 6–1   Admin Tasks for Application Domain
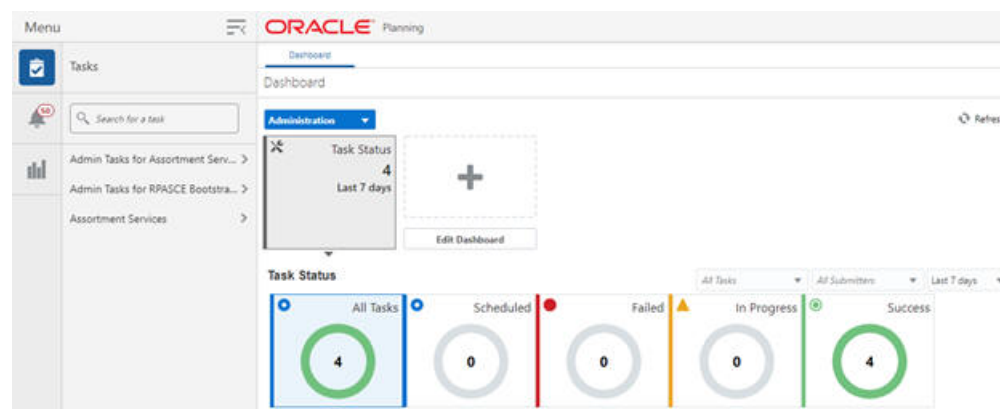
2. Select **Online Admin Tools**.

Figure 6–2   Online Admin Tools
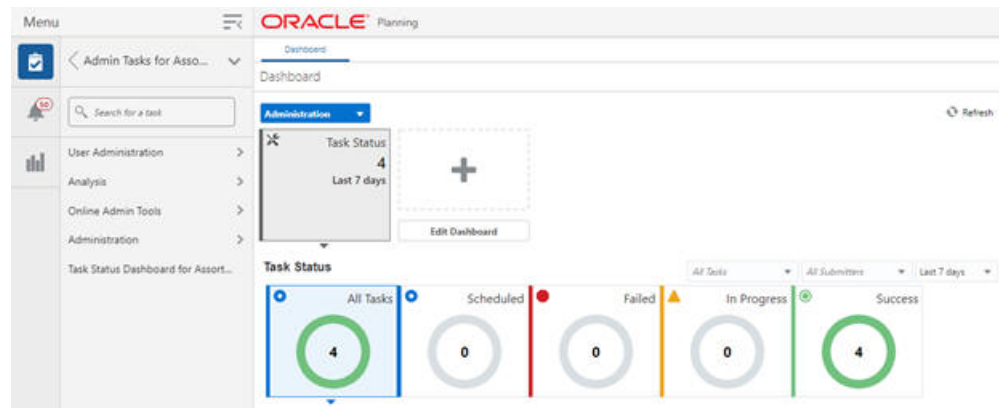
3.   To submit a new Administration task, select **Submit A New Admin Task**.

Figure 6–3   Submit a New Admin Task
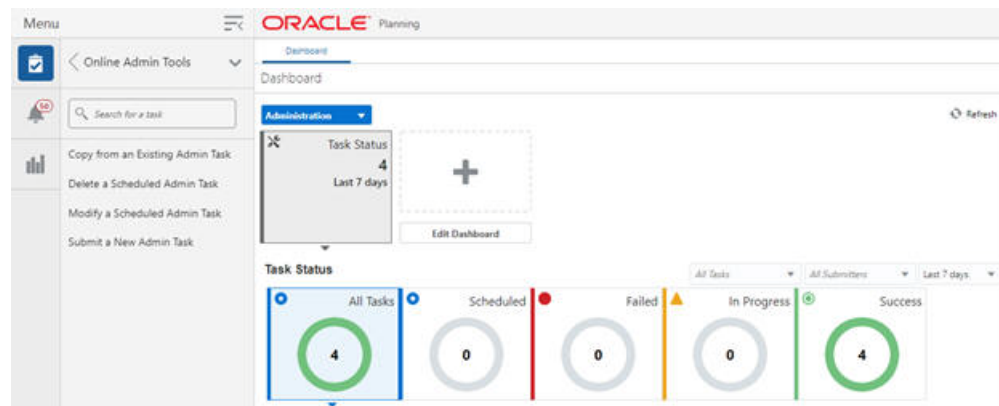
4.   Select **System Admin Tasks** and click **Next**.

Figure 6–4   System Admin Tasks
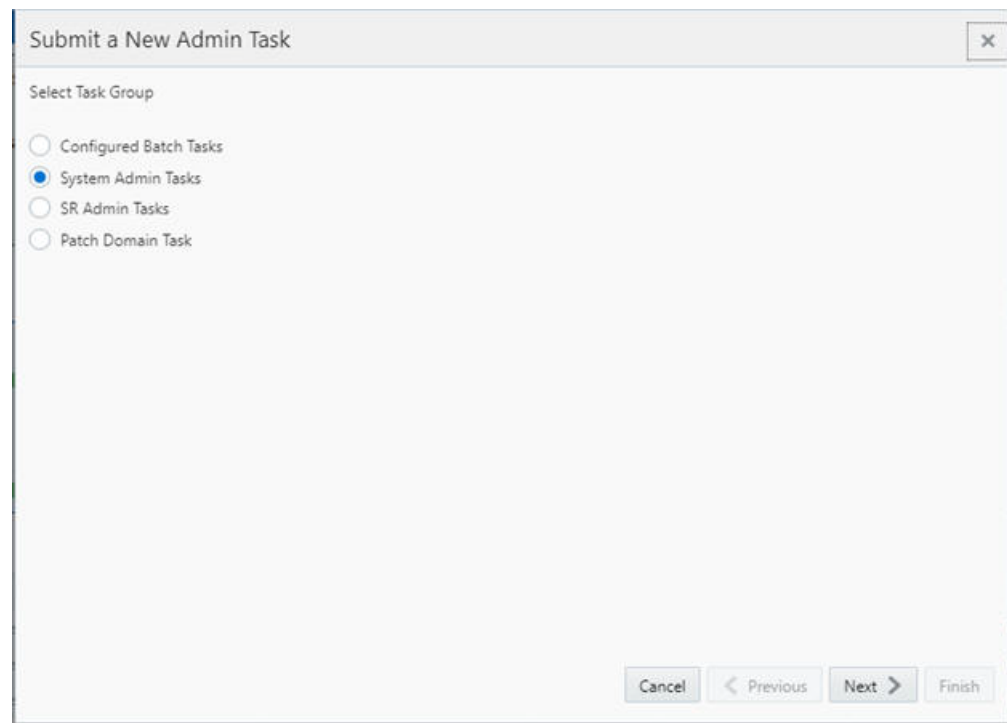
5. From the list of all available system administration tasks, select **Domain Information** and click **Next**.

Figure 6–5   Domain Information
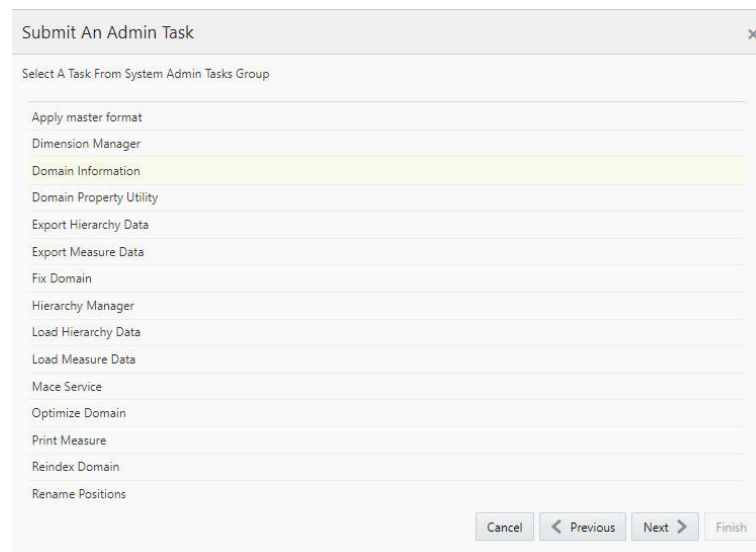
6. Enter a descriptive label in the Task Label text box.

    Select one of the following operations:

    ■   Display all information: Displays the domain version, expected domain version, domain type, associated application, history.

■ Display domain properties: You will need to select the specific properties on the next page.

Figure 6–6   Select Operation

Click **Next**.

7. If you selected Display domain properties, select one or more of the following domain properties to retrieve:

   ■ Display domain version - displays the RPASCE version of the domain.

   ■ Display associated application - displays application associated with the domain.

   ■ Display version history - Displays the version history of the domain, specifically, when the domain was upgraded to a new version of RPASCE (patch or release).

   ■ Display file size - Displays the file size information for the domain.

   ■ Display array btree density - Displays the array btree density.

   ■ Display array schemas count - Displays how many arrays in the domain are formatted for each schema. The output returns a list like the following:

PageSize	2,048	2,792 arrays of Schema 11 with KeyType ULong32 and RPAS_UT_ G/data/admin.db/commit_asap_config.ary
PageSize	4,096	49 arrays of Schema 11 with KeyType ULong32 and RPAS_UT_ G/data/admin.db/sessioninfo.ary

   ■ Display paths to dimRegistry - Display all arrays path to dimRegistry.

Figure 6–7   Display Domain Property
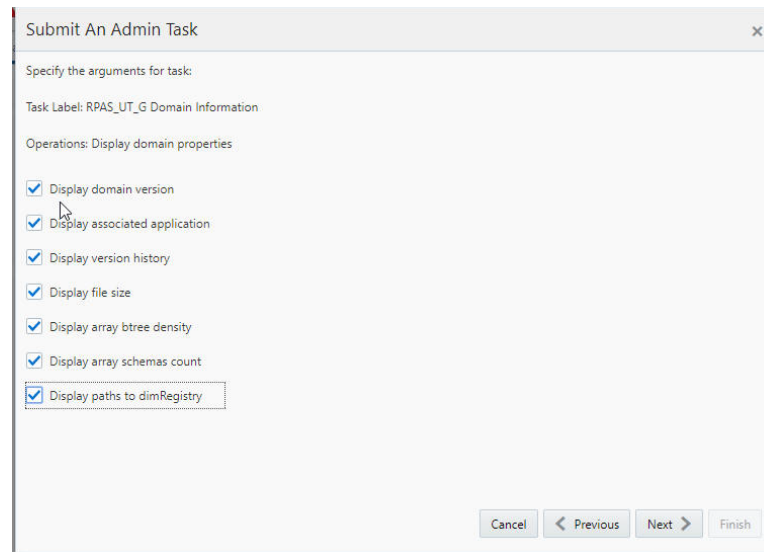
8. Click **Next**. You see Schedule Task, where you can choose to run the task ASAP or on a schedule.

Figure 6–8   Schedule Task
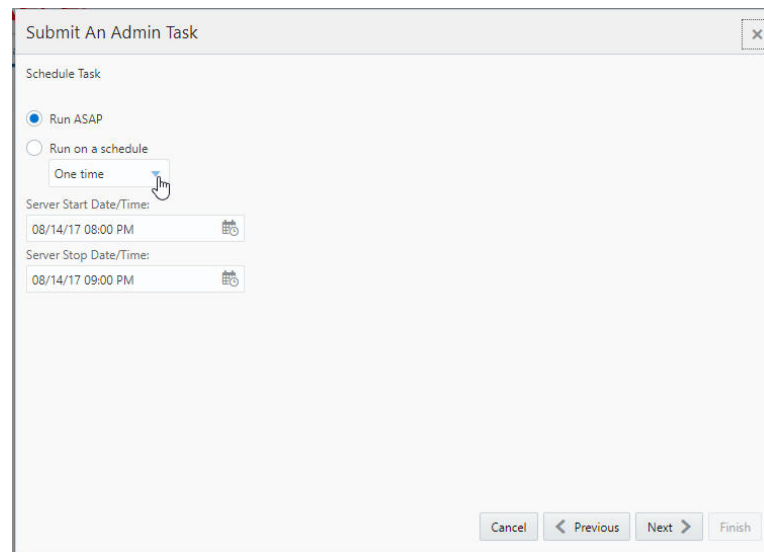
9. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 6–9 Verify and Confirm
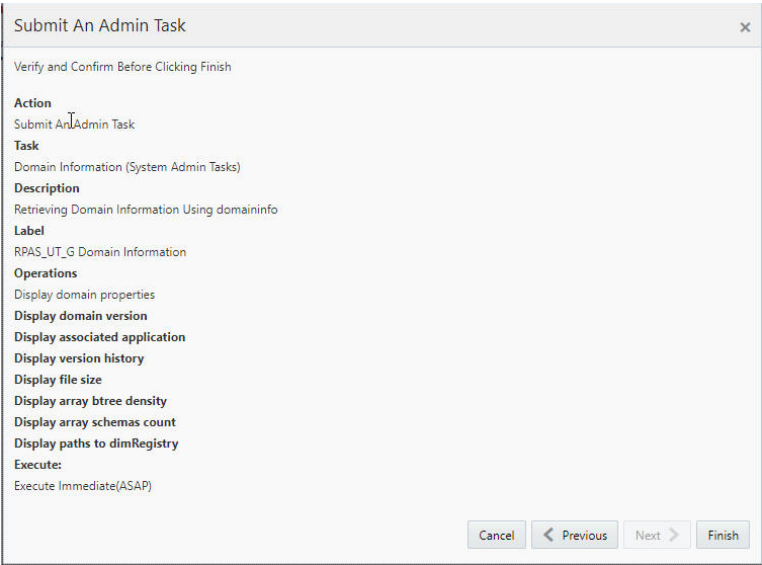
Printing Measure Information

The administration task Print Measure is used to print measure information.

Submitting a printMeasure Task

To submit a printMeasure task, follow the steps below on UI:

1. From Task Flow, select **Admin Tasks for Application Domain** under **Tasks**.

Figure 6–10 Administration
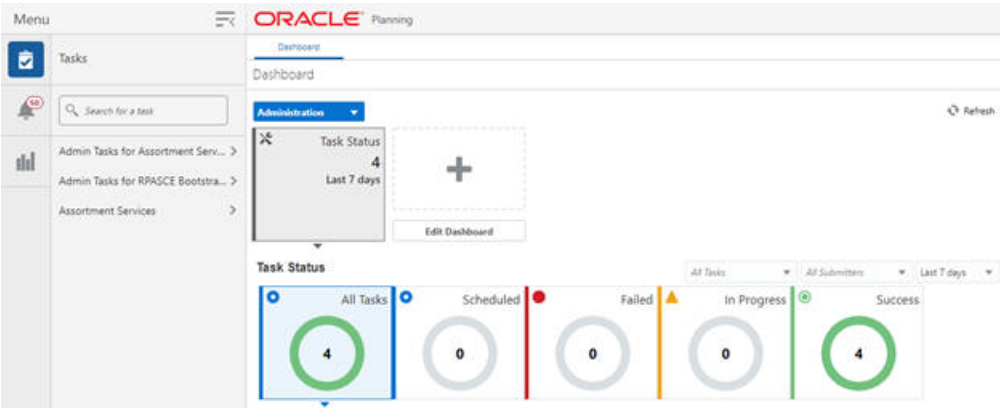
2. Select **Online Admin Tools**.

Figure 6–11   Online Admin Tools
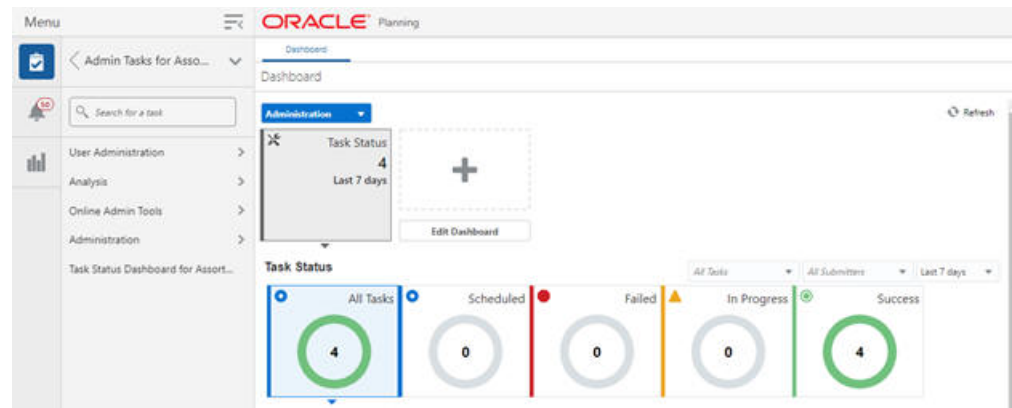
3. To submit a new Administration task, select **Submit A NewAdmin Task**.

Figure 6–12   Submit a New Admin Task
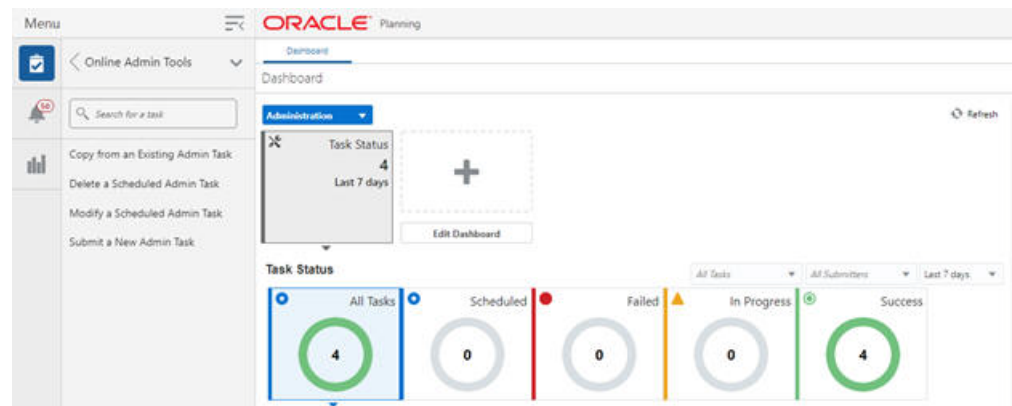
4. Select **System Admin Tasks** and click **Next**.

Figure 6–13   System Admin Tasks
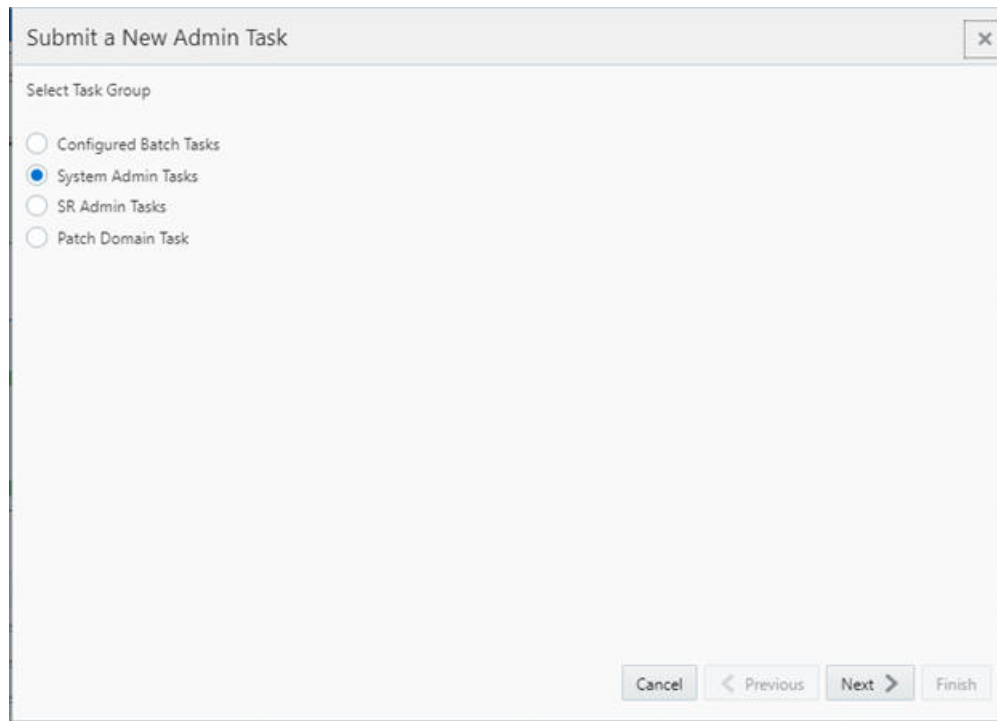
5. From the list of all available system administration tasks, select **Print Measure** and click **Next**.

Figure 6–14   Print Measure
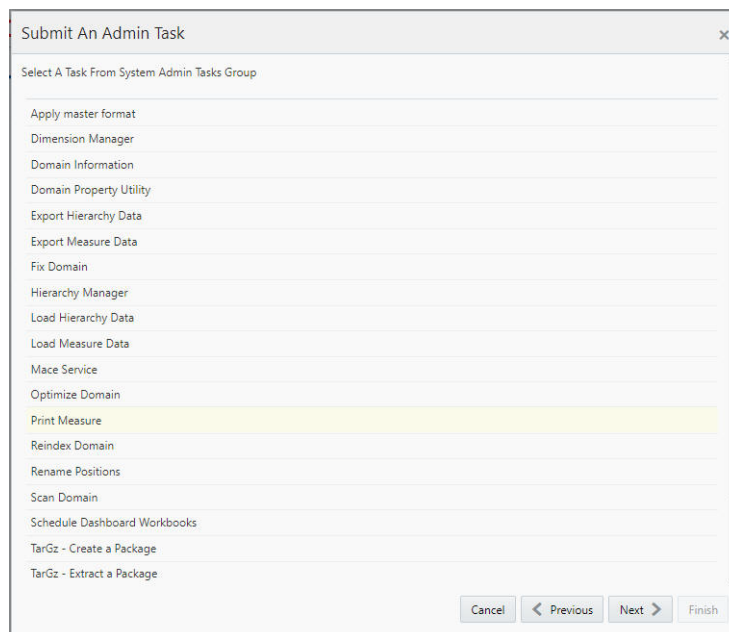
6. Enter a descriptive label in the Task Label text box.

Select a workbook from the Workbook Name drop-down list if you want to print the measure information from a workbook instead of the domain. Then select one of the following operations:

- Lists all the Measures - Returns a list all measures registered in the domain.

- List the intersection at which data is available - Returns the base intersection of the measure.

- Print specification of the measure - Returns the list of measure properties.

- List the information of a measure at its base intersection - Lists the number of dimensions, logical size, populated size, and density of a measure at its base intersection.

Figure 6–15   Print Measure Operations

Click **Next**.

7. If you selected an operation other than List all Measures, select a measure name from the drop-down list.

Figure 6–16   Measure Name

8. Click **Next**. You see Schedule Task, where you can choose to run the task ASAP or on a schedule.

Figure 6–17   Schedule Task

9. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

Click **Previous** to make changes and click **Finish** to submit the task.

Figure 6–18   Verify Selections

7

Data Administration

This chapter addresses the following tasks.

- Hierarchy Management
- Loading Hierarchies
- Exporting Hierarchy Data
- Loading Measure Data
- Exporting Measure Data
- Scan Domain Data Using Scan Domain

Hierarchy Management

The following key concepts and processes are critical to the hierarchy management process:

- Hierarchy structures are loaded into a domain using the load hierarchy data administration task.

- RPASCE uses integer indexing for simplified hierarchy administration. A set number of hierarchy positions, based on bit size, are allocated to each dimension. The pre-allocation of positions reduces the need for updating the measure data structures.

- The length of position names is 24 characters or less by default. RPASCE provides the ability to increase this length using the Dimension Manager administration task.

- Position name cannot be an empty string and must not contain any of the following characters:

    "

    (

)

    ,

    '

    \

    ^

    {

    }

[

]

!

#

=

%

 +

 `

 ctrl char

 space

ASCII uppercase characters in the position name are converted to lowercase by the application.

- RPASCE provides the ability to have placeholder positions in the domain that can be used when loading new hierarchy positions.

- RPASCE can automatically handle the movement of positions and their corresponding data between local domains when their parent-child relationships change and cause such a scenario. This is only applicable in a global domain environment.

Loading Hierarchies

The Load Hierarchy Data administration task (loadHier task) is used to load and refresh hierarchies. The loadHier task supports comma-separated values (CSV) or fixed width flat files for loading. The load file requires a .dat file extension for flat files and a .csv.dat file extension for a CSV file. When using a fixed width file, the width of the fields (number of characters) is specified in a configuration file before a domain is built. The width of fields can be increased after a domain has been built using the Dimension Manager administration task (dimensionMgr task) or by changing a property in the configuration and patching the domain. The loadHier task also provides a simple compression method that can skip duplicated values line by line.

> **Note:** This note concerns hierarchy loading with intraday:
>
> The Ride framework allows the execution of the loadHier task. Users can stay logged in and continue their activities as long as they do not perform activities such as workbook build, refresh, DPM, and custom menu scripts that touch hierarchies. The system restricts users from performing these activities while the batch process is running. Users can submit commit ASAP requests that are queued and executed at the completion of the batch process.

To manage the addition, removal, and reclassification of positions in a hierarchy, RPASCE uses a methodology called integer indexing. It is used to manage multidimensional data at the storage level.

The loadHier task supports both the loading of hierarchy positions and the purging of data in parallel. When RPASCE deletes a partition position through purging, RPASCE adjusts the cache data in parallel to maintain the correct position and domain mapping.

Multiple input files can be loaded for the same hierarchy. The extra input files must be named with a secondary extension (for example, msgs.dat.1).

The extra input files can be loaded only with the main input file. For example, you cannot load msgs.dat.1 in a separate loadHier task. Multiple files are often used when the hierarchy load data comes from different sources.

RPASCE automatically generates a backup copy of hierarchy data prior to performing a load for a hierarchy. If any type of error occurs during the load process, the hierarchy is restored from the backup copy. Note that in cases where the task is used to load multiple hierarchies at once, any hierarchies already loaded prior to the error will not be reverted; only the hierarchy in which the error was encountered will be restored.

The loadHier task stops with an error if the loadHierBk directory exists in the data directory of the domain, which indicates that a non-recoverable error may have occurred in the previous run. If this occurs, contact My Oracle Support at http://www.oracle.com/support/contact.html. The My Oracle Support team can best determine whether to delete the loadHierBk directory or copy the loadHierBk content back to the domain.
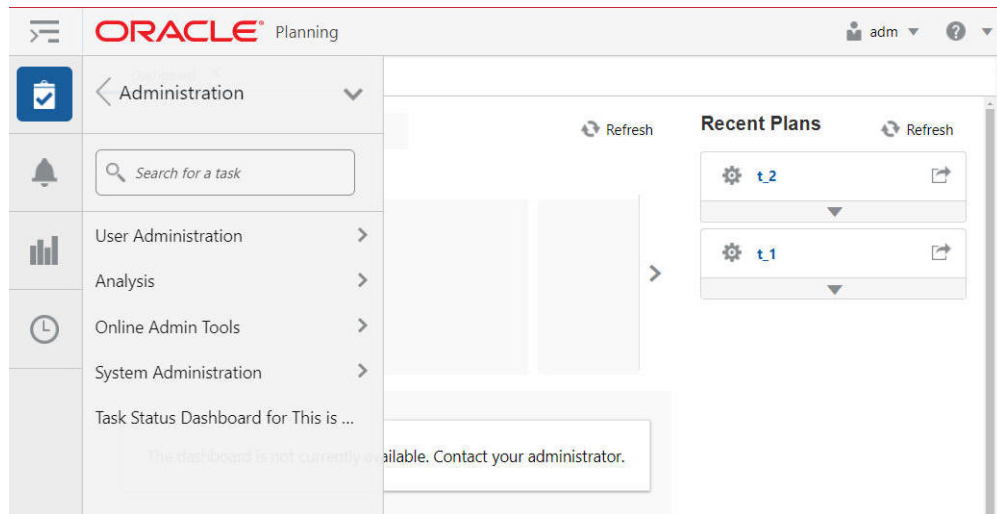
Submitting Load Hier Task

To submit a loadHier task, complete the following steps.

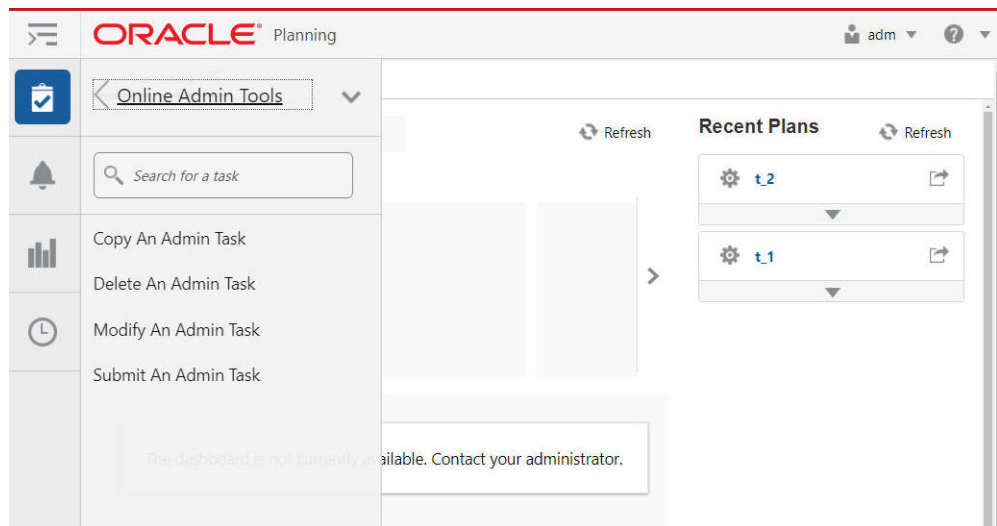1. From Task Flow, select **Administration** under **Tasks**.
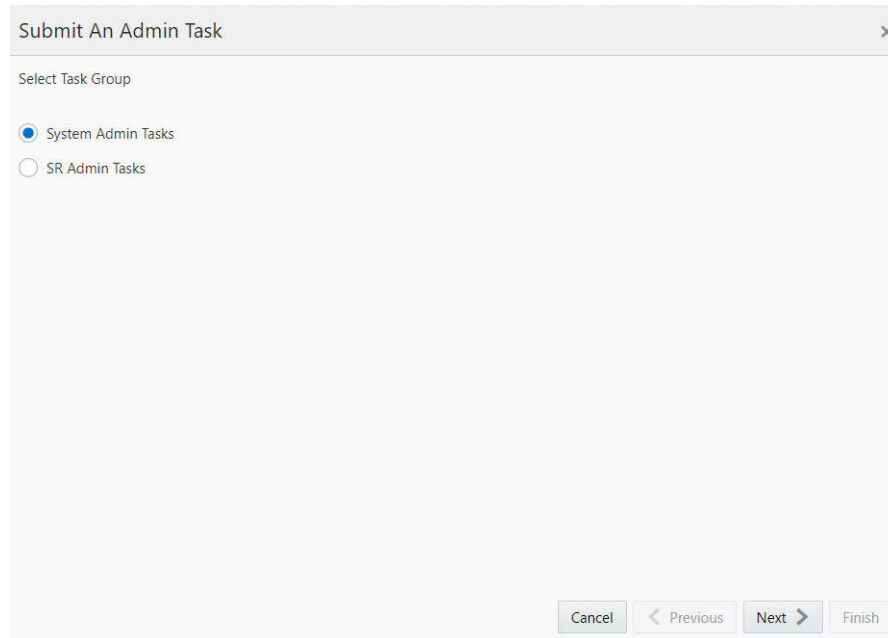
Figure 7–1   Load Hierarchy Task
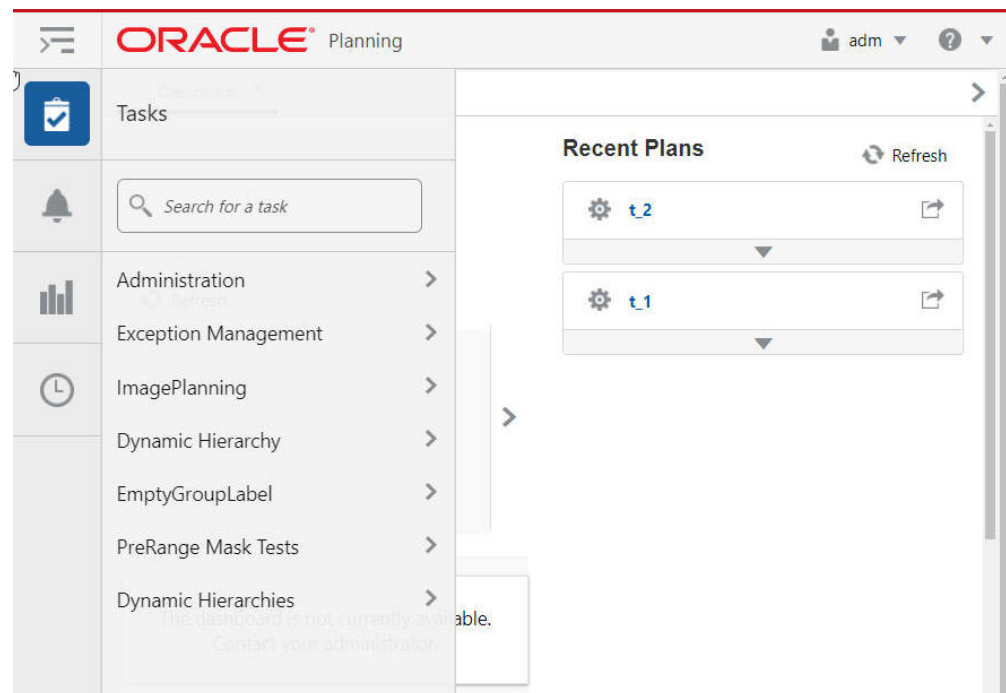
2. Select **Online Admin Tools**.

Figure 7–2   Online Admin Tools

3.   To submit a new Administration task, select **Submit An Admin Task**.

Figure 7–3   Submit an Admin Task

4.   Select **System Admin Tasks** and click **Next**.
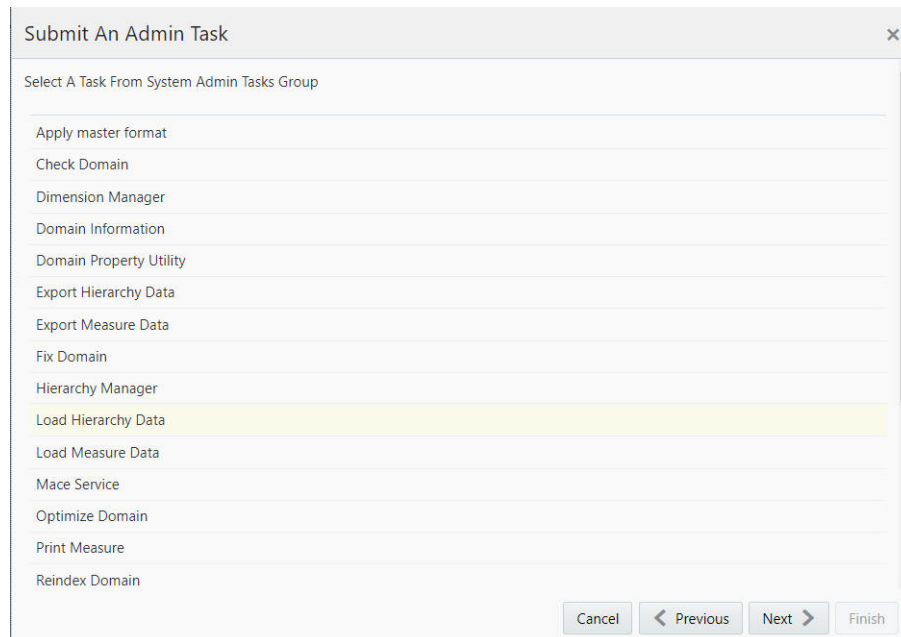
Figure 7–4   System Admin Tasks

5. From the list of all available system administration tasks, select **Load Hierarchy Data** and click **Next**.

Figure 7–5   Load Hierarchy Data

6. Specify the arguments for the Load Hier task and click **Next**.

Figure 7–6   Load Hierarchy Arguments

Arguments

- Task Label: enter a descriptive and identifiable label for the task.

- Exclusive Mode Message: specifies a message for Domain Exclusive Mode.

- RPAS_TODAY: specifies a value to overwrite the system-defined RPAS_TODAY for this Load Hierarchy run. This is optional.

- Purge Age: specifies the purge age in number of days for this Load Hierarchy run. If not specified, the task obtains the purge age from the domain. In global domains, purge age supports the purge of partition positions when the purge age is reached.

- Do not remove input files after load: if checked, it prevents the removal of input files and temporary data files that are generated during the hierarchy load process. Input files remain in the input directory of the domain after the process is completed. This option is often used for debugging or troubleshooting purposes.

- Log successfully loaded positions: when checked, the task will log successfully loaded input file lines into a loaded[HIERNAME].dat file under the processed directory.

- Enforce new hierarchy rollup changes: if checked, it enforces new hierarchy roll-up changes. New roll-up changes override or dominate existing hierarchy roll-ups if they conflict with the rollups specified in the input file. This allows you to load a hierarchy file that reclassifies one or more upper level positions while removing one or more discontinued base-level positions that roll-up to the reclassified position.

- Load user-defined positions: when checked, the task will load user-defined positions back to the domain. The data file must be in CSV format with a header line. The name of the data file must follow the current standard: <hierarchy name>.csv.dat. All user-defined dimensions must be in the data file. Any missing user-defined dimensions will cause an error. All loaded positions will have formal status.

- Force measure NA consistency: if specified, the Load Hierarchy task will enforce the measure's NA value to be cell value for newly added positions to the measure array when the measure array's NA value is different than measure's NA value.

- Operations: select one of the operations to load a specified hierarchy or load all hierarchies with .dat files present.

7. You see Schedule Task, where you choose to run the task ASAP or on a schedule.

Figure 7–7   Schedule Task

8. Click **Next** to navigate to the next page where you verify and confirm the task information. entered

Click **Previous** to make changes or **Finish** to submit the task.

Figure 7–8   Verify and Confirm
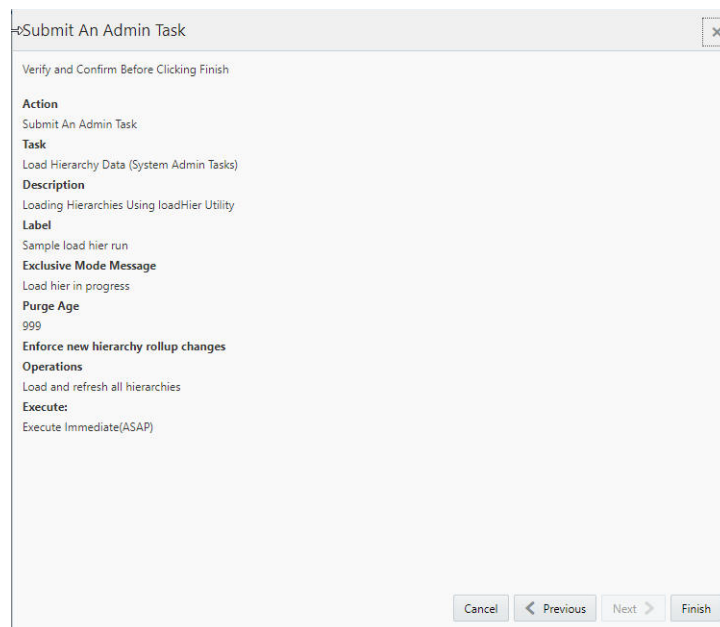
Informal Position Manager

The Informal Position Manager is a back-end domain utility that maintains informal positions for any dimension in a domain. Informal positions are those created with the Dynamic Position Manager (DPM). This utility can convert positions from formal to informal or from informal to formal. It can also remove informal positions, create informal positions in bulk, and copy data slices between positions in measures. This utility is not available directly on OAT. Instead, it is provided in the Batch Framework through the batch service catalog. Only formalizing positions are currently available with the batch service keyword "formalize".

Formalizing positions takes an input file with the following format:

- File name

  File name must be in this format: {hierName}.formalize[.extension], where the optional extension can be any arbitrary string as long as it is allowed in a file name.

  For example:

  prod.formalize.20100220

- File contents

  The file contents are in a comma-separated values format (CSV) with no header line. One position can be specified in each line with both dimension name and position name.

  For example:

  dimName1, positionName1

  dimName2, positionName2

Position Label Translation

To enable the translation of position labels for the desired dimensions, the Translate column of the Dimension definition must be set. Building or patching the domain with this configuration builds the necessary infrastructure in the domain to manage translations for those dimensions. However, label translations must be separately loaded.

Position label translations are loaded in dimension-specific translation measures for every language used by the users. If translated labels are not loaded using these measures, workbooks show position names wherever a label has to be shown. Note that for a translatable dimension, RPASCE never uses or shows the position labels from the hierarchy load file but always refers to the labels in dimension-specific translation measures. This implies that if a domain were patched to make a dimension translatable but the translation measures were not loaded, RPASCE users would see position names instead of position labels from the load file.

Dimension-specific position translation measures are named as r_<dim name>label, where <dim name> must be replaced with the name of the translated dimension. For example, if the sku dimension were to be translated, load the r_skulabel measure with translations. These measures must be loaded after loading the hierarchy because RPASCE can only load translations for already loaded positions.

The position label translation measure load files have three columns. The first column has the position names, the second column has the language identifier, and the third column has the translation for the language specified in that row.

For example, a translation measure file for the sku dimension is named r_skulabel.csv.ovr and has the content formatted, as shown in the following example. Note that the same file contains labels in four languages.

```
10006782,ENGLISH,White Nike Running Shoe size 11
10006782,CHINESE_SIMPLIF, ???? ? ??? 11
10006782,FRENCH,Taille blanche 11 de chaussure de course de Nike
10006782,ITALIAN,Formato bianco 11 del pattino corrente di Nike
10004523,ENGLISH,Black leather shoe size 8
10004523,CHINESE_SIMPLIF, ????? 8
10004523,FRENCH,Taille noire 8 de chaussure en cuir
10004523,ITALIAN,Formato nero 8 del pattino di cuoio
```

> **Note:** For a list of language identifiers, see Table 4–1. Alternatively, you can manually enter or alter translated labels using the Translations workbook in the Administration tab. In this workbook, a worksheet is available for each dimension that has translations enabled. You can manually enter translated strings for the language of interest. After they are committed, these translations are available for every new workbook.
>
> It is possible that, because of error that may have occurred when translation files were prepared, translated labels for some positions may not be loaded. In a situation where RPASCE is unable to look up the label for the locale of the machine on which the RPASCE client is being run, RPASCE looks for a non-empty label string for the English language. If it fails to find a non-empty label string for the English language, it uses or shows the loaded position name of the position.
>
> For the fixed-width format of translation measure load files, RPASCE limits the labels to 80 bytes (RPASCE uses UTF-8 encoding). For CSV format files, there is no limit. To avoid the complexity of calculating starting positions for fixed-width format files and the limitation of translation string length, use CSV files.
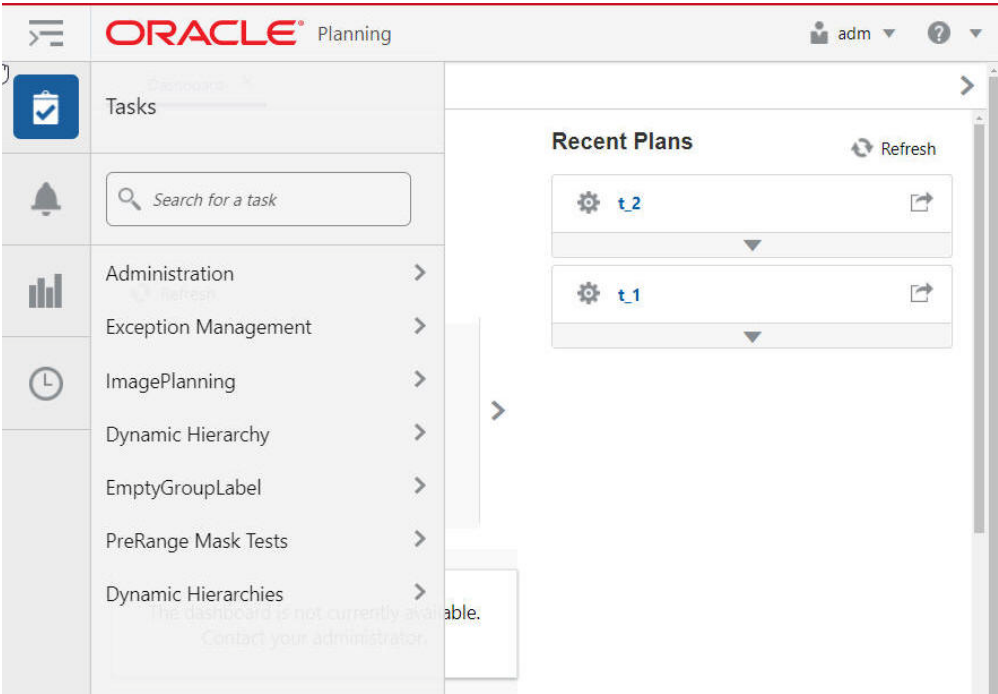
Exporting Hierarchy Data

The Export Hierarchy Data task is used to export all the positions in a hierarchy, including their rollup relations. By default, the utility assumes that the file has a CSV flat file format with fixed width format as an optional argument. The utility exports all hierarchy positions, but the file may be specified to include only formal or informal positions. The resulting file can then be used as a .dat file with a Load Hierarchy Data task.
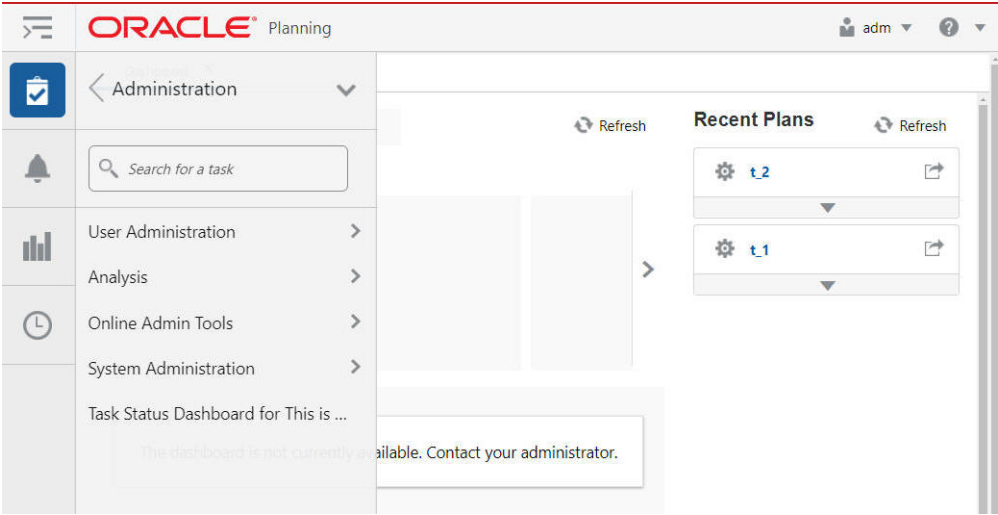
1. From Task Flow, select **Administration** under **Tasks**.

Figure 7–9   Administration

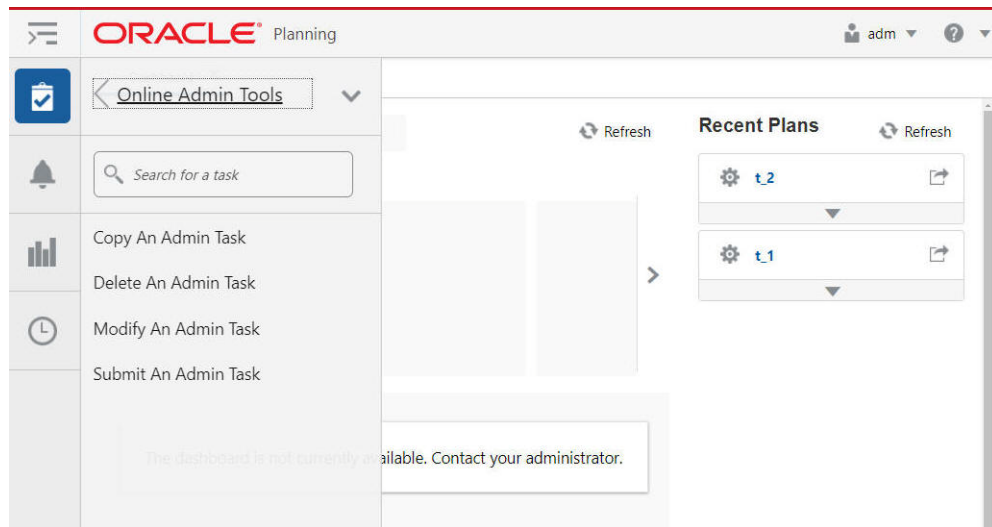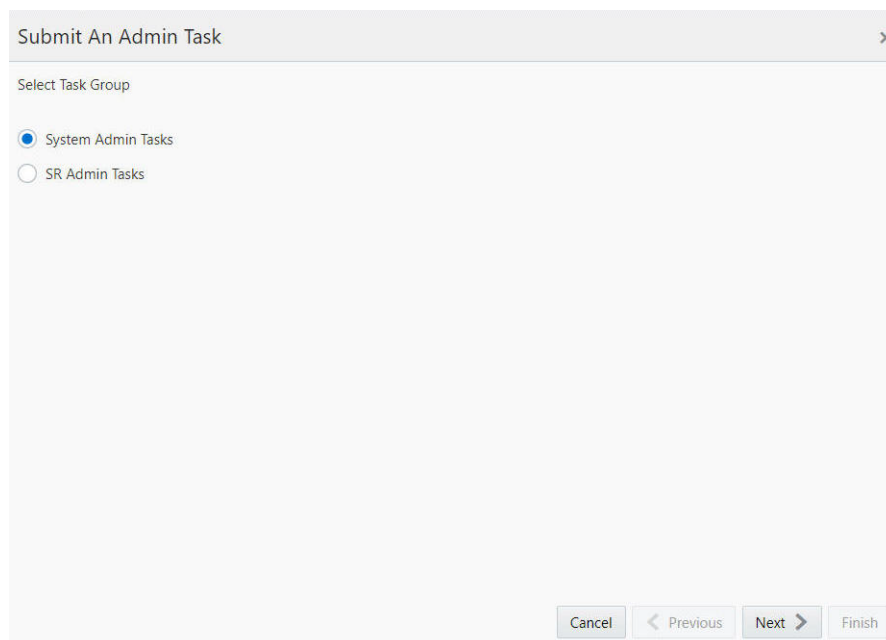2. Select **Online Admin Tools**.
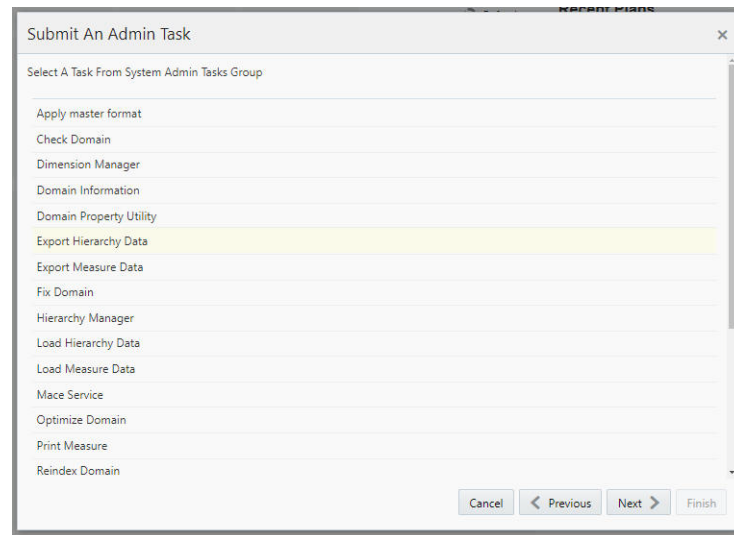
Figure 7–10   Online Admin Tools

3. To submit a new administration task, select **Submit An Admin Task**.
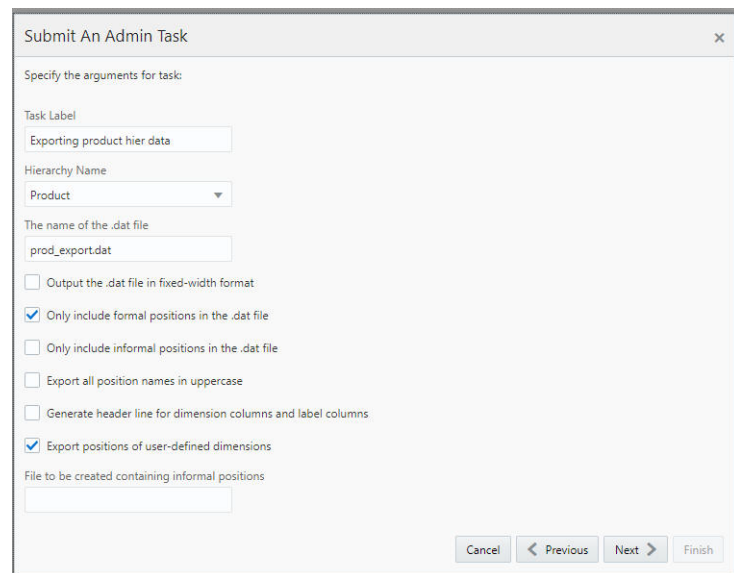
Figure 7–11   Submit an Admin Task

4. Select **System Admin Tasks** and click **Next**.

Figure 7–12   System Admin Tasks

5. From the list of all available system administration tasks, select **Export Hierarchy Data** and click **Next**.

Figure 7–13   Export Hierarchy Data

6. Specify the arguments for the task and click **Next**.

Figure 7–14   Export Hierarchy Arguments

Arguments

- Task Label: enter a descriptive and identifiable label for this task.

- Hierarchy Name: specifies the name of the hierarchy to export data.

- The name of the .dat file: specifies the output file to be created. This .dat file can be used as input file for Load Hierarchy Data tasks.

- Output the .dat file in fixed-width format: when checked, the output file will be in fixed-width format instead of comma-separated values (CSV). The width of each column is set during domain build from the configuration.

- Only include formal positions in the .dat file: if checked, the task will only export formal positions in the .dat file. Informal positions will be skipped.

- Only include informal positions in the .dat file: if checked, the task will only export informal positions in the .dat file. Formal positions will be skipped. This selection and Only include formal positions in the .dat file are mutually exclusive.

- Export all position names in uppercase: export all position names in uppercase.

- Generate header line for dimension columns and label columns: applies to CSV output file only.

- File to be created containing informal positions: specifies the file to be created in the same directory as the Output .dat file for informal positions in a format that can be used to set position back to formal status using the informalPositionMgr utility.

7. You see Schedule Task, where you can choose to run this task ASAP or on a schedule.

Figure 7–15   Schedule Task

8. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

   Click **Previous** to modify make changes or **Finish** to submit the task.

Figure 7–16   Verify and Confirm
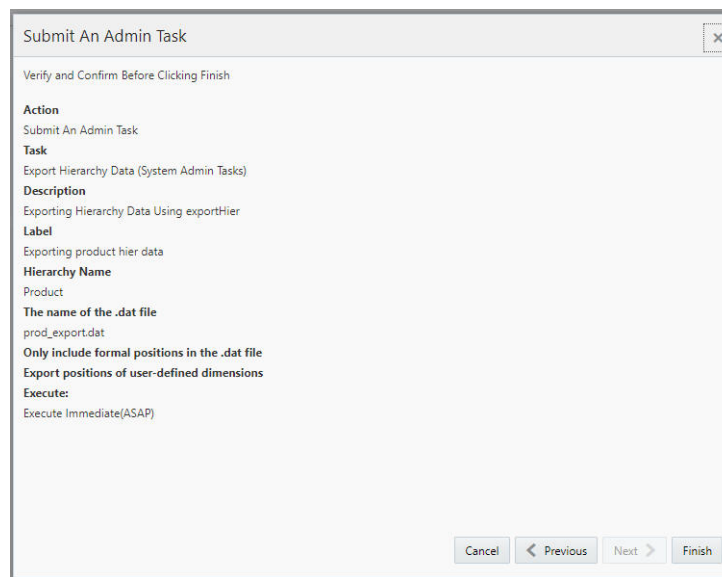
Loading Measure Data

The Load Measure Data task (loadMeasure task) is used to load measure data from text files into the domain. The administrator must specify the measure names. The Load Measure Data task supports the use of fixed width and CSV (comma separated variable) files for loading measure data. RPASCE recommends the use of CSV files to reduce the size of the load file and to reduce disk I/O time. To load measure data, system administrators must create and transfer one or more load files in the input folder of the domain directory. The administrator can then submit the Load Measure Data task to load data.

Load File Names and Load Behavior

Administrators must pay close attention to file naming. If a file name has not specifically been configured in the domain configuration, the file must be named the same as the measure name, with the appropriate extension, depending on the type of the load. For example, if the measure is named "rsal" and does not have a filename configured in the domain configuration, then the basic filename will also be "rsal". This name must be appended with one of the following extensions to indicate the type of load. If the load is an overlay, then the filename must be rsal.ovr; if it is an increment, the file name must be rsal.inc, and so on. If a CSV file is being used, then the load type extension must be prefixed with the .csv extension (for example, rsal.csv.ovr and rsal.csv.inc).

RPASCE supports the following types of loads (identified by file name extension):

- .ovr (Overlay): Existing values in the measure are overlaid with the values in the input file. Any values not included in the input file are not changed in the measure.

> **Note:** For string type measures, an empty cell in the .ovr file is treated as a valid string; as a result, the loadmeasure task overwrites the previously loaded string with an empty string. For other measure types, an empty cell in the .ovr file is treated as invalid data. It is discarded and the previously loaded value is retained.

- .rpl (Replace): The existing measure is cleared and the values in the input file are taken as the new values for the measure. Existing values for cells that do not exist in the load file are switched to NA. In other words, all data at the base intersection for the measure is removed before cells are populated with the data from the incoming file.

- .inc (Increment): Increment mode must only be used with numeric measures in which the load file contains incremental values. Therefore, if a cell had a value of 2 and the .inc file provided a value of 3 for the cell, then the new value for the cell is 5 (2 incremented with 3).

- .clr (Clear): Clear mode is a variation of replace mode. It is used when measure data is loaded in parts or staggered in time, so that data for all positions grouped by an aggregate level position is replaced if one or more positions for that group of positions are being loaded. In other words, data at the base intersection of a measure is partially cleared based on incoming data and the clearint attribute for the measure. The clearint attribute defines an intersection above the base intersection. All cells at the base intersection that are descended from a given position at the clearint level will be removed if data exists in the incoming file for at least one of those descending positions. For example, assume that there are four regions, each with several stores, and the data is loaded region by region or for a subset of regions at a time. When loading data, ensure that data for a region is completely replaced with the new load if the load file has data for one or more stores from that region; however, other regions must be left untouched. This is made possible by clear loads where the clear intersection (clearint) property of a measure specifies the aggregate level at which to group positions for completely replacing the data. In this example, the clear intersection is at the region level. Clear intersection does not have to be performed along one hierarchy, but can be performed at the intersection of multiple hierarchies. However, if you load multiple .clr files with region as the clear intersection and data for one of those regions is contained in multiple files, then the last loaded .clr file for that region will replace any information that the previous .clr files loaded for that particular region. The loadmeasure task allows more than one load file to be present in the input folder at the same time for the same measure. If more than one load file is present in the input folder at the same time, each will be loaded. Since RPASCE has a strict naming convention for measure file names, in order to add more than one load file at the same time, administrators must append the filenames as described above with file-distinguishing extensions. For example, with the file names rsal.csv.ovr.1 and rsal.csv.ovr.2, RPASCE does not care about the form of the multi-file extension. The extensions can be anything, number or text, and RPASCE will still load them.

The loadmeasure task also allows multiple types of load files to be present in the input directory at the same time. RPASCE loads .rpl files first, then .clr, .ovr, and .inc files. Since .rpl files completely erase existing measure data and then load the given data, you must not have multiple .rpl files at the same time.

Loading Multiple Measures from One File

The loadmeasure task allows multiple measures to be loaded from a single file. You can load measures from CSV files or fixed width files.

CSV Files

If a CSV file is used for loading multiple measures, loadmeasure task will use the start positions of measures as defined in the Data Interface Tool for the fixed-width file format to determine the order of columns in the CSV format. For example, if a file named multiple is used to load measures A, B, and C, where the start position (for fixed-width file format) for the measure values have been configured to be 40, 110, and 70, respectively, then when using the CSV file multiple.csv.ovr, loadmeasure will assume that after the dimension columns, the first column is measure A, then C, and then B, because 40 (A) is less than 70 (C) is less than 110 (B).

It is not necessary to load all measures in a multiple measure file. Administrators can choose to load only a subset of measures from the multi-measure file. However, if administrators wish to reuse the same file for loading different measures at different times in a batch, they must select the Do not remove input files option to ensure that the loadmeasure task does not move the file to the processed folder after processing the first load request.

> **Note:** Although it is not required to specify all measures contained in the multi-measure CSV file in a single loadmeasure command, there is no way to skip data columns in the CSV file.

Fixed Width Files

With a fixed width file, a single measure's data can be loaded from a file containing multiple measures.

Loading Data from Below the Base Intersection of the Measures

The loadmeasure task supports loading measure data from an intersection lower than the base intersection of the measure. The load intersection has to be pre-specified in the configuration (loadint property), and the load time aggregation (loadagg property) method must also be specified.

When loadmeasure loads data from below the base intersection, all low-level data corresponding to a cell at the base intersection must be available in the load file for RPASCE to be able to correctly aggregate the low-level data to the base level. A mistake in the values of a subset of cells that aggregate up to one cell at the base level can only be corrected by reloading the data for all low-level cells that correspond to the cell at the base level. If any low-level cells are missing, RPASCE replaces their value with NA.

To perform a lower level load, RPASCE first aggregates the data and then applies the appropriate load type to update the measure value, overwriting the existing value with the aggregate of the input cells if .ovr files were used, or incrementing the existing value with the aggregate of the input cells if .inc files were used.

Staging Measure Loads

RPASCE supports the notion of stage-only measures. For stage-only measures, loadmeasure queues the loaded data in an intermediate staging area, but does not load it into the measure until it is called with the Apply staged loads parameter. For

stage-only measures, loadmeasure must be called twice, once to stage the measures and then with the Apply staged loads parameter to subsequently load the staged data in the measure arrays. The loadmeasure task cannot simultaneously stage loads and apply the staged loads.

Measure staging must be performed when measure data can arrive from different sources, in different load formats, and staggered in time, when system administrators want to queue all these loads up and apply them at once while honoring the data arrival queue. Measure staging can be performed while the system is online, as it does not cause measure data-related contention. (It has the potential to cause metadata-related contention.) When staging measure data, loadmeasure splits the data and purges the data files if data purging is enabled; it does not purge measure data until the loads are applied. This staging time preprocessing significantly reduces the load time when the loads are actually applied.

> **Note:** The replace (.rpl) format cannot be used for staging. Furthermore, data loads from below the base intersection of the measure cannot be staged.

Running Pre-Load or Post-Load Scripts

The loadmeasure task provides the ability to automatically run scripts before and after the data load. These are referred to as pre-processing and post-processing scripts.

When loadmeasure is called, it checks for the existence of scripts named pre<measurename>.sh in the ./scripts directory of the domain. If the scripts exist, they will be run prior to the execution of the utility. Similarly, after the utility has completed running, it checks for the existence of scripts named post<measurename>.sh and executes them.

When multiple measures are loaded in a single call, only the pre-processing script for the first listed measure has any effect on the data.
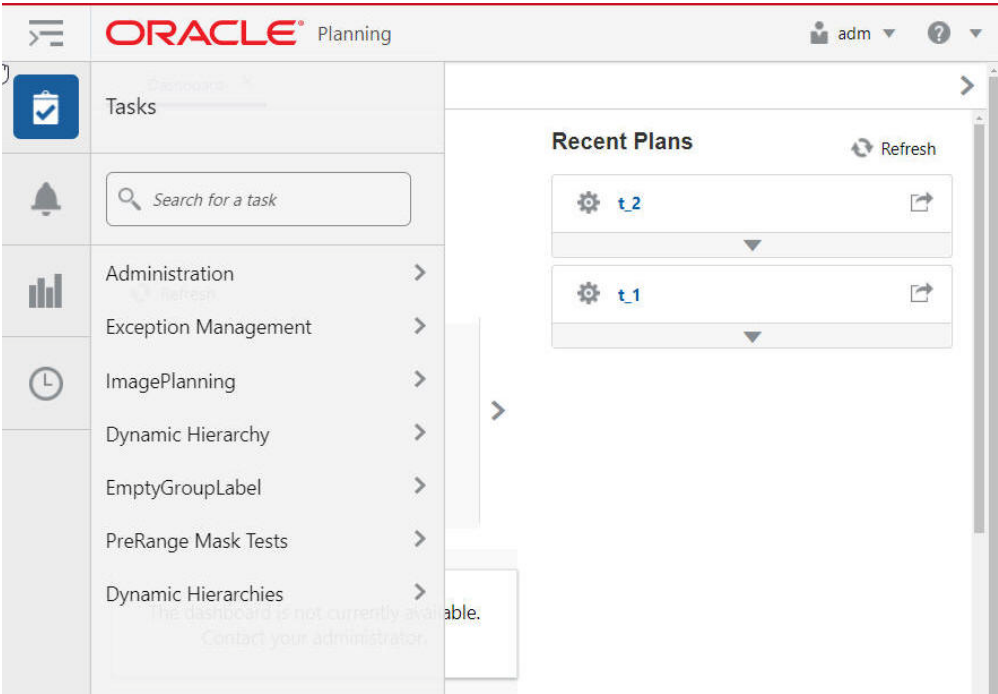
Purging Old Measure Data

System administrators can purge old measure data during a load. When the base intersection of a measure involves the Calendar hierarchy, the setting for the purge age measure property defines how and when existing data gets purged to a NA value. If the purge age has not been set, the data is never purged. If a purge age of zero or more has been set, data is purged for all dates before RPAS_TODAY - purge age days. That is, if the purge age is 5, then at data load time, all data that is older than 5 days before RPAS_TODAY will be purged.

Submitting a Load Measure Task
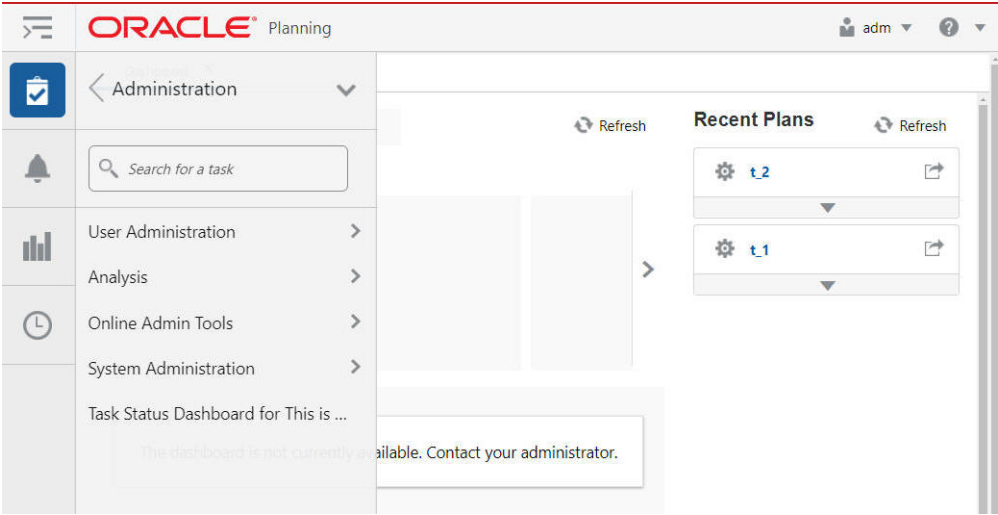
To submit a load measure, complete the following steps:

1. From Task Flow, select **Administration** under **Tasks**.
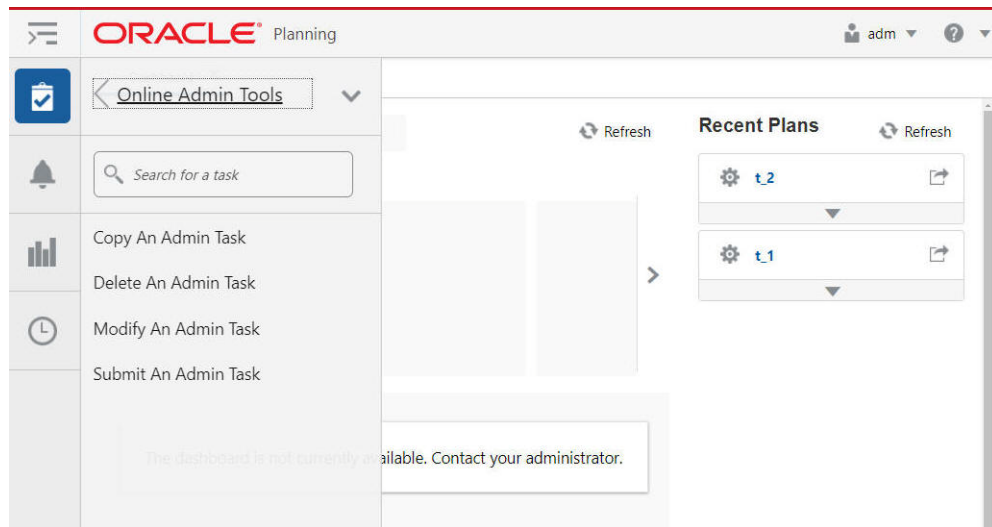
Figure 7–17   Administration
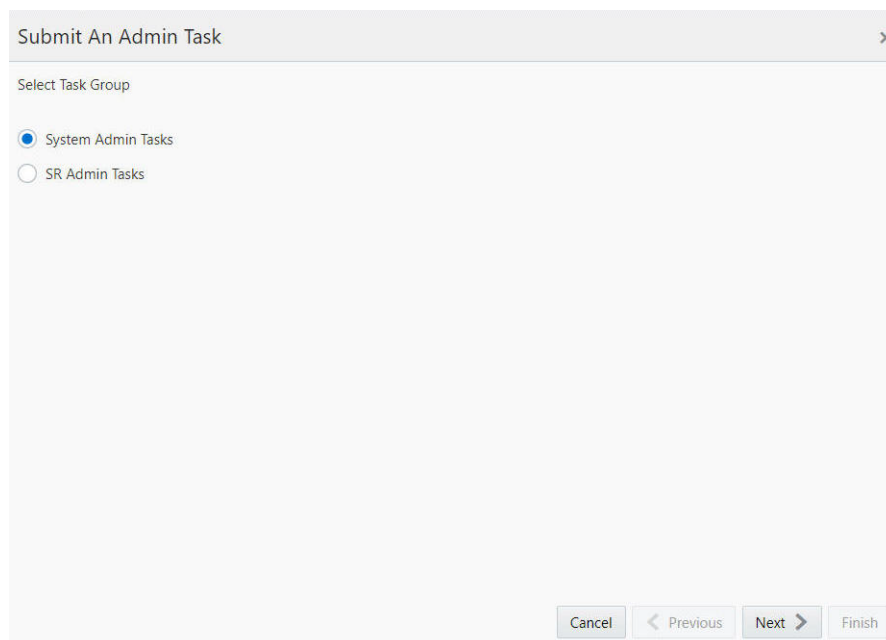

2. Select **Online Admin Tools**.
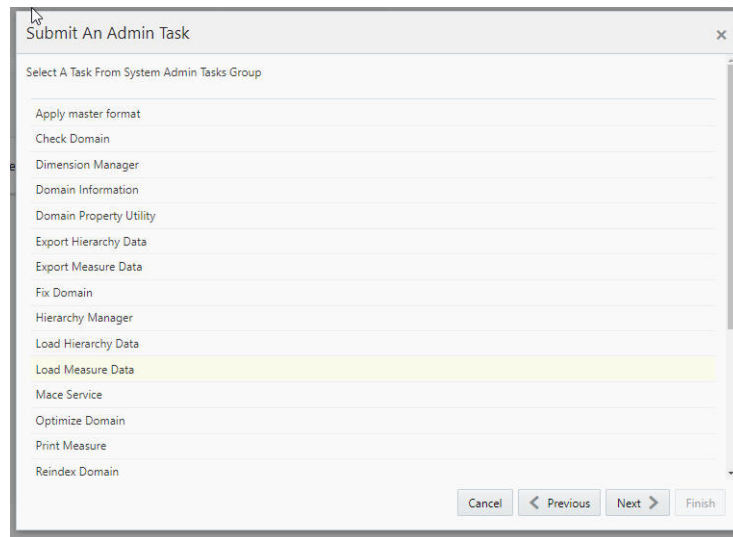
Figure 7–18   Online Admin Tools

3. To submit a new Administration task, select **Submit An Admin Task**.
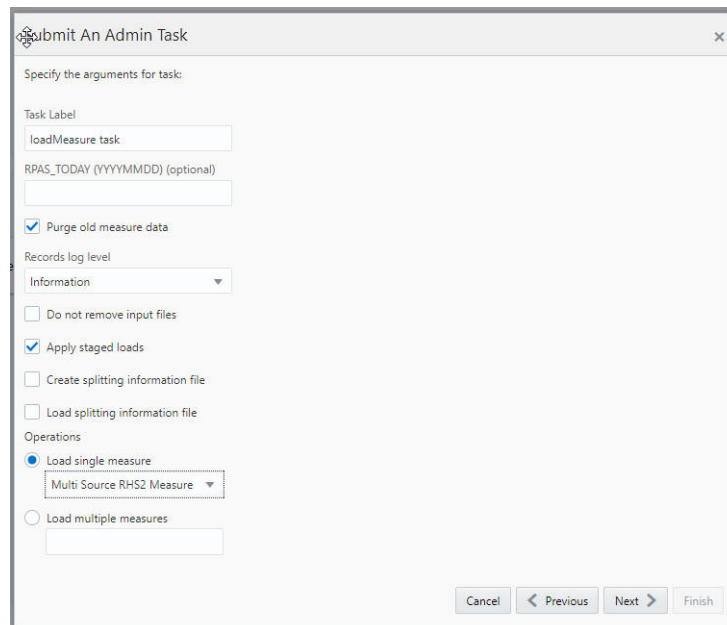
Figure 7–19   Submit an Admin Task

4.   Select **System Admin Tasks** and click **Next**.

Figure 7–20   System Admin Tasks

5.   From the list of all available system administration tasks, select **Load Measure Data** and click **Next**.

Figure 7–21   Load Measure Data

6. Specify arguments for the load measure task and click **Next**.

Figure 7–22   Load Measure Arguments

Arguments

■   Task Label: enter a descriptive and identifiable label for this task. This label is displayed on the Task Status Dashboard.

■   RPAS_TODAY: enter a date (in YYYYMMDD format) if you want to overwrite the system-defined RPAS_TODAY. The value only applies to this task and is optional.

■   Purge old measure data: if checked, the task will purge old measure data even if no new data is loaded. This option can be applied to stage-only measures without having to apply loads. When a measure has the Calendar hierarchy in its base intersection, the setting for the purgeAge measure property defines

how and when existing data is purged to a NA value. If purgeAge has not been set, the data is never purged. If a purge age of zero or more has been set, data is purged for all dates that are before RPAS_TODAY - purgeAge days. That is, if purgeAge is five, at data load time all data that is more than five days before RPAS_TODAY will be purged.

- Records log level: specifies the log level for any per record logs. Record logs include messages indicating whether the record line was shorter than expected, a data column was missing, one or more positions were missing, a record could not be parsed, or there were data conversion errors.

- Do not remove the input files: when checked, the task will not move the input files to the processed directory. This option is used when a single file is used to load multiple measures, but not all measures from the file are loaded at once. This option instructs the task to leave the load file behind for subsequent loading of unloaded measures. The user can use this option to perform intermediate processing between measure loads from the same file.

- Apply staged loads: if this option is selected, the task will apply any staged loads for the named measure. If the measure is registered to be a stage-only measure, loadmeasure will put the load in a staging area but will not update the measure until loadmeasure is called again with this argument. When this argument is used, loadmeasure applies all loads that have been queued up in the staging area. It clears out the staged loads unless the measure's loadsToKeep property has been set to a non-zero number. In that case, it does not clear out the latest loadsToKeep loads. Note that only .ovr, .inc, and .clr loads can be staged. The .rpl loads cannot be staged. Additionally, staging is only allowed for base intersection loads. RPASCE cannot stage loads where the load intersection is below the base intersection of the measure. This argument must not be used for measures that are not stage-only.

- Create splitting information file: causes the input files in the global domain to be split across the local domains, but does not do any further processing of the input files. Subsequently, a load measure task can be used with the load splitting information file argument to load these pre-split input files into the local domains. File-splitting is a fairly time consuming activity and can consume up to 80 percent of the load time. System integrators may be able to improve batch performance by breaking away file-splitting from actual measure loading. This is useful if a multi-measure file is being used in such a way that subsets of measures are loaded at different steps in a batch process.

- Load splitting information file: loads the pre-split input files (created by the Create splitting information file) into the local domains.

- Operations: specifies the loading of single or multiple measures. To load a single measure, select the radio button of Load single measure and a measure from the drop-down list. To load multiple measures, select Load multiple measures and enter a comma-separated list of measure names.

7. You see Schedule Task, where you can choose to run the task ASAP or on a schedule.

Figure 7–23   Schedule Task

8. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 7–24   Verify and Confirm

Exporting Measure Data

The Export Measure Data task can be used to export domain or workbook measure data from RPASCE in CSV file format. A single measure or multiple measures may be exported based a specified intersection. If the intersection is not specified, then the base intersection of the measure will be used. If the measure's base intersection is not

the same as the export intersection, the measure's default aggregation method will be used to aggregate data to an intersection higher than the base intersection, or replication will be used for spreading measure data if the data is required at an intersection lower than the base intersection.

The Export Measure Data task supports the export of data in a user-specified range, which can be a single mask measure, a range specified on the Calendar dimension, or a combination of the two.

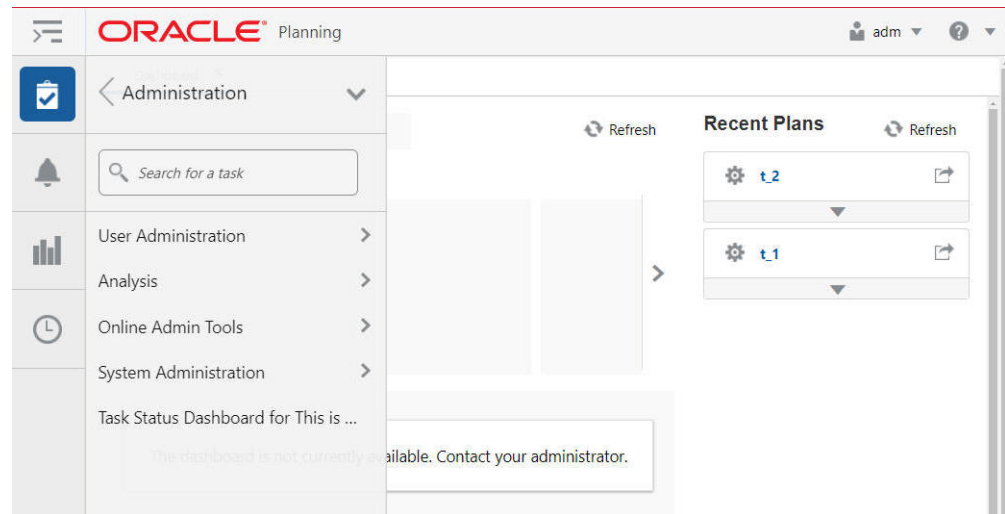## Submitting an Export Measure Data Task

To submit an export measure data task, complete the following steps:

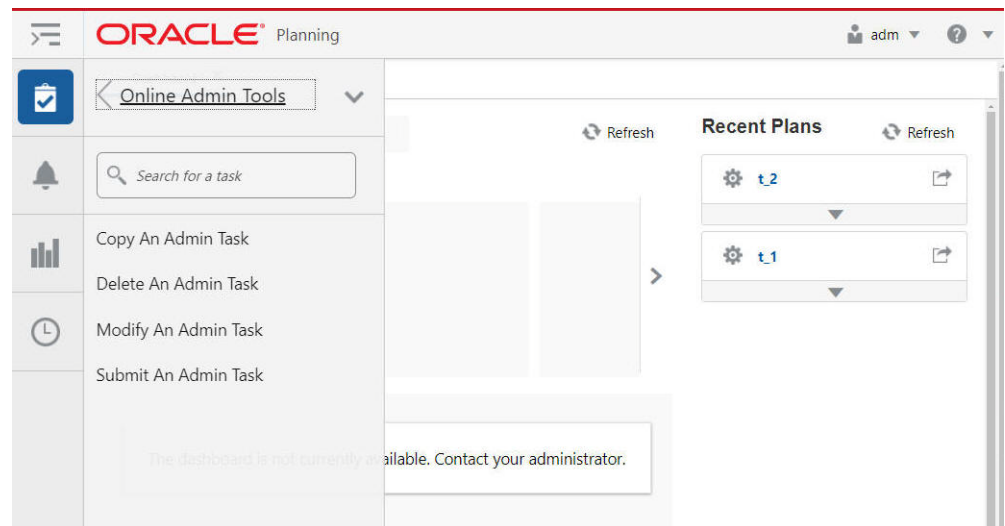1. From Task Flow, select **Administration** under **Tasks**.

Figure 7–25 Administration
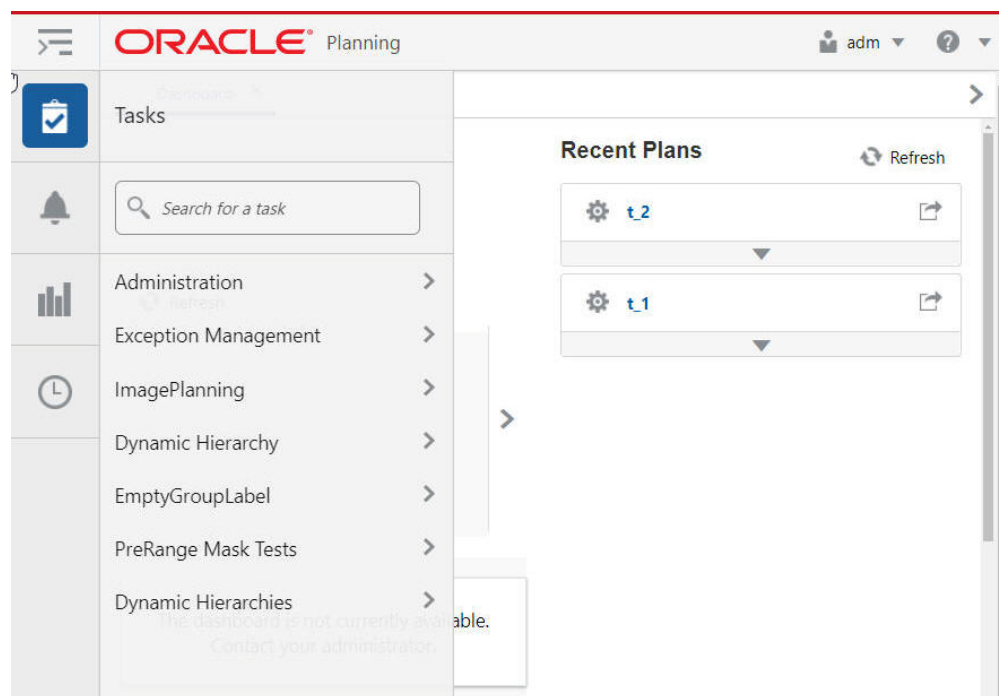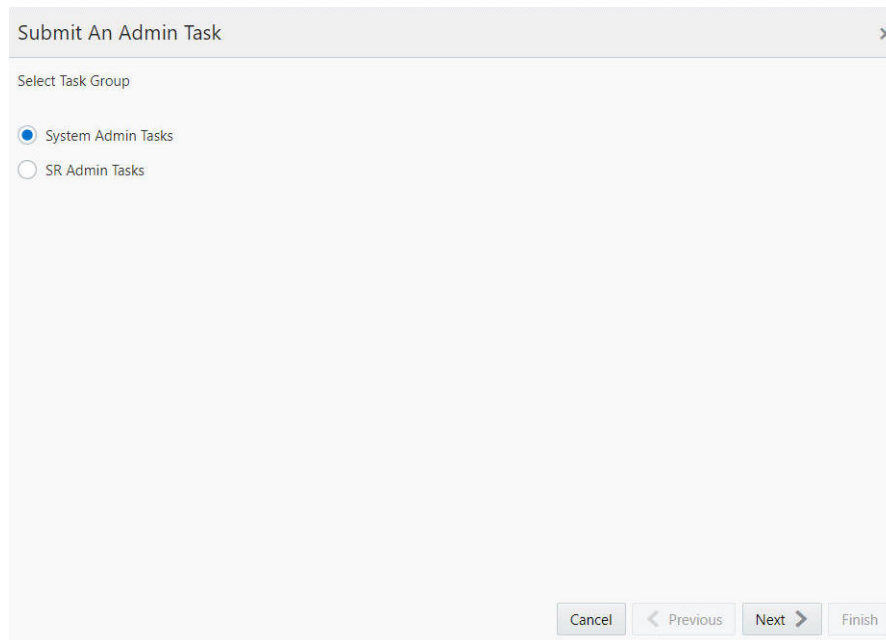

2. Select **Online Admin Tools**.

Figure 7–26   Online Admin Tools

3. To submit a new administration task, select **Submit An Admin Task**.

Figure 7–27   Submit an Admin Task

4. Select **System Admin Tasks** and click **Next**.
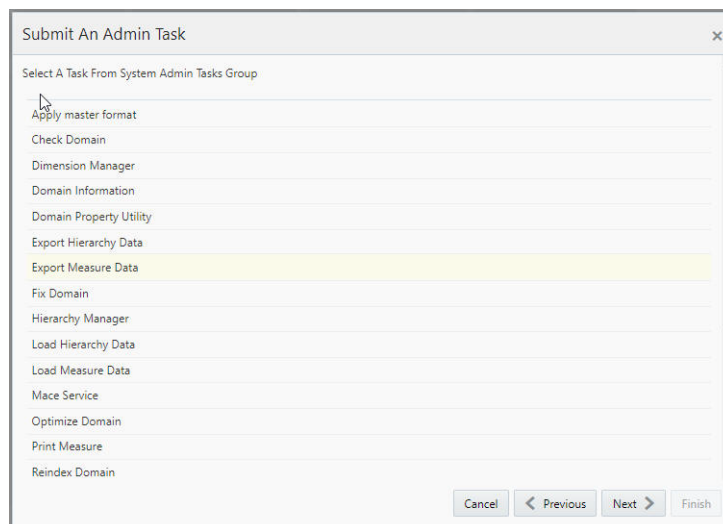
Figure 7–28   System Admin Tasks

5. From the list of all available system administration tasks select **Export Measure Data** and click **Next**.

Figure 7–29   Export Measure Data

6. Specify arguments for the export measure task.

Figure 7–30   Export Measure Arguments

Arguments

- Task Label: enter a descriptive and identifiable label for this task. This label is displayed on the Task Status Dashboard.

- Workbook Name: select a workbook from the drop-down list if you want to export measure data from a workbook. This argument is optional. Leave it unspecified if you want to export measure data from the domain.

- Export Intersection: specifies the intersection at which to export measures. If the measure's base intersection is higher than the export intersection, replication is used to spread the measure down to the export intersection. If the measure's base intersection is lower than export intersection, the measure's default method (defagg) is used for aggregation. The export intersection must be either at, above, or below the base intersection of the measure. The export intersection cannot have some dimensions above the dimension in the base intersection of the measure and some below. The RPASCE dimension names in an intersection must be four characters in length. If an RPASCE dimension name is less than four characters long, then an underscore character ("_") must be used as a filler at the end of a dimension name. This argument is optional.

- File name for data output: specifies the data output file name. If Export Intersection is not specified, all measures must be at the same intersection.

- Append new output to current output file: appends new output to the current output file. If not specified, the current output file will be erased and replaced with new data.

- Export position names in upper case: if this option is selected, all position names will be converted to upper case in the output file; if not selected, position names will be in lowercase, since they are stored in lowercase in the domain.

- Specify a mask measure name: specifies a mask measure that must be a valid Boolean measure registered. In the current measure store, its base intersection must be at the same export intersection.

- Specify the range of positions (start:end): specifies a range of positions along the innermost dimension. Only values in the range are considered for export.

- Operations: choose to export single or multiple measures. For a single measure, select one from the drop-down list; for multiple measures, enter a comma-separated list of measure names.

7. Click **Next**. You see Schedule Task, where you can choose to run this task ASAP or on a schedule.

Figure 7–31   Schedule Task

8. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 7–32   Verify and Confirm

Scan Domain Data Using Scan Domain

The Scan Domain online administration task is a domain task used for detecting data loss and repairing data corruption in an RPASCE database.

Data loss occurs when an RPASCE process is abnormally terminated. This can happen when an external mechanism, such as a power failure, causes a sudden termination of an RPASCE process. Data loss can also occur due to an unexpected program breakdown.

Data corruption can occur if an external program modifies the RPASCE database files or an unforeseen defect occurs in the processes using the RPASCE database (an extremely rare event).

The Scan Domain online administration task can detect both corruption and data loss, but it can only fix corruption.

While the utility is attempting to perform a repair of the databases, it can use a operation (backup database files) to enable backing up the original databases. While running in detection mode (check the data loss or database corruption option), the utility does not change any of the RPASCE databases, and therefore, it does not create such backups.

In detection mode, the task logs a list of databases with data loss or data corruption to the online administration task log file.

Scan DomainUsage

The online administration task can detect database data loss and corruption. It can also be used to repair corruption. Data loss can only be detected.

If the user intends to detect both corruption and data loss, it is more efficient to run the task once with both the options check database corruption and data loss.

When running the Scan Domain online administration task to detect unused data, the user sees a list of databases that may not be needed by the domain. This information includes:

- User directories for users who are not registered in the domain

- Measure databases whose corresponding measures have been removed from the domain

- Other databases not referenced by the domain (for example, measure load databases, backup databases, and temporary databases)

Managing Scan Domain

Use the Scan Domain online administration task to detect corruption in the domain.

To use Scan Domain:

1. From Task Flow, select **Administration** under **Tasks**.

Figure 7–33 Administration
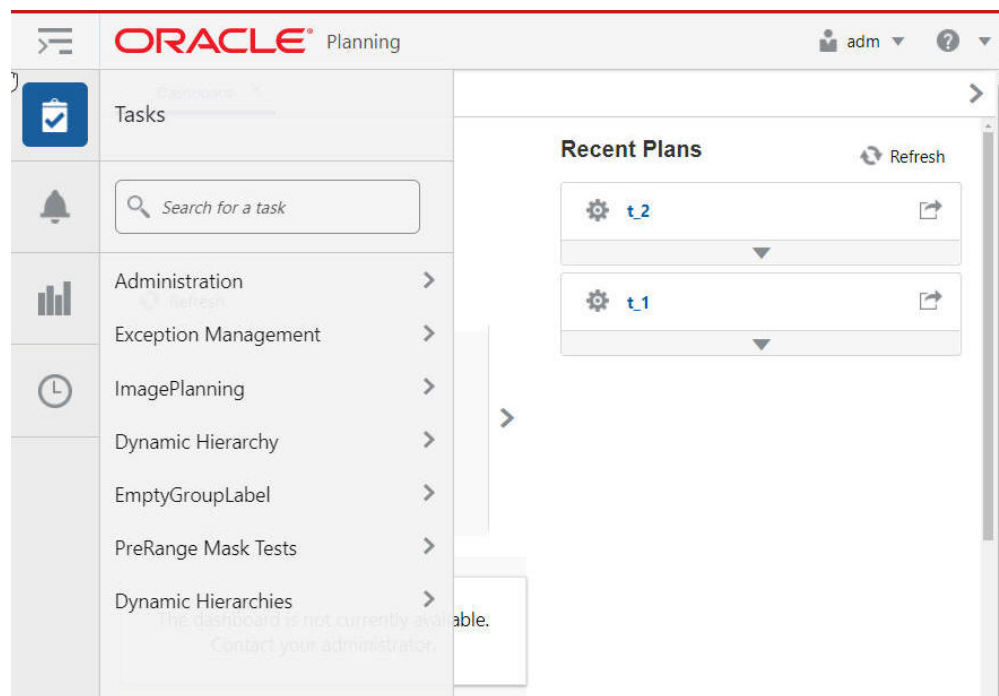
2. Select **Online Admin Tools**.

Figure 7–34   Online Admin Tools
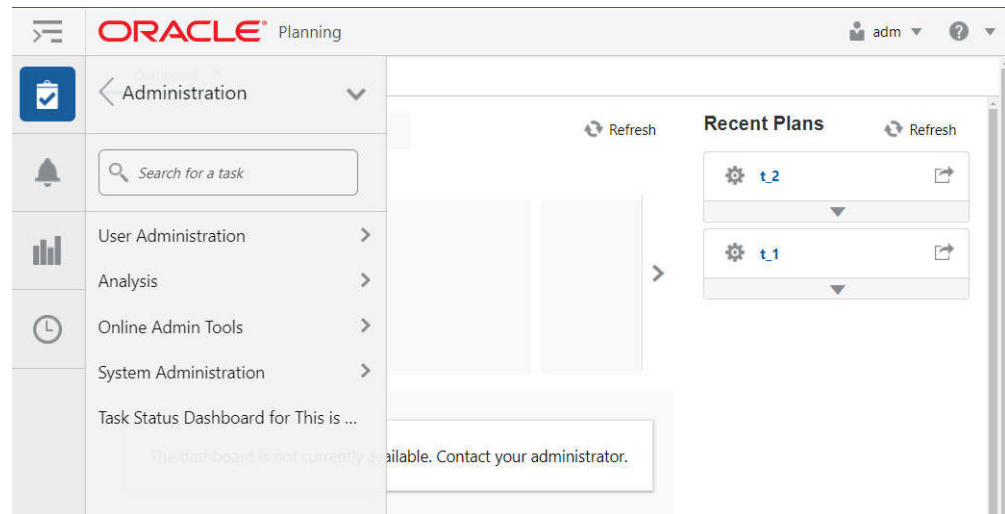
3. To submit a new Administration task, select **Submit a New Admin Task**.

Figure 7–35   Submit an Admin Task
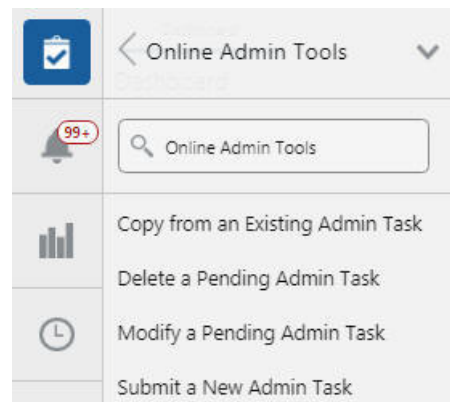
4. Select **System Admin Tasks** and click **Next**.

Figure 7–36   System Admin Tasks
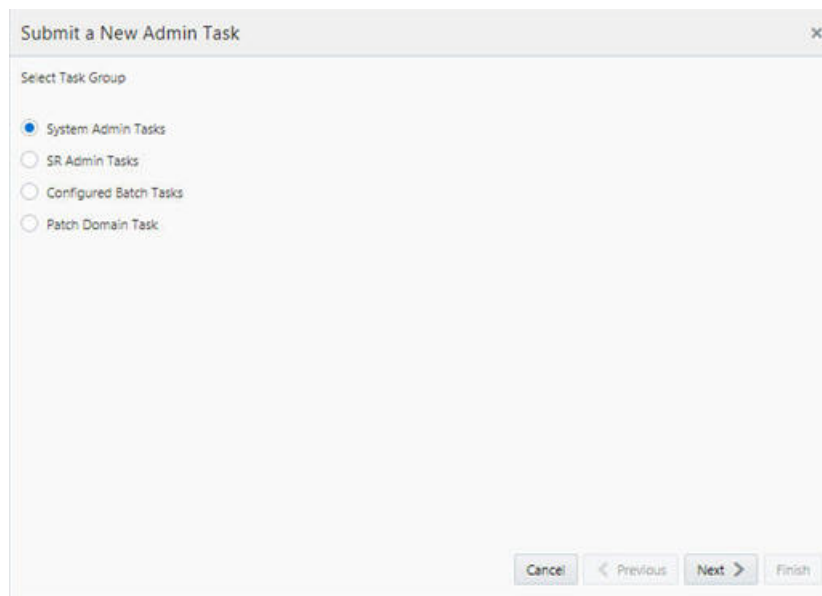
5. From the list of all available system administration tasks, select **Scan Domain** and click **Next**.

Figure 7–37   Scan Domain
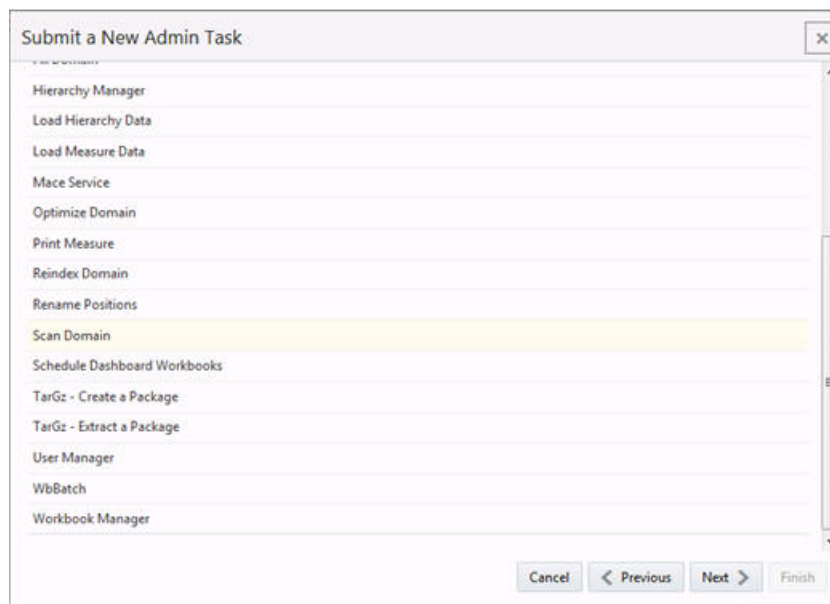
6. Provide a task label, select **Repair database corruption**, and click **Next**.

Figure 7–38   Repair Database Corruption

7. Select **Backup database files** or **Skip folder containing user information** and click **Next**.

Figure 7–39   Task Arguments

8. You see Schedule, where you can choose to run the task ASAP or on a schedule.

Figure 7–40   Run ASAP

9. Click **Next** to navigate to the next page, where you verify and confirm the task information entered

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 7–41   Verify and Confirm

10. To view the status/log file, click the task label.

Figure 7–42   Notifications
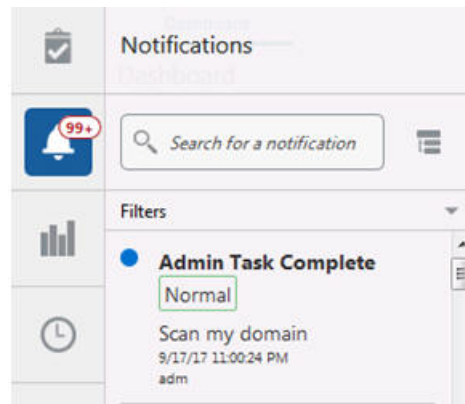
11. To view the Online Administration Dashboard, navigate to Administration task flow and click the Task Status Dashboard Sub Task.

Figure 7–43   Status
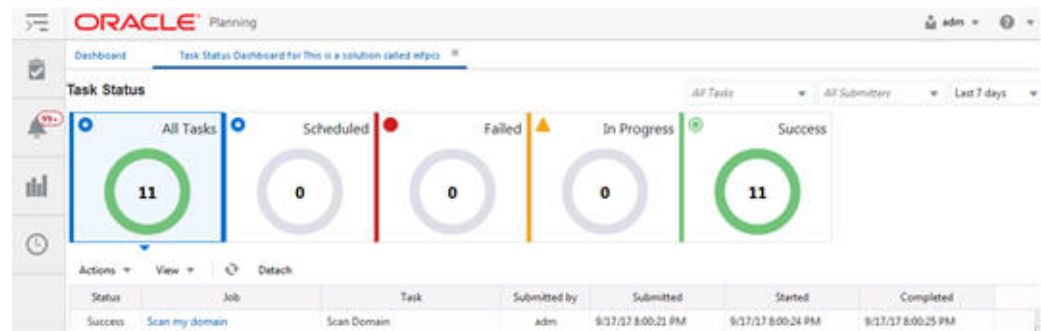
Table 7–1 provides descriptions of the arguments used by the scanDomain utility.

Table 7–1   Scan Domain OAT Arguments

Argument	Description
Check for data loss	Checks for data loss in the specified domain.
Check database corruption	Checks for database corruption in the specified domain. Detects array corruption caused by application issues
Repair database corruption	Detects and repairs the database corruption on the domain caused by application operations (batch and online). Allows uses to skips the folder containing user information while scanning and take backs up database files before repairing them.

> **Note:** Scan Domain is enhanced to self heal the BTree database from array issues caused by application runs. Issues that were previously detected by Fix Domain can now be detected and fixed by Scan Domain.
>
> The scan is based on the UReport. This report is generated at the beginning of a online administration task job run and is deleted after the run completes. The presence of the UReport indicates that the job run had an error and requires a scan. Scan Domain is enhanced to detect and fix array corruption that is currently identified by Fix Domain.

Fix Domain

The Fix Domain online administration utility is used to detect many types of corruption that can occur within an RPASCE domain. This utility can be used to perform a number of tasks, which are described in this section. In all cases, Fix Domain can execute to perform an examination to detect problems.

Remove Inconsistent Measure Attributes

It is possible for inconsistencies in measure attributes appear in the domain that interfere with operations performed by RPASCE. By executing Fix Domain with the Clean Measure Attributes operation, users can analyze a domain for such discrepancies.

Remove Partially Unregistered Measures

It is possible for the process of unregistering a measure to exit without completely removing the measure from the domain. By executing Fix Domain with the Clean Measure operation, users can analyze a domain for partially unregistered measures.

Repair Discrepancies in Hierarchy Information

It is possible, in integer-indexing domains, for the hierarchy and dimension information of an array to become out of synch with the information of the domain. By executing Fix Domain with the Repair Hierarchy operation, users can analyze a domain for de-synchronized dimension information in the domain.

Repair Unlinked Arrays

In pre-integer indexing domains, discrepancies may exist between the dimension information in an array and the dimension information of the domain. When these corrupted domains are converted to integer indexing, these arrays are converted to rely on internal dimension information and cannot be resolved against the dimension information of the domain. By executing Fix Domain with the Fix Arrays operation, users can analyze a domain to detect these unlinked arrays.

Hierarchy and Dimension Information

In general, this type of discrepancy is a result of a domain that did not build correctly. Due to the severity of this type of problem, Fix Domain cannot resolve such issues.

Measure Information

It is possible for measure information to appear in some but not all the metadata arrays of a domain. Depending on which arrays contain information about a measure, Fix Domain may be able to report the problem.

Workbook Information

It is possible for information about workbooks, workbook templates, and workbook template groups to appear in some but not all of the metadata arrays of the domain. Depending on which arrays contain information about the workbook, template, or group, Fix Domain may be able to report the problem.

User Information

It is possible for information about RPASCE users and user groups to appear in some but not all of the metadata arrays of the domain. Depending on which arrays contain information about the user or group, Fix Domain may be able to report the problem.

Managing Fix Domain

Use the Fix Domain task to detect corruption in the domain.

To use Fix Domain, complete the following steps:

1. From Task Flow, select **Administration** under **Tasks**.

Figure 7–44   Administration
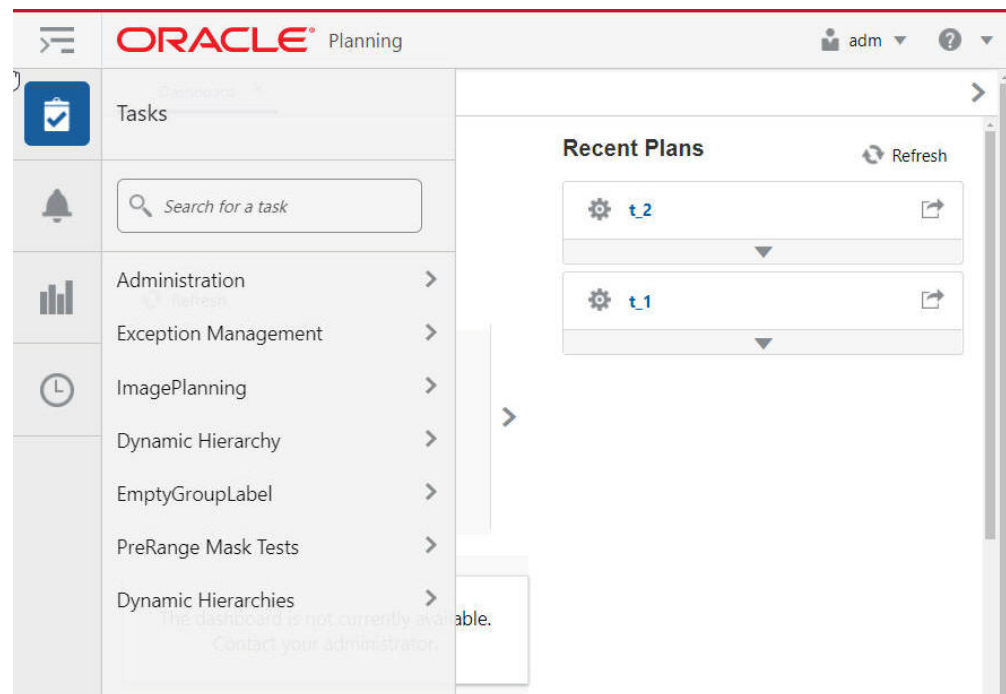
2. Select **Online Admin Tools**.

Figure 7–45   Online Admin Tools
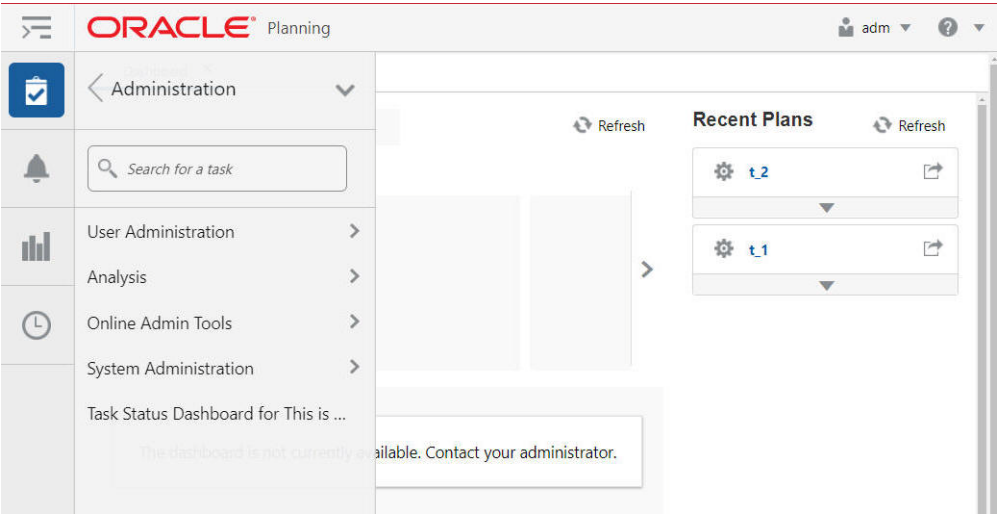
3.   To submit a new Administration task, select **Submit a New Admin Task**.

Figure 7–46   Submit an Admin Task
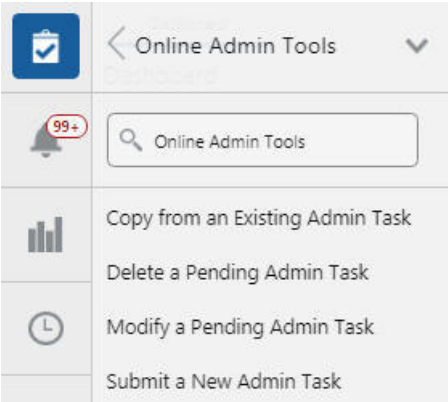
4.   Select **System Admin Tasks** and click **Next**.

Figure 7–47   System Admin Tasks
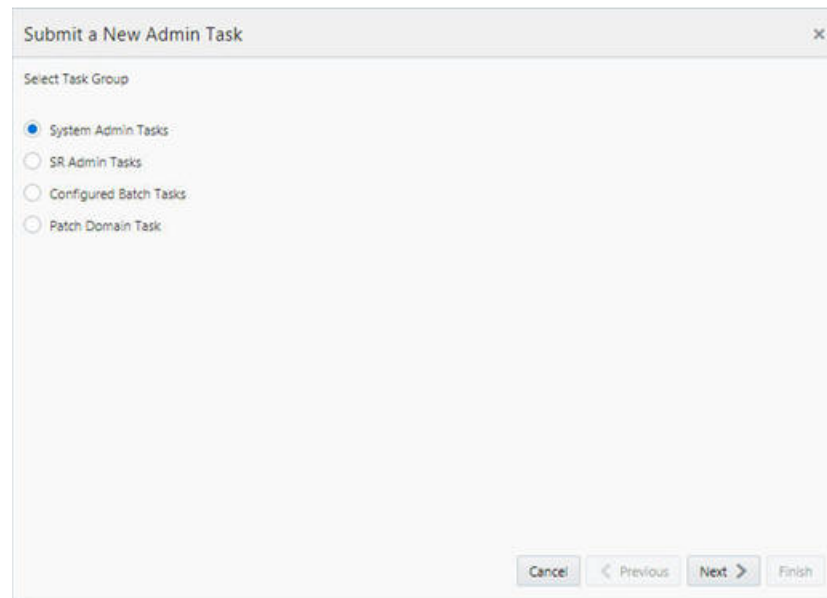
5. From the list of all available system administration tasks, select **Fix Domain** and click **Next**.

Figure 7–48   Fix Domain
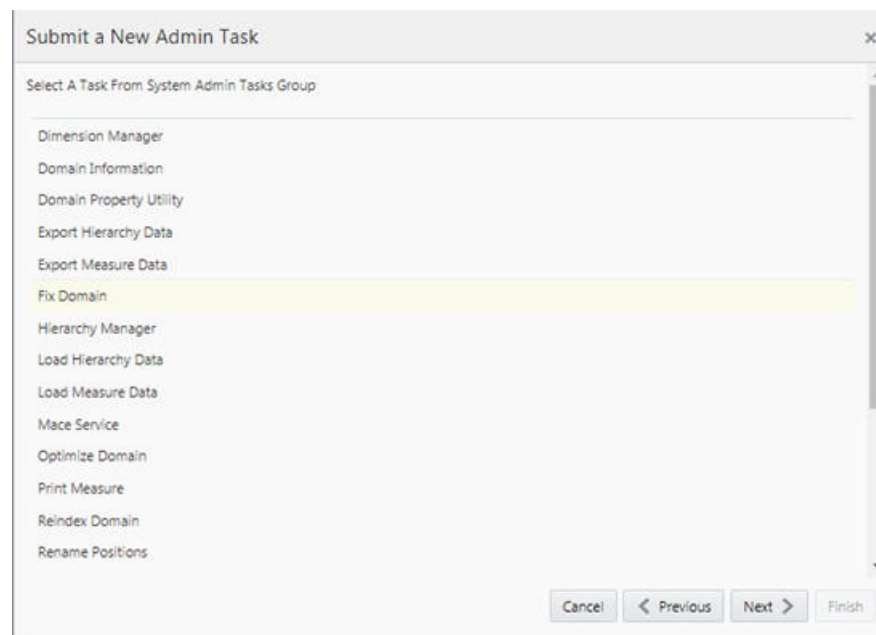
6. Enter a descriptive label in the Task Label text box, select **Clean Measure Attributes**, and click **Next**.

Figure 7–49   Task Arguments

7. You see Schedule Task, where you can choose to run the task ASAP or on a schedule.

Figure 7–50   Schedule Task

8. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

   Click **Previous** to make changes or **Finish** to submit the task.

Figure 7–51 Verify and Confirm

9. To view the status/log file, click the task label.

Figure 7–52 Notification
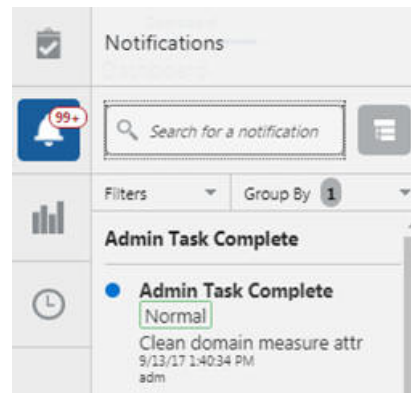
10. To view the Online Administration Dashboard, navigate to Administration task flow and click the **Task Status Dashboard Sub Task**.

Figure 7–53 Status
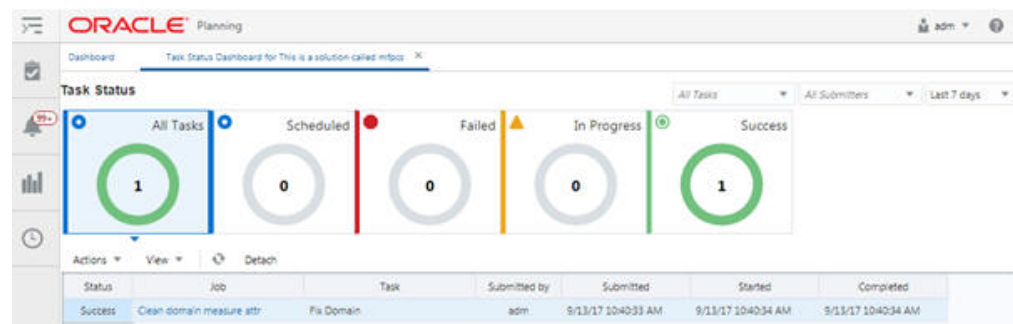
Fix Domain Usage

Table 7–2 lists the arguments that are used with Fix Domain.

Table 7–2   Fix Domain OAT Arguments

Arguments	Description
Clean Measure Attributes	Detects measure attributes that do not exist in the domain.
Clean Measure	Detects partially unregistered measures in the domain.
Repair Hierarchy	Detects inconsistencies in hierarchy information in arrays within the domain.
Fix Arrays	Detects unlinked arrays.

8

Operational Utilities

This chapter details the following operational utilities of RPASCE:

- Setting Miscellaneous Domain Properties

- Setting the RPAS_TODAY Domain Variable

- Accessing the Calculation Engine Using mace

- Managing Segments Using wbbatch

- Managing Workbooks Using wbmgr

- Asynchronous Task Properties

Setting Miscellaneous Domain Properties

This section describes the Domain Property Service task.

Submit Online Admin Task for Domain Property Service

The Domain Property Utility service is used by the administrator to inspect and reset properties of a domain. The supported functions include:

- Reset properties to Default Value

- Display current value of properties

- Display current value of property

- Update property to value provided (name=value)

To submit a task for domain property utility, complete the following steps:

1. From Task Flow, select **Administration** under **Tasks**.

Figure 8–1   Administration
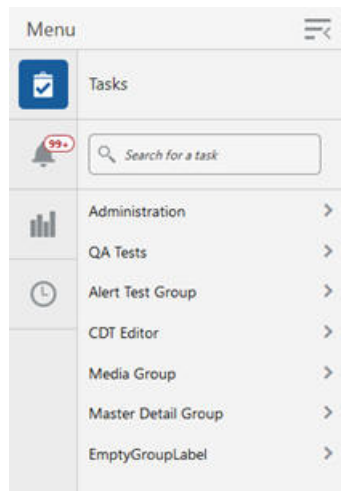
2. Select **Online Admin Tools**.

Figure 8–2   Online Admin Tools
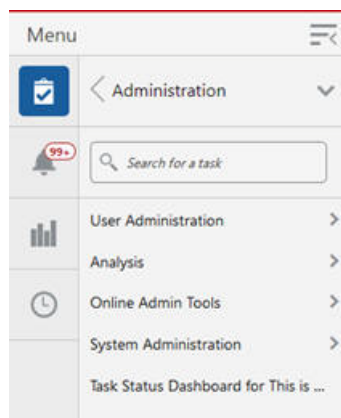
3. To submit a new Administration task, select **Submit Admin Task**.

Figure 8–3   Submit an Admin Task
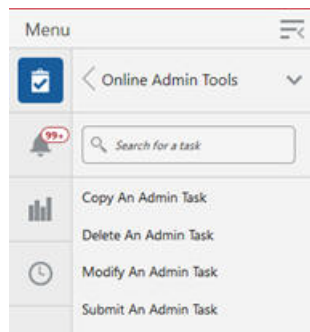
4. Select **System Admin Tasks** and click **Next**.

Figure 8–4   System Admin Tasks
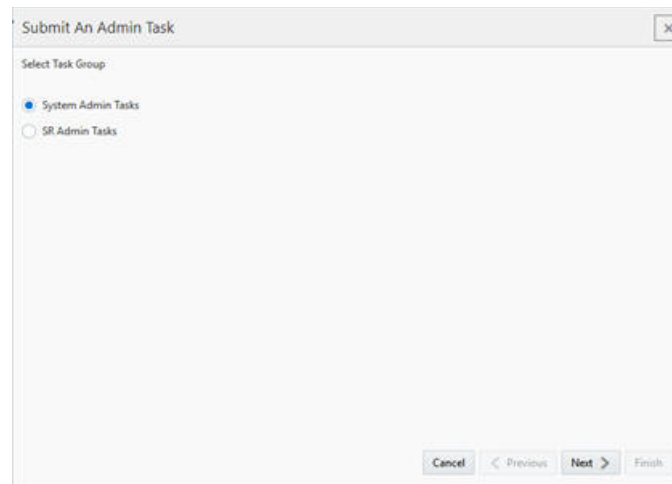
5. From the list of all available system administration tasks, select **Domain Property Utility** and click **Next**.

Figure 8–5   Domain Property Utility
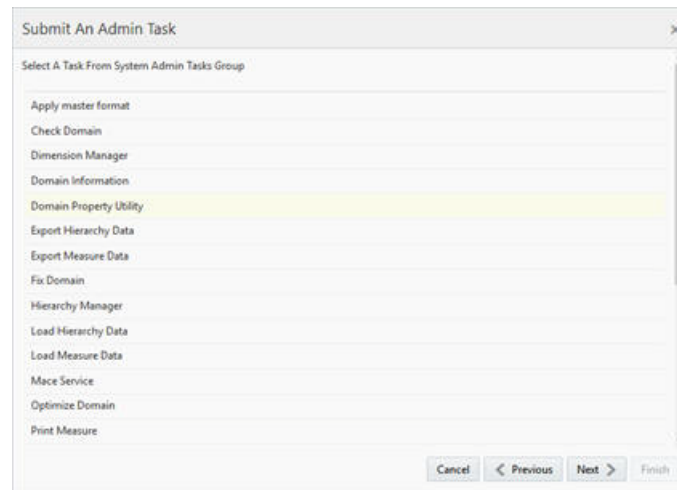
6. Enter a descriptive label in the Task Label text box.

   Select from the following operations:

   - Reset properties to default value

   - Display current value of properties

   - Display current value of property

   - Update property to value provided (name=value)

Figure 8–6   Select Operation

Each available task may also have extra input boxes for the administrator to provide arguments for that task.

Reset Properties to Default Value

This service resets domain properties to the default value. Users can use it to restore domain properties back to the initial state. An administrator can define an arbitrary property in the domain. This service only resets a set of properties defined by RPASCE. It will not reset any properties that the administrator has defined for applications that are not known to RPASCE.

This service also does not modify the domain property, domain_type.

To reset properties to default value, follow the procedure described in Setting Miscellaneous Domain Properties, steps 1 through 5.

1.   When you are asked to specify the argument for the task, select **Reset properties to default value** and click **Next**.

Figure 8–7   Reset Properties

2. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–8   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–9   Verify and Confirm

4. You see the just-scheduled task in the dashboard.

Figure 8–10 Status
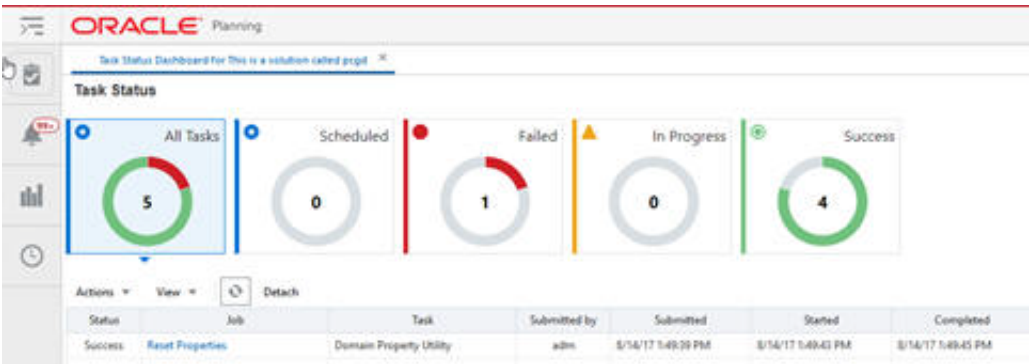
5. Click the task to see the log.

Figure 8–11 Reset Properties Log
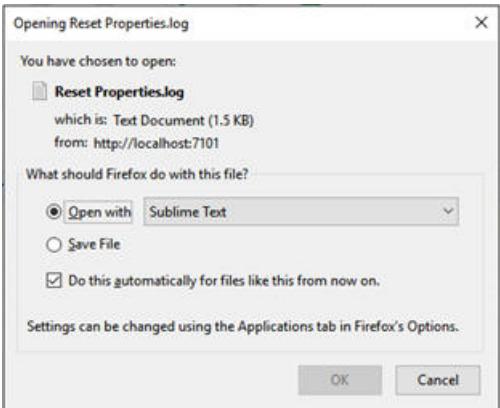
Display Current Value of Properties

The user can use this domain properties service to display the current value of all domain properties currently defined in the domain.

To Display the current value of properties, begin by following the procedure described in Setting Miscellaneous Domain Properties, steps 1 through 5.

1. Enter a descriptive label in the Task Label text box.

   When you are asked to select the operation, select **Display current value of properties** and click **Next**.

Figure 8–12   Current Value of Properties

2. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–13   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–14   Verify and Confirm

4.   You see the just-scheduled task in the dashboard.

Figure 8–15   Status
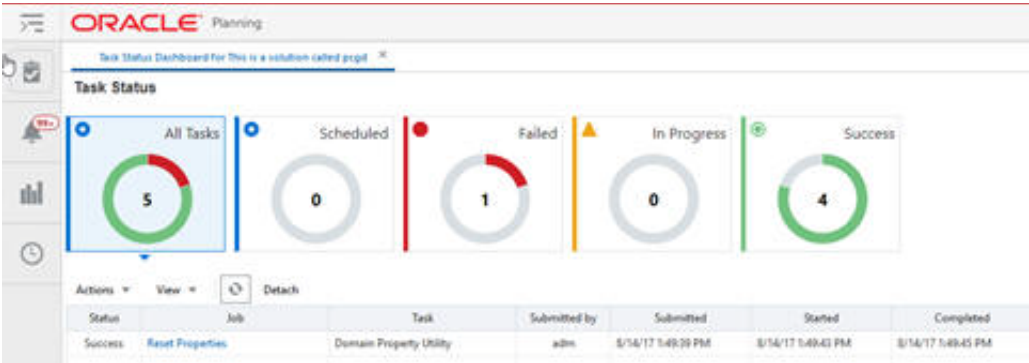
5.   Click the task to see the log.

Figure 8–16   Current Properties

Display Current Value of Property

The user can use this domain properties service to display the current value of a particular domain property. User must specify the valid property name for this service.

To display the current value of a specific property, begin by following the procedure described in Setting Miscellaneous Domain Properties, steps 1 through 5.

1.   Enter a descriptive label in the Task Label text box.

When you are asked to specify the operation, select **Display current value of property** and type in a valid property name in the input box right beneath it.

Figure 8–17   Display Current Value of Property

2. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–18   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–19   Verify and Confirm
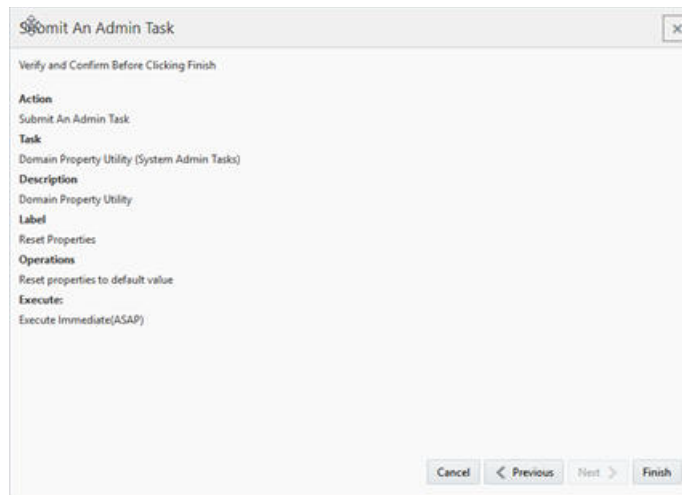
4. You see the just-scheduled task in the dashboard.

Figure 8–20   Status
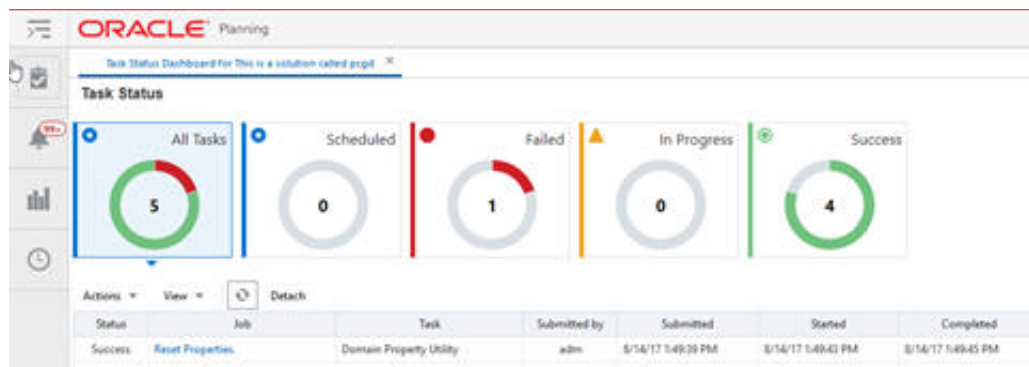
5. Click the task to see the log.

Figure 8–21   Domain Type

Update Property to Value Provided (name=value)

The user can use this domain properties service to update the value of a particular domain property. Users must specify the valid property name and value in the form name=value.

To update a property to the value provided, begin by following the procedure described in Setting Miscellaneous Domain Properties, steps 1 through 5.

1. Enter a descriptive label in the Task Label text box.

When you are asked to specify the operation, select **Update property to value provided** and type in the name=value expression in the input box right beneath it.

Figure 8–22   Update Property to Value

2. Click **Next**. Specify the value type of the property. For the property navalue_opt, select boolean or numeric.

Figure 8–23   Property Value Type

3. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–24   Schedule Task

4. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–25   Verify and Confirm

5. You see the just-scheduled task in the dashboard.

Figure 8–26   Status
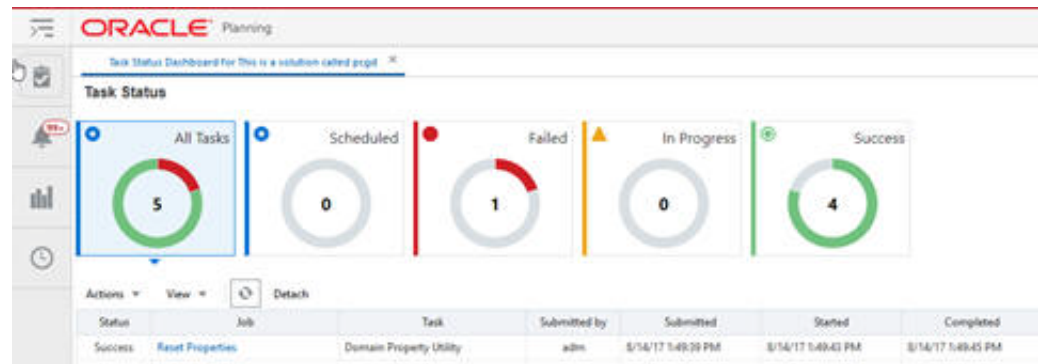
6. Click the task to see the log.

Available Domain Properties

Table 8–1   Available Domain Properties

Property Name	Type	Description
fixedwidth_utf8	Boolean	If this property is set to true, loadhier, loadmeasure, and exporthier will use character count for the start position and the width of the field. If it is false or unset, fixed-width uses byte count instead of character count. By default, this property is set to false.
ovr_def_admin_privileges	Boolean	Using the Security Administration workbook, administrators can set workbook template access for every user in the system. Non-administrative users cannot access the workbook templates to which they have not explicitly been given access. However, if a user is an administrator, by default, they can see all the workbooks in the system. Some retailers want to prevent this from happening. Reasons for this include reducing clutter and having different kind of administrators manage different administrative tasks in their RPASCE systems. Ability to control template access for administrators from the Security Administration workbook is made possible by setting this domain property to true. By default, this property is false.
skipped_records_log_limit	Integer	The limit on the number of lines in the skipped/invalid records log file. The default value is 1000.

Setting the RPAS_TODAY Domain Variable

Customers previously set the RPAS_TODAY environment variable as a global setting. This setting is then applied to all their tasks and operations scheduled through OAT. In order to accomplish this, customers would contact a cloud administrator (or raise an SR), who would set the RPAS_TODAY environment variable and restart all services.

Now, customers themselves can set the RPAS_TODAY variable using the OAT task (as a domain property and not as an environment variable) RPAS_TODAY Domain Variable. They do not need to call a cloud administrator or raise an SR. Once the customer has set it, the property is applied to all processes.

Note that the following priority order list is adhered to by the RPAS_TODAY setting:

1. If RPAS_TODAY is set during the batch process, the OAT task wizards take the highest precedence. See"RPAS_TODAY for Online Administration Tools Tasks"

2. Then, RPAS_TODAY is set using the OAT task, RPAS_TODAY Domain Variable.

3. Then, the RPAS_TODAY environment variable is set in the backend before the services are started.

You can access the RPAS_TODAY Domain Variable task by selecting System Admin Tasks -> RPAS_TODAY Domain Variable. You see the Submit a New Admin Task dialog box, shown in Figure 8–27.

Figure 8–27   Setting the RPAS_TODAY Variable

The following options are available:

- View RPAS_TODAY setting. Use this option to see the current value for the RPAS_ TODAY domain property set in the domain.

- Set RPAS_TODAY setting. Use this option to set the RPAS_TODAY domain property.

- Unset RPAS_TODAY. Use this option to clear the RPAS_TODAY domain property.

Note that this task will be run in offline mode. As a result, when the task daemon is able to run, all non-administrative users will be logged off and the task will be executed. This prevents users from running any batch or workbook operations while the RPAS_TODAY setting is updated in the domain.

Accessing the Calculation Engine Using mace

The mace service (Multi-Dimensional Array Calculation Engine) is for informational use only. The administrator can use the service to inspect rule and rule group

information currently configured in the domain. It does not allow the administrator to modify any existing rule or rule groups or run any expressions or rule groups in the domain to modify domain measure data. The supported functions include:

- Search for a specific string

- Validate a specific expression

- Print the specified rule

- Print the specified rule group

- Print all rule and rule groups

Submit Online Admin Task for Mace Service

To submit a mace service task, complete the following steps:

1. From Task Flow, select **Administration** under **Tasks**.

Figure 8–28   Administration
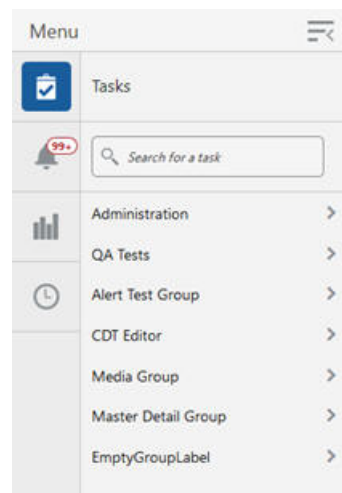
2. Select **Online Admin Tools**.

Figure 8–29   Online Admin Tools
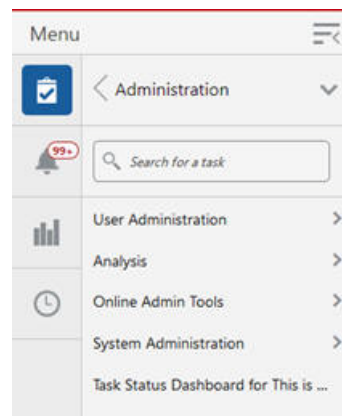
3. To submit a new Administration task, select **Submit Admin Task**.

Figure 8–30   Submit an Admin Task
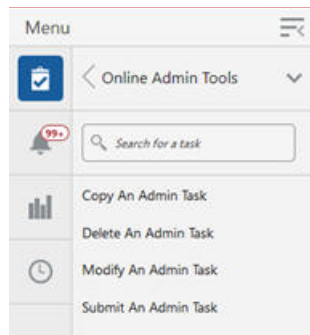
4. Select **System Admin Tasks** and click **Next**.

Figure 8–31   System Admin Tasks
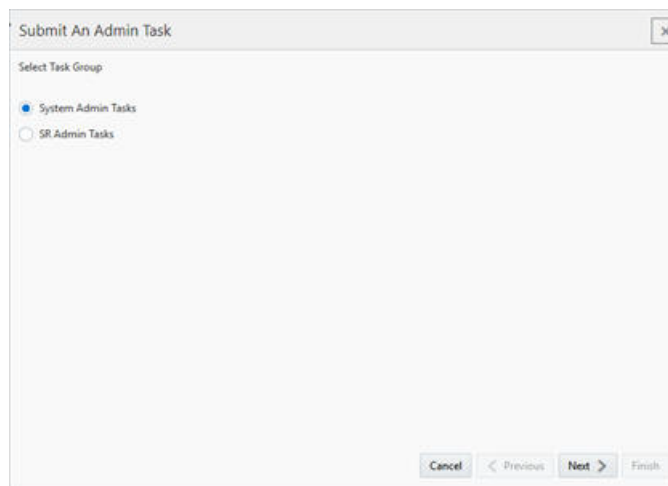
5. From the list of all available system administration tasks, select **Mace Service** and click **Next**.

Figure 8–32   Mace Service
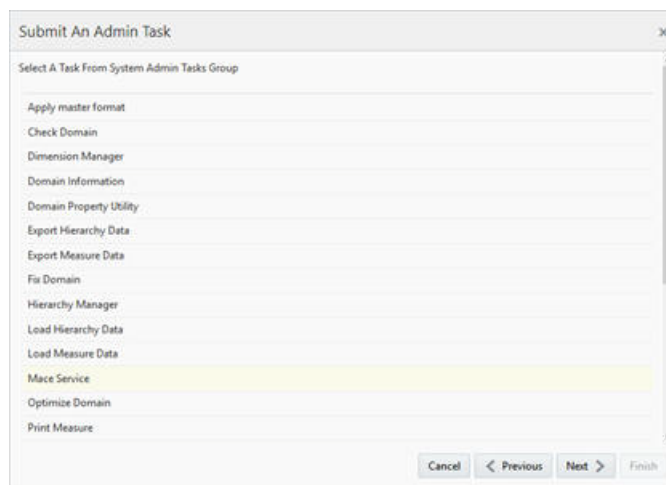
6. Enter a descriptive label in the Task Label text box.

Select an operation from the following list of operations.

- Search for a specific string

- Validate a specific expressions

- Print the specified rule

- Print the specified rule group

- Print all the rules and rule groups

Figure 8–33   Mace Operations

Search for a Specific String

This service finds all occurrences of a user-specified string in all the expressions configured in the domain. The string search is case insensitive. If a user searches for "ttlsales", expressions such as "TtlSales=..." and "TTLSALES = ..." will be identified. A user can search for any meaningful text in the expressions, not just measure names. For example, a user can search for function names to show all expressions configured to evaluate that function. Or a user can search for a modifier such as "level([clnd].[week])" to show all expressions that used that modifier.

The most common use for this function is to discover how a certain measure is calculated or referenced by rules configured in the domain. In this case, the administrator can search for the measure name as a string. Mace will return all rules that reference the measure.

1. Select **Search for a specific string**, enter the search term (here, 'TotalSales') in the Search for a specific string text box and click **Next**.

Figure 8–34   Specific String

2. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.
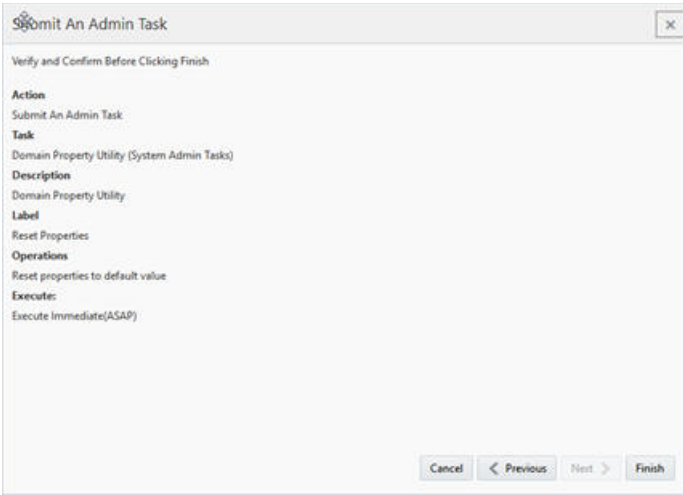
Figure 8–35   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–36   Verify and Confirm
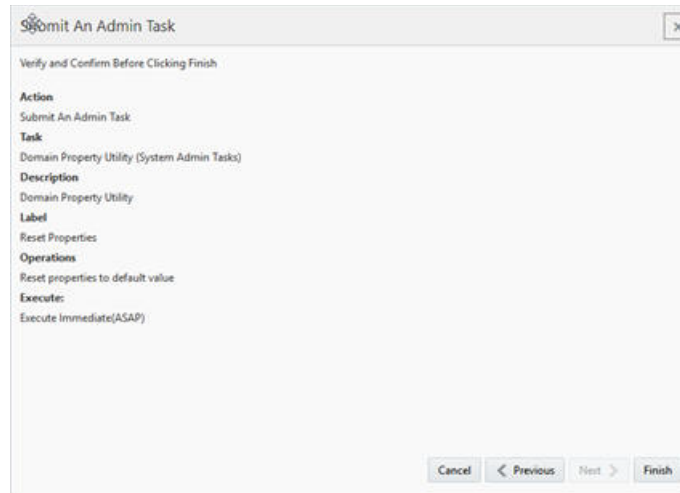
4. You see the just-scheduled task in the dashboard.

Figure 8–37   Status
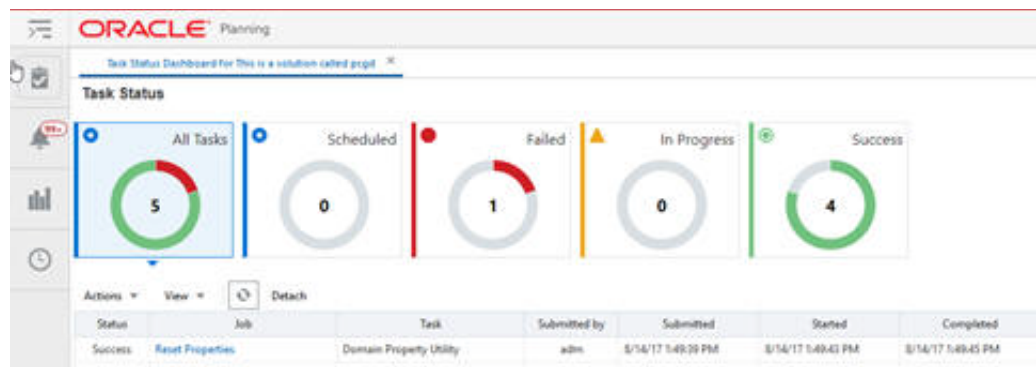
5. Click the task to see the log.

Figure 8–38   Measure Search
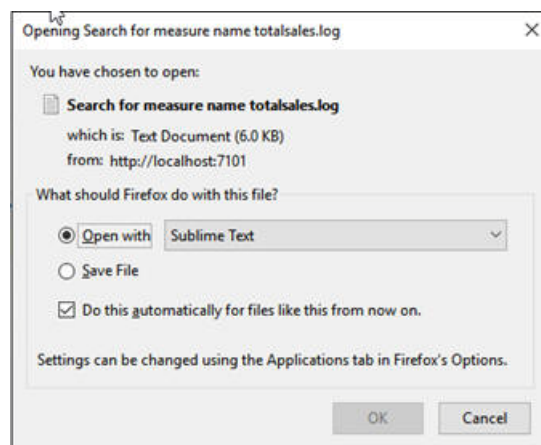
6. The log file is named after the task label. Search for measure name totalsales.log.

Figure 8–39   Search Log
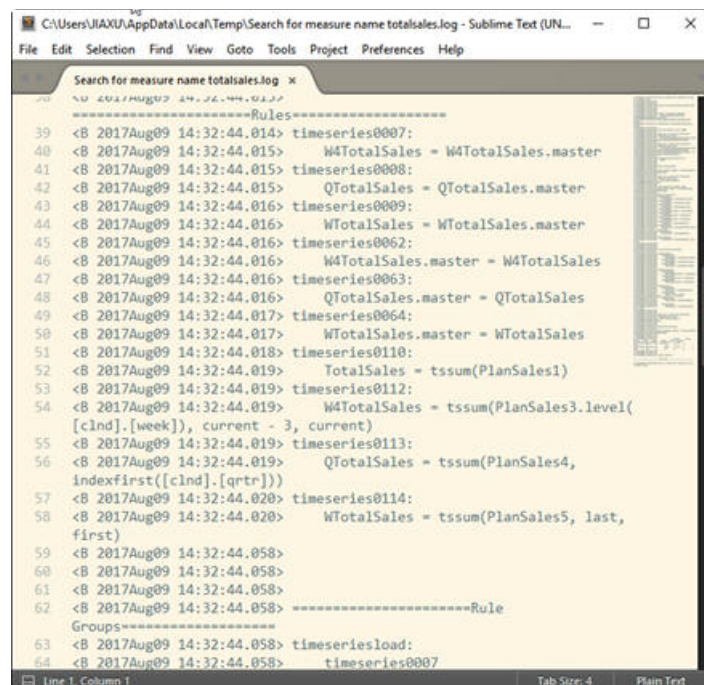
The output lists expressions involving the measures WtotalSales, QtotalSales, and W4TotalSales, which all include the string "TotalSales".

Validate a Specific Expression

This service determines whether or not an RPASCE expression is written in the correct syntax. If it is not, some helpful diagnostic information is provided.

1. Select **Validate a specific expression**, enter the expression to be validated, and click **Next**.

Figure 8–40   Validate a Specific Expression

2. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

*Figure 8–41 Schedule Task*

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

   Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–42 Verify and Confirm

4. You see the just-scheduled task in the dashboard.

Figure 8–43   Status
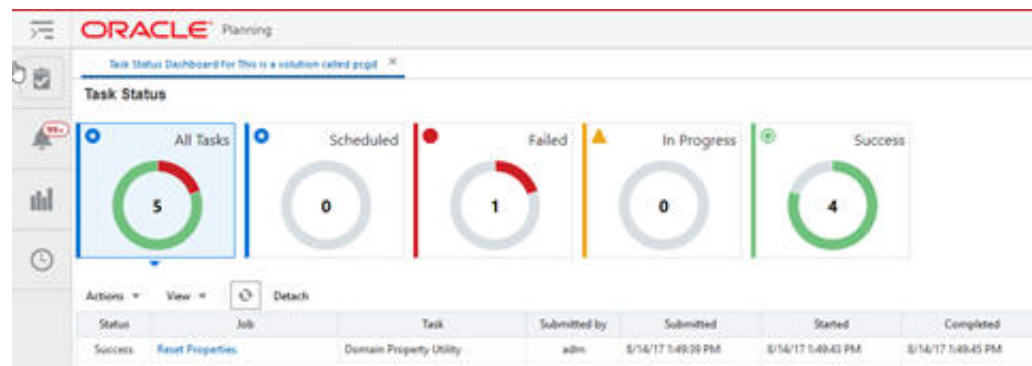
5. Click the task to see the log.

Print the Specified Rule

This service prints out all the configured expressions in the user-specified rule. The administrator can use this service to inspect the content of the rule.

1. Select **Print the specified rule**, enter the rule name to be printed, and click **Next**.

Figure 8–44   Print Operation

2. You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–45   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

   Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–46   Verify and Confirm

4. You see the just scheduled task in the dashboard.

Figure 8–47   Status
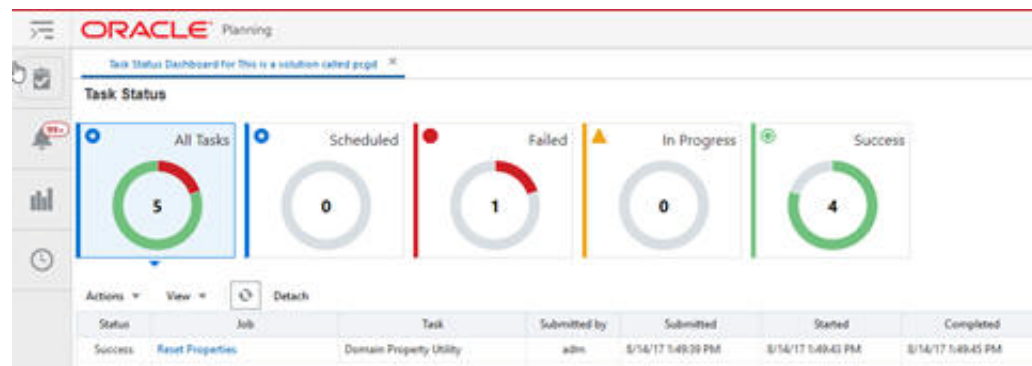
5.   Click the task to see the log.

Print the Specified Rule Group

This mace service prints out all rules and the expressions configured for a particular rule group. The administrator can use this service to inspect the content of a complete rule group.

1.   Select **Print the specified rule group**, enter the rule group name to be printed, and click **Next**.

Figure 8–48   Print Rule Group Operation

2.   You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–49   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

   Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–50   Verify and Confirm

4. You see the just-scheduled task in the dashboard.

Figure 8–51   Status
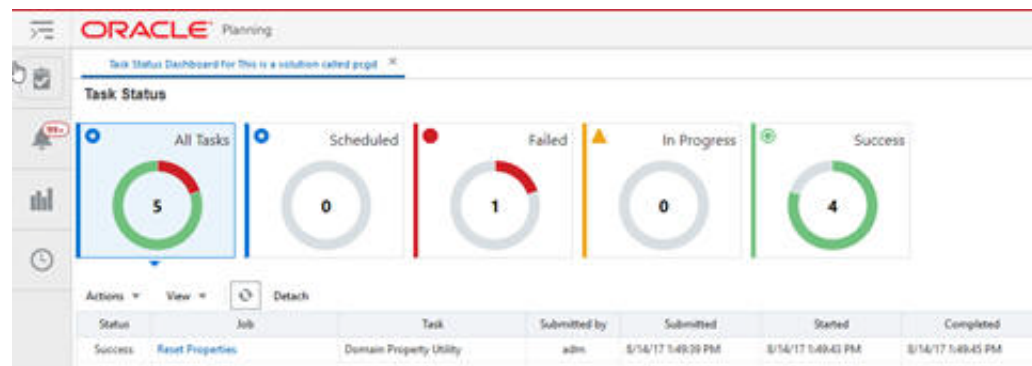
5.   Click the task to see the log.

Print All the Rules and Rule Groups

This mace service prints out all the expressions configured for all rule groups in the domain. The administrator can use this service to inspect the content of all rule groups.

1.   Select **Print all the rules and rule groups** and click **Next**.

Figure 8–52   All Rule Groups Arguments

2.   You see Schedule Task, where you can run the task ASAP or on a schedule. If you choose to run on a schedule, you can select a frequency or a specific start and stop date and time.

Figure 8–53   Schedule Task

3. Click **Next** to navigate to the next page, where you verify and confirm the task information entered.

    Click **Previous** to make changes or **Finish** to submit the task.

Figure 8–54   Verify and Confirm

4. You see the just-scheduled task in the dashboard.

Figure 8–55   Status
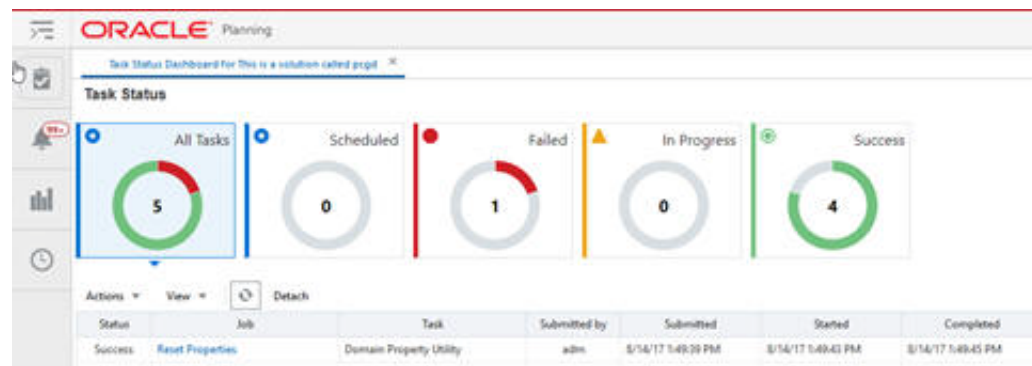
5. Click the task to see the log.

Managing Segments Using wbbatch

The Segment Manager utility, WbBatch, is used by the administrator to manage the segment and build, commit, refresh, or remove a segment as well as the batch operation on the workbooks in the build, commit, and refresh queues.

Figure 8–56   WbBatch

WbBatch provides the following functionality:

-   Remove a segment

-   Remove workbook for a segment

-   Build a segment

-   Commit a segment

-   Refresh a segment

-   Add segment to batch build queue

-   Add segment to batch commit queue

-   Add segment to batch refresh queue

- Remove segment from batch build queue

- Remove segment from batch commit queue

- Remove segment from batch refresh queue

- Start batch operation on all/build/commit/refresh queues

- Print segment in all/build/commit/refresh queues

Remove a Segment

This option is used to select the segment from the drop-down list and remove the segment from the domain. It also deletes all the workbooks that are built based on this segment.

Figure 8–57   Remove a Segment

Figure 8–58   Remove Segments Arguments

Remove Workbook for a Segment

This option is used to select the segment and delete its workbook, built based on this selected segment, while this segment remains.

Figure 8–59   Remove Segment Workbook

Build a Segment

This option is used to select a segment from the segment drop-down list and build a workbook based on it. The segments that already have their associated workbook are filtered from this drop-down list.

Figure 8–60   Build Segment Arguments

Commit a Segment

This option is used to select a segment and commit its change in the workbook to the domain. The drop-down list only retains those segments that have their associated workbooks in the domain. If this segment is in the commit queue, it will be taken out once this commit operation completes.

Figure 8–61   Commit Segment Arguments

Refresh a Segment

This option is used to select a segment and refresh its workbook data from the domain. The drop-down list only retains those segments that have their associated workbooks in the domain.

Figure 8–62   Refresh Segment Arguments

Add Segment to Batch Build Queue

This option is used to add the selected segments to the build queue. The segment list includes all the segments in the domain, and it supports multiple segment selection operation.

Figure 8–63   Add Segment

Add Segment to Batch Commit Queue

This option is used to add the selected segments to the commit queue. The segment list includes all the segments that have their associated workbook in the domain, and it supports multiple segment selection operation.

Figure 8–64   Segment Batch Commit Queue

Add Segment to Batch Refresh Queue

This option is used to add the selected segments to the refresh queue. The segment list includes all the segments that have their associated workbook in the domain, and it supports multiple segment selection operation.

Figure 8–65   Segment Batch Refresh Queue

Remove Segment from Batch Build Queue

This option is used to remove the selected segments out of the build queue. The segment list includes all the segments in the domain, and it supports multiple segment selection operation. If the selected segment is not in the queue, it will provide a warning message in the output log file.

Figure 8–66   Remove Segment Batch Build Queue

Remove Segment from Batch Commit Queue

This option is used to remove the selected segments from the commit queue. The segment list includes all the segments that have their associated workbook in the domain, and it supports multiple segment selection operation. If the selected segment is not in the queue, it will provide a warning message in the output log file.

Figure 8–67   Remove Segment Batch Commit Queue

Remove Segment from Batch Refresh Queue

This option is used to remove the selected segments from the refresh queue. The segment list includes all the segments that have their associated workbook in the domain, and it supports multiple segment selection operation. If the selected segment is not in the queue, it will provide a warning message in the output log file.

Figure 8–68   Remove Segment Batch Refresh Queue

Start Batch Operation on All/Build/Commit/Refresh Queues

This option is used to batch process all the segments in the all/build/commit/refresh queues.

Figure 8–69   Start Batch Operation
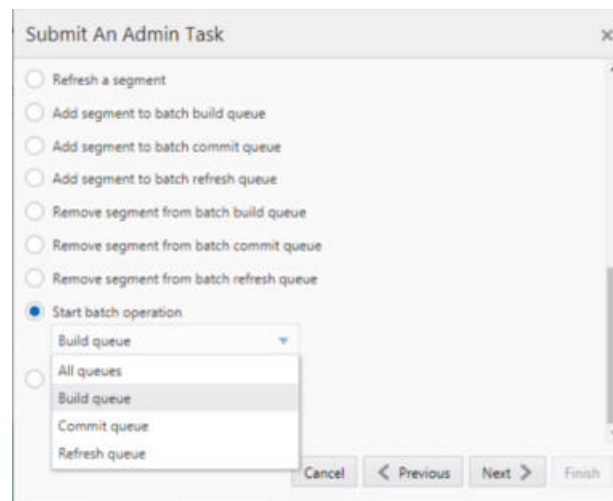
Print Segment in All/Build/Commit/Refresh Queues

This option is used to print out the detailed information of the segments in the all/build/commit/refresh queues.

Figure 8–70   Print Segment
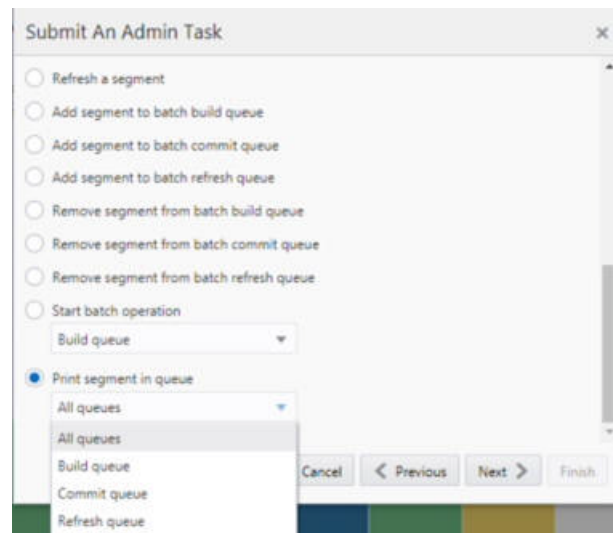
FAQ

Question: How to build/commit/refresh multiple segments at the same time.

Answer: Add those segments into the build/commit/refresh queue, then start a batch operation on that queue.

Managing Workbooks Using wbmgr

The Workbook Manager Utility, WbMgr, is the administration task that can be used to inspect or remove existing workbooks. It supports workbook domain transparency, which means a user can use it to manage all the workbooks throughout the domains (including the workbooks in the master domain and the ones in the local domains).

Figure 8–71   Workbook Manager

WbMgr provides the following functionality:

- List all the workbooks in the domain
- Show segment label (optional)
- List workbooks belonging to a specific user
- Show segment label (optional)
- List workbooks belonging to a specific template
- Filtered by a specific user
- Show segment label (optional)
- Remove all workbooks in the domain
- Remove workbooks belonging to a specific user
- Support multiple workbook selection
- Remove workbooks belonging to a specific template
- Filtered by a specific user
- Print detailed information about workbooks
- Support multiple workbook selection

List All Workbooks in the Domain
This option is used to list all the workbooks in the domain. A wizard page
(Figure 8–73) provides an optional argument to display the segment label in the output
file.

Figure 8–72   List All Workbooks in the Domain

Figure 8–73   List All Workbooks in the Domain Arguments

List Workbooks Belonging to a Specific User
This option is used to list all the workbooks owned by a specific user (from the user
drop-down list) in the domain. A wizard page (Figure 8–75) provides an optional
argument to display the segment label in the output file.

Figure 8–74   Workbooks Belonging to a Specific User

Figure 8–75   Workbook Belonging to a Specific User Arguments

List Workbooks Belonging to a Specific Template

This option is used to list all the workbooks built based on the workbook template (from workbook template drop-down list) in the domain. A wizard page (Figure 8–77) provides two optional arguments, a segment label to display the segment label in the output file and user filtering by the workbook owner (from the user drop-down list).

Figure 8–76   Workbook Belonging to a Specific Template

Figure 8–77   Workbook Belonging to a Specific Template Arguments

Remove All Workbooks in the Domain

This option is used to delete all the workbooks in the domain.

Figure 8–78   Remove All Workbooks in the Domain

Remove Workbooks Belonging to a Specific User

This option is used to delete all the workbooks owned by a specific user (from the user drop-down list) in the domain. A wizard page (Figure 8–80) can be used to select multiple workbooks to be deleted.

Figure 8–79   Remove Workbooks Belonging to a Specific User

Figure 8–80   Remove Workbooks Belonging to a Specific User Arguments

Remove Workbooks Belonging to a Specific Template

This option is used to delete all the workbooks built based on the workbook template (from the workbook template drop-down list) in the domain. A wizard page (Figure 8–82) provides the option to filter the workbook by the workbook owner (from the user drop-down list).

Figure 8–81   Remove Workbooks Belonging to a Specific Template

Figure 8–82   Remove Workbooks Belonging to a Specific Template Arguments

Print Detailed Information About Workbooks

This option is used to print out the detailed information about the workbook selected by the user. It supports the selection of multiple workbooks. The detailed information about workbook includes Label, Last opened by, User privilege, Group privilege, World privilege, User ID, User Name, Group, Created, Modified, and Committed.

Figure 8–83   Print Detailed Information About Workbooks

Asynchronous Task Properties

This section describes the Asynchronous Task properties introduced in
rpasConfig.properties on RPASCE Client under <install_base>/config/rpas/ folder.
Any updates made to this file require a restart of the Weblogic server.

Asynchronous Task Execution Property

In a cloud environment, if task execution (for example, workbook create) takes too
much time, then a TCP socket timeout can occur. If this property is true, the task
executes asynchronously so that the request is alive. If this property is set to false, the
execution is synchronous. The default value is false.

asynchronous.task.execution=false

Asynchronous Task Execution Poll Interval Property

If a task executes asynchronously, the polling interval time can be configured. The
default value is 1000 milliseconds.

asynchronous.task.execution.pollinterval=1000

Asynchronous Task Timeout Limit Property

This property sets a timeout limit for the submitted asynchronous task. If the
submitted asynchronous task does not finish within this limit, then that asynchronous
task will be cancelled. The default value is 3600000 milliseconds (that is, one hour).

asynchronous.task.submit.timeout.limit=3600000

Asynchronous Task Initial Wait Property

This property sets a initial wait time before starting the asynchronous poll, which improves the turnaround time for short running transactions. The default value is 10000 milliseconds (that is, 10 seconds).

asynchronous.task.initial.wait=10000

9

Patch Domain

Administrators can use the Patch Domain task to patch the domain.

In order to upload the domain configuration into the cloud environment, create an archive (either .zip or .tar.gz) containing the config directory and all of its contents. This archive file must be named as _config.zip or _ config.tar.gz. This archive file must be placed in the config subdirectory of the SFTP incoming location on the server. It may be updated as often as necessary in support of domain build or patch activities. Remember to create the COMMAND/COMPLETE trigger when you are finished placing files in this directory.

Run the OAT task with a Label and an offline message that will be displayed to the users while the patch domain is in progress.

10

Batch Administration

This chapter addresses the following batch administration tasks:

- Run Batch Measure Load Group
- Run Batch Measure Export Group
- Run Batch Calc Group
- Run Batch Task Group
- Clean Up Task
- Retrieve Batch Control Files
- Upload Batch Control Files
- Upload Custom Translations
- Remove Custom Translations
- Other Translation Tasks

Application-Specific Batch Tasks

The application-specific batch tasks are a set of tasks that help the Administrator to load/export a predefined set of data and hierarchies. These tasks are found in the Online Administration Tools as Configured Batch Tasks.

Each task in turn calls predefined batch services to do the required tasks. For example, MFP Cloud Service uses the Enterprise Edition batch framework to define the batch process.

Some tasks require input data files or hierarchy files as input. These files must be uploaded before the tasks are scheduled. Some tasks export files out of the application. Those files are copied to the SFTP export location, and Administrators can download these exported files.

The SFTP file location is a space provided to the Administrator to load import files and access exported batch files. For more information, see "Uploading and Downloading Batch Files."

The following table shows the list of application-specific administration tasks that are pre-configured for MFP Cloud Service:

Configured Batch Tasks
Run Batch Measure Load Group
Run Batch Measure Export Group

Configured Batch Tasks
Run Batch Calc Group
Run Batch Task Group
Clean up Task
Retrieve Batch Control File

> **Note:** These are OAT tasks only and will work only if the batch control files are loaded to the domain. Refer to the *Oracle Retail Merchandise Financial Planning Cloud Service Implementation Guide* and *Oracle Retail Merchandise Financial Planning Cloud Service Administration Guide* for sample control files.

Run Batch Measure Load Group

This task is used to load a predefined set of measures grouped as a measure set in the batch control file batch_loadmeas_list.txt. These set of measures can be loaded together. All required data files must be uploaded to the SFTP location before scheduling this task. For more information, see "Uploading and Downloading Batch Files." If there is no file present to load a particular measure in that measure set, it will be logged as not found and skipped.

This process will only load the data into those measures and will not run any subsequent calculations. The Administrator may need to schedule the necessary Run Batch Calc Group task after scheduling this task in order to process the loaded data.

For details about the list of measure files that are loaded for different measure sets and information on the data file formats, see the *Oracle Retail Merchandise Financial Planning Cloud Service Implementation Guide*.

With Enterprise Edition implementations, the Run Batch Measure Load Group task can be customized to load different sets of measures by making changes to the batch control files. For information about changing the batch control files and how to upload them, see the *Oracle Retail Merchandise Financial Planning Enterprise Edition Cloud Service Implementation Guide*.

Run Batch Measure Export Group

This task exports a predefined set of measure data into flat files. The predefined set of measures to be exported are defined in the batch control file batch_exportmeas_list.txt. This set of exported data can be used for integration with other applications. Typically, the data is exported to a database that creates a report or allows a query by users. The Administrator needs to schedule the Set Export Flags under the Run Batch Calc Group task to set the required export flags before scheduling this task. Exported files for this task are put into the SFTP Export location; the Administrator can download the files from this location. For more information, see "Uploading and Downloading Batch Files."

With Enterprise Edition implementations, the Run Batch Measure Export Group task can be customized to export new versions of data or remove a particular version of the list of measures exported by changes to the batch control files. For information about changing the batch control files and how to upload them, see the *Oracle Retail Merchandise Financial Planning Enterprise Edition Cloud Service Implementation Guide*.

Run Batch Calc Group

This task is used to run a predefined set of all batch rules (as batch calc sets) against the domain to do the necessary data aggregation that is defined in the batch control file batch_calc_list.txt.

With Enterprise Edition implementations, the Run Batch Calc group task can be customized to change the batch calculations and the order in which they run against the domain by changing the batch control files. For information about changing the batch control files and how to upload them, see the *Oracle Retail Merchandise Financial Planning Enterprise Edition Cloud Service Implementation Guide*.

Run Batch Task Group

This task is used to schedule a predefined set of batch tasks such as load measure, batch calc, and batch exports in a sequential order. The batch control file batch_exec_list.txt defines the steps in the specific order that will be executed during batch task execution. This task is used to schedule a typical daily or weekly MFP Cloud Service batch. If any task aborts, a log is created and the process terminates. This task has the option to restart from the last failed service rather than running all the completed services again.

> **Note:** Typically, the user must schedule this task within a time frame after a domain backup and when no users are logged into the domain. The Administrator must work with Oracle to find out the time of backups and any other jobs.

With Enterprise Edition implementations, the Run Batch Task Group task can be customized by making changes to the batch_exec_list.txt control file. For more details about the list of internal services this task can call and how to change the order of services to this task, see the *Oracle Retail Merchandise Financial Planning Enterprise Edition Cloud Service Implementation Guide*.

Clean Up Task

This task is provided to clean up the processed files or unprocessed files from different cloud service locations to be run by the retailer on an on-need basis. This task has the following three options to select the locations to clean up when this task is scheduled:

- **Clean FTP Holding Area** - When incoming files are sent to the FTP site, a process then moves those files to an internal holding area from which the batch framework can process them. In cases where files are sent using FTP and then not used by any batch process (for example, if the batch configuration has changed or if a file was misnamed), it becomes stranded in the internal holding area. This option is provided to remove all files from the internal holding area. Note that all files currently in the internal holding area will be removed, so this option must not be used when any scheduled batch processes are executing.

- **Clean FTP Export Area** - Select this option to remove all files from the outgoing FTP holding area.

- **Clean processed folders** - All processed input files are archived within the domain/input/processed folders. This option can be used to remove all the processed directories within a domain to reduce the growing domain size.

Retrieve Batch Control Files

This task allows the currently configured batch control files to be retrieved, in case they must be inspected or modified. The set of files are packaged together as the archive file batch_control.tar.gz, and placed into the FTP area for retrieval. No parameters are required for this OAT task.

Upload Batch Control Files

This task allows the configured batch control files to be uploaded in case they must be modified. The batch control files must be uploaded to the incoming FTP area under batch_control directory. No parameters are required for this OAT task.

Translation Tasks

The RPASCE platform and its applications update the default translations in the domain as part of the regular upgrade and patch process. If the customer has existing custom translations for any of the default RPASCE strings, these translations may be wiped out after a patch or upgrade. To ensure that the custom translations are maintained after patches and upgrades, use the Translations task. This task provides an upload option that manages the custom translations for both the server and the UI. This task includes other subtasks related to translation.

Upload Custom Translations

Use this task to upload custom translations. The translations archive file name does not have any special restrictions but must have a .zip or .tar.gz extension. The translations files in it must be either without a top-level folder or with one folder containing all the files. If there is a top-level folder, the name of the archive must match the folder name. The task looks for input translation archives in the new location, $INCOMING_FTP_PATH/translations. The second page of the upload task provides a drop-down list of all available translation archives. The user can select one and continue. The translation dat files are loaded into the domain using loadHier; the translation measures files are loaded using loadmeasure. The UI translation files are copied to <install_base>/config/client folder. In addition, a copy of all the uploaded translation files are maintained internally. This helps to list and download just the custom translations.

Here is an example for the translation archive file.

```
$ tar -tzf tr.tar.gz
tr/
tr/MultiSolutionBundle_de.properties
tr/MultiSolutionBundle.properties
tr/MultiSolutionBundle_es.properties
tr/r_langattr.csv.ovr
tr/MultiSolutionBundle_it.properties
tr/r_measdescripti.csv.ovr
tr/MultiSolutionBundle_en_US.properties
tr/MultiSolutionBundle_pt.properties
tr/r_measlabel.csv.ovr
tr/MultiSolutionBundle_ja.properties
tr/MultiSolutionBundle_en_AU.properties
tr/MultiSolutionBundle_fr.properties
tr/r_dimlabel.csv.ovr
tr/r_grplabel.csv.ovr
tr/r_msglabel.csv.ovr
tr/msgs.csv.dat
```

```
tr/lngs.csv.dat
tr/MultiSolutionBundle_en_GB.properties
tr/r_hierlabel.csv.ovr
tr/MultiSolutionBundle_en_CA.properties
tr/r_wbtglabel.csv.ovr
tr/r_wbtlabel.csv.ovr
tr/MultiSolutionBundle_nl.properties
tr/r_measpicklist.csv.ovr
```

Note that, for the server side translation files with dat and ovr extension, you must ensure that only the changed records are present in the translation files. For example, if you change only the dimension labels listed below but the remainder of the dimension labels are default ones, then you must keep only the ones you have changed in the r_dimlabel.csv.ovr measure data file. This ensures that changes to the other dimension labels during patch will not be reverted to the old strings.

```
hier,english,Custom Hierarchy
meas,english,Custom Measure
rgrp,english,Custom Rule Group
lang,english,Custom Language
wb,english,Custom Workbook
```

If you do not edit the translation input files directly, but instead use the translation admin workbook to add the custom translations, then you must do the following. Use exportHier and exportmeasure to export the translation hiers and measures; diff against the default translation files present in $RPAS_HOME/domain/ml_input to find only the custom translations.

Similarly, for client side translation files such as MultiSolutionBundle_nl.properties, keep only the changed lines in the file. They will be merged with the default translations.

Remove Custom Translations

This task has the options shown in Figure 10–1.

Figure 10–1   Remove Custom Translations

Use the options as follows:

■   Remove server translations. Remove the custom server translations and load the default server translations back into the domain using loadHier and loadmeasure.

■   Remove UI translations. Remove the custom UI translations from the location <install_base>/config/client. The default UI translation files are present in the default location and will be in effect once the UI is restarted.

- Delete uploaded translations. If this is not checked, the custom server and UI translation files continue to exist in the internal storage location and are not deleted. They can be re-applied using the Apply Custom Translation task from the other translations tasks drop-down list. If this option is checked, custom translations must be uploaded again.

Other Translation Tasks

This task consists of the following subtasks.

Apply Custom Translation

If translation files have previously been uploaded using the Upload Custom Translations task, it must now be applied to the domain using this task. However, the Upload task also applies the translations, and the patch scripts automatically call the Apply task during the domain patch process. Therefore, in a regular upgrade and patch scenario, the manual execution of this task is not required. However, this option is exposed through OAT to be used if required. For example, if the Remove Custom Translations task is executed without the Delete Uploaded Translation option, the Apply task can be run in order to re-apply the previously uploaded custom translations.

Download All Translations

This task exports all the translations that are currently in use by the server and UI. This includes both custom translations and the remaining default translations. The resulting archive file is placed in the OUTGOING_FTP_PATH location. The name of the file is in the format Translations_<timestamp>.tar.gz.

Download Custom Translations

This task exports only the custom translations that are currently in use by the server and UI. The resulting archive file is placed in the OUTGOING_FTP_PATH location. The name of the file is in the format CustomTranslations_<timestamp>.tar.gz.

List Custom Translations

This task lists the custom translations that are currently in use by the server and UI. To view the output of this task, examine the task log in the OAT dashboard.

11

Uploading and Downloading Batch Files

This chapter describes the file upload/download process. For the batches that load data, the Administrator must upload the relevant input files for the data to the SFTP server before running the batch. After running any export task, the exported files can be found in the export location of the SFTP server.

For details regarding file contents and formatting of the upload and download files, see the *Oracle Retail Merchandise Financial Planning Cloud Service Implementation Guide*.

Adding Authorized Keys

This section describes the process to generate a 2048 bit RSA key and add the same to the SFTP server. This is done with the help of the WinSCP tool on Windows. However, the same can be done using ssh-keygen on Linux as well.

Prerequisites:

- The WinSCP tool must be installed before performing the following process.

- The Private/Public keys must be generated and the Public key must be associated with your SFTP Account for the file upload/Download.

1. Launch WinSCP and select Tools > Run PuttyGen.

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

Figure 11–1   Key Generator
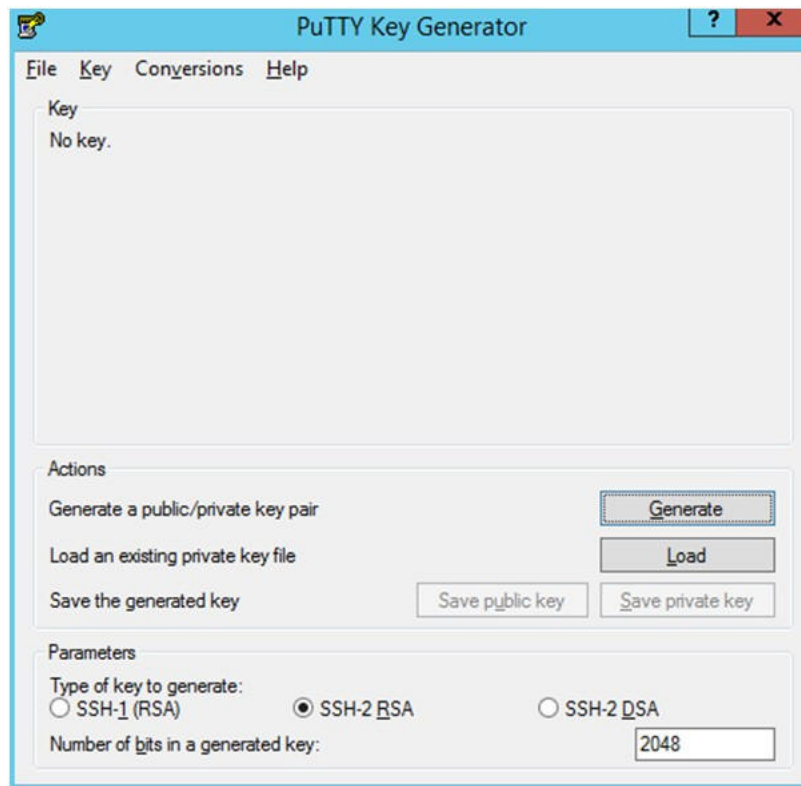
3. Move the mouse over the blank space in the window until the key is generated. Moving the mouse over the blank space creates a random pattern which is used for key generation.

Figure 11–2   Key Generator Progress
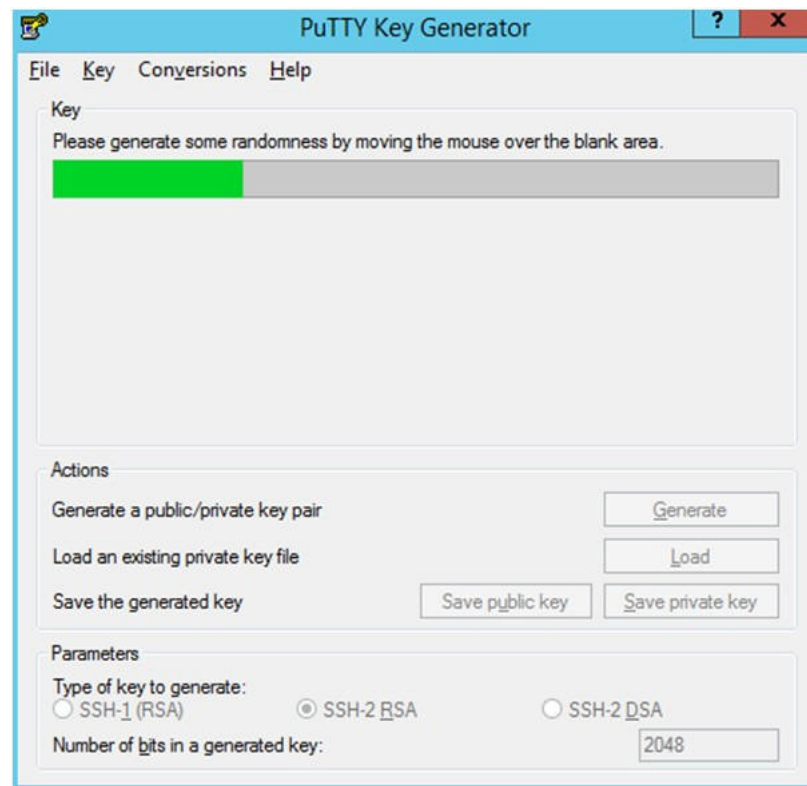
4. Once the key is generated, click **Save public key** to save the public key to a file.

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

6. Open an SR with Oracle Support, to associate the public key with your SFTP account (attach the key with the SR).

Logging In to WinSCP

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

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

2. Enter the user name and click **Advanced**.

3. Click **Authentication**.

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

Figure 11–3   Advanced Site Settings
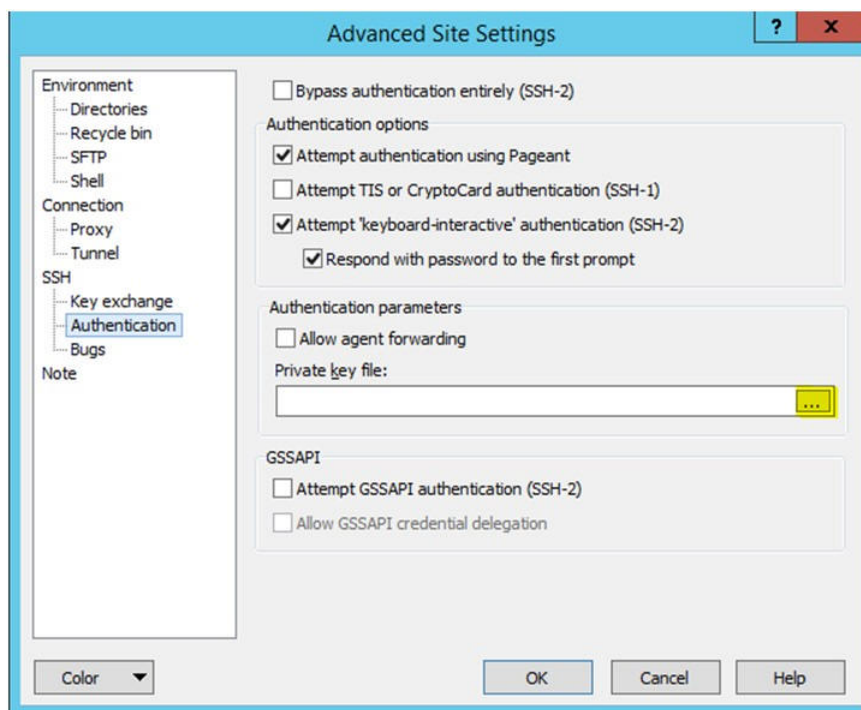
5.   After loading the private key file, click **OK**.

Figure 11–4   Private Key File Loaded
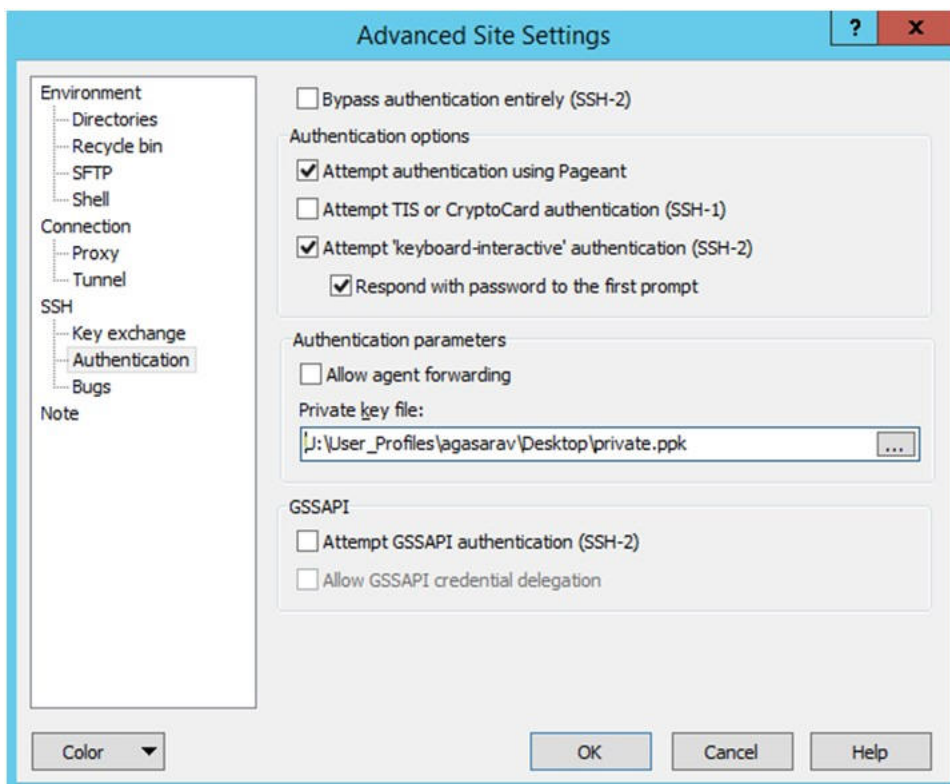
6. Click **Login**. The window does not prompt for a password and logs in to the SFTP server. Provide a passphrase if one has been set up.

Uploading the Batch File

The following diagram provides an overview of the FTP/internal holding area/domain and how the files move.

Figure 11–5   File Movement Between Internal Holding Area and Domain
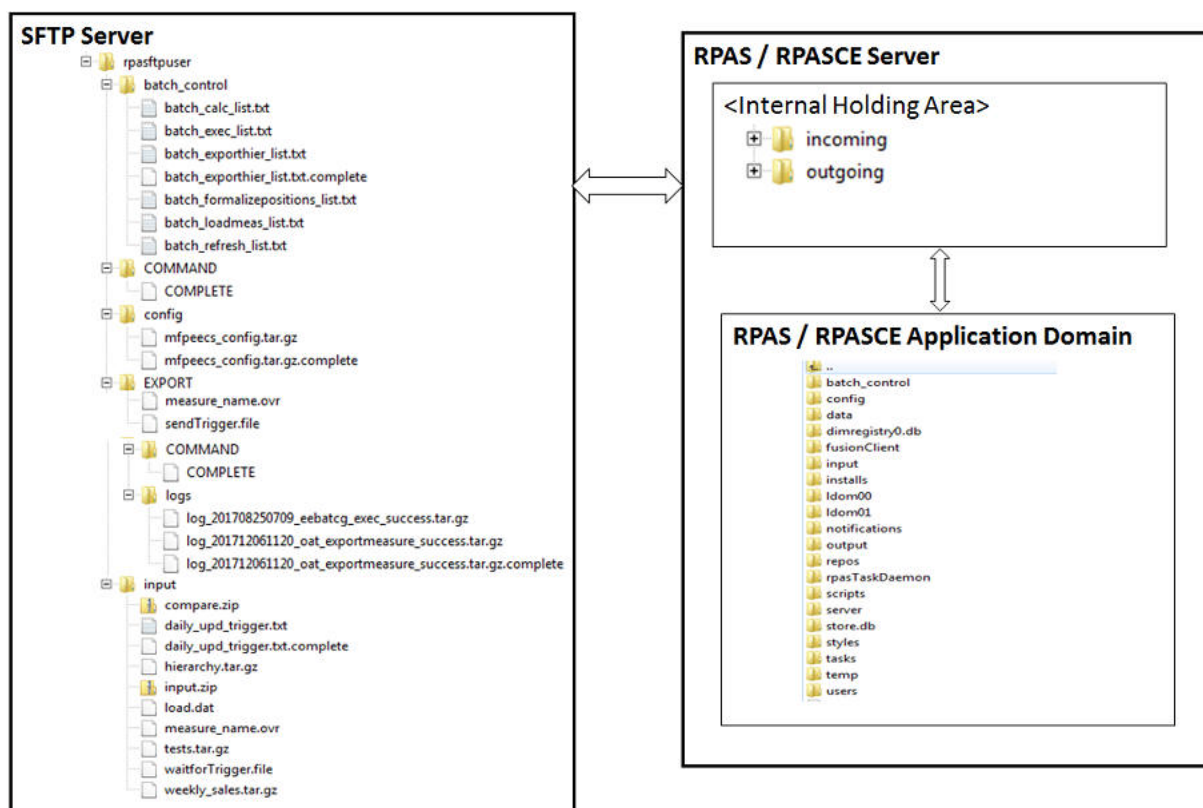
The following steps walk you through the process to upload batch (data/Hierarchy) file. For information about the administration tasks that require uploading batch files, see "Batch Administration."

For information about the file contents of various exports and formatting, see the information on data load in the *Oracle Retail Merchandise Financial Planning Cloud Service Implementation Guide*.

To upload the batch file:

1. Log in to WinSCP. Follow the steps in "Logging In to WinSCP."

2. Transfer all the data files to the directory /<SFTP User>.

3. Create a directory named COMMAND under /<SFTP User> if it does not already exist.

4. Change to the /<SFTP User>/COMMAND directory.

5. Transfer an empty file named COMPLETE.

6. Run the required batch/task from the Online Administration Tools.

File Downloads

The following steps walk you through the process of accessing download files. For information about the administration tasks that create different exports from the applications, see "Batch Administration."

For information about the file contents of various exports and formatting, see the information on exports in the *Oracle Retail Merchandise Financial Planning Cloud Service Implementation Guide*.

To download files:

1. Log in to WinSCP. Follow the steps in "Logging In to WinSCP."

2. Change the directory to /<SFTP User>/EXPORT or /<SFTP User>/Export/logs (for log files).

3. Create a directory named COMMAND under /<SFTP User>/EXPORT if it does not already exist.

4. Change to the /<SFTP User>/EXPORT/COMMAND directory.

5. Transfer an empty file named COMPLETE.

6. Run the required batch/task from the Online Administration Tools.

7. Download all the data files. The log files are in .zip format.

> **Note:** The log file names are appended with the success/fail text in it along with the time stamp for easy identification.
>
> For example: log_201802150047_eebatch_loadmeas_success.tar.gz and log_201802211218_eebatch_calc_fail.tar.gz

12

RPAS Test Automation

This chapter describes an RPAS utility called "rpac" (RPAS Pluggable Automation Component). It can be used to perform various tests on an existing domain, based on a set of test cases contained in XML files.

This appendix contains the following sections:

- Introduction
- RPAC in Batch Framework
- Writing Test Cases
- Example
- Output
- Schema
- RPAC Restricted Tags for Cloud Deployment Domain

Introduction

The rpac utility is used to run highly customizable automated test suites for the RPAS server. The XML-based framework allows rapid development of tests and ensures that test cases can be quickly copied across locations. The flexible XML schema allows the user to test a large number of RPAS features, such as writing data to a domain, building a workbook, running custom menu options, and verifying the results.

The XML framework also introduces consistency, accuracy, and reproducibility in the test results. To reproduce a test case, all that is needed is the XML file and a copy of the domain.

Note that rpac operates directly against the domain itself and therefore does not test any Client/Server interactions.

RPAC in Batch Framework

The rpac utility is supported for use with Enterprise Edition deployments. RPAC tests are specified in XML-format text files and cover a range of RPASCE Domain and Workbook activities. In order to support the validation of a newly installed or patched environment, in the context of configured daily or weekly batch operations, RPAC for Cloud deployments is supported through new entries in the EE Batch task catalog. These tasks allow a pre-production domain to be set to a known state through a combination of hierarchy load and measure load files, and then can compare both Domain and Workbook measures to known values represented either directly in the

test xml files or in data comparison files (similar to a measure data load file, but used only for comparison rather than for loading).

Three types of collateral files are involved in the RPAC testing process:

- Input data file set: a group of hierarchy (.dat) and measure (.ovr, .clr, or .rpl) data files that should be loaded into the domain before any RPAC tests are run. Uploaded to FTP site in the rpac directory as input.tar.gz or input.zip.

- Test file set: one or more .xml files where tests and test suites are defined using the available set of RPAC tags and attributes. Uploaded to FTP site in the rpac directory as tests.tar.gz or tests.zip.

- Comparison data file set: an optional way to efficiently validate that one or more measures currently contain an expected set of values. Uploaded to the FTP site in the rpac directory as compare.tar.gz or compare.zip.

Each of these collateral file archives, once sent through the FTP interface, will be kept internally to be used every time an RPAC-enabled batch execution sequence is run. Updates to the collateral files can be sent to the FTP site before the next call of the initrpac batch task and will be brought into the active environment at that time. Note that when any of the collateral file archives is updated, the previous contents are entirely removed from the internal storage area, so the replacement archive file must be a complete set of files of that type. (This prevents stale test scripts or data files from being left in the environment.)

The two EE Batch tasks, initrpac and runrpac, are detailed in the batch task catalog in "Initialize Testing Environment: initrpac" and "Execute Automated Tests: runrpac" in the Implementation Guide. The initrpac task is expected to be run once, at the start of the RPAC-enabled batch exec sequence; the runrpac task can be called multiple times, including at separate points during the batch exec sequence, if needed. Here is an example batch execution sequence that shows how an existing weekly batch specification might be augmented with RPAC tests:

```
# Standard Weekly Batch Cycle
weekly | unpack | weekly_sales.tar.gz~ftp
weekly | hierload | prod~14~N
weekly | hierload | loc~14~N
weekly | measload | load_oo_list
weekly | calc | batch_fcst
weekly | autobuild |
# RPAC-enhanced Batch Cyle
validate | initrpac |
validate | hierload | prod~14~N
validate | hierload | loc~14~N
validate | measload | load_oo_list
validate | runrpac | RPAC_Domain_Tests~DomainTests.xml
validate | calc | batch_fcst
validate | runrpac | RPAC_Workbook_Tests~WorkbookTests.xml
```

The first section, labeled "weekly", represents a weekly batch sequence that might run at midnight every Saturday. Note that updated hierarchy and measure data files for the week are sent through FTP in an archive file named "weekly_sales.tar.gz" using the unpack task.

The second section shows how the weekly batch sequence has been augmented with RPAC tests and named "validate". Note that the unpack task from the weekly sequence has been left out, and in its place initrpac is called to place the test data input files into the domain. If new or updated RPAC test collateral files have been placed on the FTP server, they will be brought in at this point and used.

There are two sets of RPAC tests in this sequence, specified by the runrpac task entries. The first runs immediately after the hierarchy and measure files are loaded, and validates expected values in the domain. The second test set is executed after some further calculations have been run, and builds one or more workbooks, then validates values within them as well.

When RPAC-enabled batch sequences are run, the primary log file (available through the Online Administration dashboard as well as through the FTP log archive package) will show a brief summary of test results. Full test details and log files are available in the complete log archive package from the batch exec run, available in the FTP area once the execution has completed.

Note that the latest version of this guide specifies which RPAC features are available for Cloud deployments. Due to Cloud security constraints, some RPAC features, primarily the <SHELL> tag, have been disabled; however, inclusion of RPAC tests as a step in existing batch execution sequences should fully compensate for this restriction.

Writing Test Cases

The structure of an rpac test script is defined by the following tags.

Table 12–1   Test Case Tags

Tag	Description
<testscript>	The outermost tag for the test script. This tag contains one or more <testsuite> tags.
<testsuite>	This tag defines a suite of tests that are designed to run together. Each suite contains its own <setup>, <teardown>, <testcase>, and <workbook-operations> tags. This tag has a name attribute that can be used with the -suite command line attribute to specify which suites are executed in this run.
<once>	The once tag contains tag elements used to set up the environment once before the first test case is executed. It is a tag like setup, but only happens once at the start of a test suite. The once tag has been added to allow a set of measures to be backed up once during the run of the test suite.
<finalize>	The finalize tag contains tag elements used to restore the environment after the final test case is executed.
<setup>	This tag sets the preconditions that are common to all test cases. Operations inside this tag are run before every test case is executed.
	Example operations:
	■ Set RpasToday so the test date is consistent for every test run.
	■ Pre-set measure values so tests operate in a known state.
	■ Check any preconditions to make sure this test run is valid.

Table 12–1   (Cont.) Test Case Tags

Tag	Description
<teardown>	This tag contains cleanup code that is run after all test cases. Operations inside this tag are run after every test case is executed. This is used to restore a domain or workbook to its default state to remove dependencies between independent test cases.
	Example operations:
	■ Set domain measure values back to their initial state.
	■ Run batch scripts to remove any test-specific files that were generated.
<workbook-operations>	This tag allows the test script to build, refresh, commit, and close workbooks.
	Workbook operations may exist outside of test cases, but you must encapsulate them wherever possible inside a testsuite tag for logging purposes.
<testcase>	This tag is where the code to perform each test goes. It should perform operations, then use the assert tags to verify the results. The name of this tag is used in the output to log timing results and success/failure. **Note:** Code in the <setup> tag is executed prior to each test case, and code in the <teardown> tag is executed after each test case.

The optional attribute role for tags testsuite, once, setup and teardown may be required for an rpac application being written for an RDM enable domain. The role assigned to the testsuite tag affects all test cases, and if no role is assigned to once, setup, and teardown, the role assigned to the testsuite tag will affect these tags as well. A role assigned to the once tag is applied to the tags contained in once, and if no role is assigned to setup and teardown, then this role will apply to these tags as well. A role assigned to the setup tag is applied to the tags contained in setup, and if no role is assigned to teardown, then this role will apply to teardown as well. A role assigned to the teardown tag only applies to the tags contain in teardown.

It is recommended that the teardown section restore the domain to its previous state. This removes dependencies between test cases and ensures consistency from one run to the next.

The tags once, setup, and teardown are unique. Only the first occurrence of each of these tags is used. These tags can appear in any order under the testsuite tag, and all of these tags are optional. The code contained in these tags is executed at a set point during the run of unit test code. However, for the tag testcase, the order of occurrence in the testsuite is the order of execution.

Example

This section provides a sample XML file that contains some rpac tests that run against an MFP domain. This test illustrates how the setup and teardown methods work, as well as how to use the various workbook operations.

The first test case demonstrates building the workbook. The first step in this operation is to shell out to the wbmgr utility to clear out all existing workbooks. This step prevents the domain from growing after repeated operations, but you should only use this step in cases where you know that existing workbooks are not needed.

After that, the test verifies the preconditions of the test case. In this example, the test verifies the preconditions performed in the setup code by checking that the domain measure value is "true". After that, the test makes the wizard selections and builds the workbook. Once the workbook is built, an assertion verifies that the load rules correctly set the workbook value.

Once this test case is complete, the teardown code is executed, along with the setup code for the next test case. The test verifies that the teardown code correctly set the workbook's measure value to "false" and then issues a commit. Once the workbook is committed, the domain value is checked to see if it was properly updated. Once again, the teardown code is executed and the setup code is run at the beginning of the next test case.

For the third test case, the setup code should have initialized the domain value back to "true", and the workbook value back to "false". The test case verifies these operations, then performs a workbook refresh and verifies that the workbook value has been correctly set to "true".

After that, the workbook is closed and the test suite is complete.

> **Note:** Because the teardown code modifies workbook data, the workbook close step cannot be encapsulated inside of a test case. If it was, the teardown code would try to modify a workbook that was no longer there and produce a failing result.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!---
  Copyright (c) 2004, 2012, Oracle and/or its affiliates. All rights reserved.
  File: wbtest.xml
  TestCase Description:
  - Setup:
  Set the RPAS Today value
  Set the domain's measure value to "true"
  - Teardown:
  Set the workbook's measure value to "false"
  - Build: Build a workbook and validate the load rule
  - Commit: Validate the commit rule
  - Refresh: Validate the refresh rule
  - Close the workbook
-->
<testscript>
  <testsuite name="MFP RPAC Test 11">
    <!--Setup executed at the beginning of every testcase-->
    <setup>
      <rpas-today>20070101</rpas-today>
      <edit-domain-measure-value name="ipopappenbb" namedkeys="h1_2007:100:1"
value="true"/>
    </setup>

    <testcase name="Enable OP Approval Workbook: Build">
      <!--Clear all existing workbooks-->
      <shell>wbmgr -d [DOMAIN] -remove -all</shell>

      <!--Verify pre-conditions are true-->
      <assert-domain-measure-value-eq name="ipopappenbb" namedkeys="h1_2007:100:1"
value="true"/>

      <workbook-operations>
        <build template-name="EnableOp" label="Enable OP Approval" user="adm">
          <!--Make positions selection for the wizard page-->
          <wizard-page-settings id="wiz_EnableOpssn">
            <set-tree-selections control-name="tree1" dim-name="ssn"
rollup-name="ssn">
              h1_2007
```
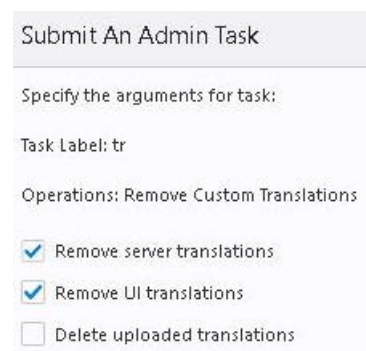
```
              </set-tree-selections>
            </wizard-page-settings>
            <wizard-page-settings id="wiz_EnableOpdept">
              <set-tree-selections control-name="tree1" dim-name="dept"
rollup-name="dept">
                  100
              </set-tree-selections>
            </wizard-page-settings>
            <wizard-page-settings id="wiz_EnableOpchnl">
                  1
            <set-tree-selections control-name="tree1" dim-name="chnl"
rollup-name="chnl" >
              </set-tree-selections>
            </wizard-page-settings>
          </build>
        </workbook-operations>

        <!--Validate the load rule worked correctly->
        <assert-workbook-measure-value-eq name="ipopappenbb" keys="0:0:0"
value="true"/>
      </testcase>

      <!--Commit the workbook and assert the measure values in domain-->
      <testcase name="Enable OP Approval Workbook: Commit">
        <assert-domain-measure-value-eq name="ipopappenbb" namedkeys="h1_2007:100:1"
value="true"/>
        <assert-workbook-measure-value-eq name="ipopappenbb" namedkeys="h1_
2007:100:1 value="false"/>
        <workbook-operations>
          <commit type ="now" template-name="EnableOp" label="Enable OP Approval"
user="adm"/>
        </workbook-operations>

        <!--Validate the commit rule worked correctly->
        <assert-domain-measure-value-eq name="ipopappenbb" namedkeys="h1_2007:100:1"
value="false"/>
      </testcase>

      <!--Refresh the workbook and assert measure values in workbook and domain-->
      <testcase name="Enable OP Approval Workbook: Refresh">
        <assert-domain-measure-value-eq name="ipopappenbb" namedkeys="h1_2007:100:1"
value="true"/>
        <assert-workbook-measure-value-eq name="ipopappenbb" namedkeys="h1_
2007:100:1" value="false"/>
        <workbook-operations>
          <refresh template-name="EnableOp" label="Enable OP Approval" user="adm"/>
        </workbook-operations>

        <!--Validate the refresh rule worked correctly->
        <assert-workbook-measure-value-eq name="ipopappenbb" namedkeys="h1_
2007:100:1" value="true"/>
      </testcase>

      <!--Close the workbook-->
      <workbook-operations>
        <close template-name="EnableOp" label="Enable OP Approval" user="adm"/>
      </workbook-operations>

      <!--Teardown executed at the end of every testcase-->
      <teardown>
```

```
         <edit-workbook-measure name="ipopappenbb" keys="0:0:0" value="false"/>
     </teardown>
  </testsuite>
</testscript>
```

Output

When rpac executes, it prints the status of the current execution to the console. Once all the test cases are complete, the summary of test cases is stored in unitest.xml under the current directory. These results contain the failing test case, as well as any error messages that were captured. If -coverage was specified, then a coverage.xml file is generated containing the passing and failing results.

Example:

```
[rpasHome] rpac -d testDomain -input wbTest.xml -title "'MFP OpApproval RPAC Test"

Running rpac tests for RPAC : MFP Tests...

Now executing wbmgr in domain.
Deleting all workbooks in the domain...
Destroying workbook 't0000000000' in domain.
wbmgr completed successfully
5.292541
0.308167
0.244300


Ran 3   tests: 0   errors, 0   failures in 5.85 seconds
```

If something had gone wrong, the output would look like this:

Example:

```
 [rpasHome] rpac -d testDomain -input wbTest.xml -title "'MFP OpApproval RPAC
Test"

Now executing wbmgr in domain
Deleting all workbooks in the domain...
Destroying workbook 't0000000000' in domain
wbmgr completed successfully

Failure
0.192497
Failure

Ran 3   tests: 0   errors, 2   failures in 6.42 seconds

Contents of unittest.xml
After execution, rpac creates a unittest.xml file containing the test results.
Here is an example from the successful execution above:

<unittest name='MFP OpApproval RPAC Test' runcount='3' timestamp='2012-10-17
10:06:49' elapsedTime='13.72 seconds'>
</unittest>
```

Here is the unittest.xml output from the failing run:

```
<unittest name='MFP OpApproval RPAC Test' runcount='3' timestamp='2012-10-17
10:07:38' elapsedTime='6.42 seconds'>
  <failurelist>
    <failure>
```

```
    <testname> </testname>
    <message>Got Error: Expected measure &apos;ipopappenbb&apos; at ArrayKey(0,
0, 0, 0) is false;
                        false == true</message>
  <file>OpApproval.xml</file>
    <line>52</line>
  </failure>
  <failure>
    <testname> </testname>
    <message>Got Error: Expected measure &apos;ipopappenbb&apos; at ArrayKey(0,
0, 0, 0) is false;
                        false == true</message>
    <file>OpApproval.xml</file>
    <line>60</line>
  </failure>
  </failurelist>
</unittest>
```

Schema

This section provides schema details.

1. Top-Level Parent Tag

This section provides details about the top-level parent tag.

1.1 testscript

The testscript is the top-level tag, which contains all the rpac tags.

1.2 testsuite

The testsuite contains setup, teardown, and testcase tag elements.

1.3 testcase

The testcase contains a list of tag elements to be executed.

1.4 once

The once tag contains tag elements used to set up the environment once before the first test case is executed. It is a tag like setup, but only happens once at the start of a test suite. The once tag has been added to allow a set of measures to be backed up once during the run of the test suite.

1.5 finalize

The finalize tag contains tag elements used to restore the environment after the final test case is executed.

1.6 setup

The setup tag is used to set any preconditions that must be true prior to running each test case. These can include initializing measure values, setting the date, or running batch scripts to prepare the domain.

1.7 teardown

The teardown tag contains tag elements used to restore the environment to a default state after a test case is executed. This can include returning measures to their default values or running batch scripts to clean up the domain.

1.8 log

The log tag logs a message to the console. It has an optional attribute, msg, containing the message to be logged.

Sample Usage

Here is the sample implementation of the above tags

```
<testscript>
<testsuite name="MFP RPAC Test">
<setup>
<!--- setup code ->
</setup>
<testcase name="Measure Analysis Workbook : build">
<!---testcase code ->
</testcase >
<teardown>
<!--- teardown code ->
</teardown>
<finalize>
    <log msg="Do Finalize" />
</finalize>
</testsuite>
</testscript>
```

2. Generic Tags

This section provides details about generic tags.

2.1 Attribute

The attribute tag specifies the attributes of a tag in tagdoc.xml.

2.2 Description

The description describes the functionality of each tag and its attributes in tagdoc.xml.

2.3 Fail

Fail creates a failure message.

Attribute	Description	Category
msg	Message to be used in the failure text	Optional

2.4 key

Sets the key specs value of the named key.

Attribute	Description	Category
name	Key name	Required
keyspec	Key value spec	Required

2.5 rpas-today

Sets the RpasToday environment variable to the value of this tag. RPAS_TODAY tells RPAS what day that it should think today is.

Sample usage:

```
<rpas-today>20070101</rpas-today>
```

Where: the date is given in yyyymmdd format.

2.6 shell

The shell tag starts a shell process with the command provided as value of this tag.

Attribute	Description	Category
working-directory	Directory to use for shell; if not provided then the current working directory is used.	Optional
check-status	Boolean to determine if a non-zero return status throws an exception.	Optional
is-shell	Boolean to indicate that the process is a shell.	Optional
assert-status-equal	Assert return status equals provide value.	Optional
assert-status-not-equal	Assert return status does not equals provide value.	Optional
output-file	File to which shell command redirects its output.	Optional

Sample usage:

```
<shell check-status="true" is-shell="false">wbmgr -d [DOMAIN] -remove -all</shell>
```

2.7 register-measure

Registers a measure with the provided properties.

Attribute	Description	Category
name	Measure name	Required
Intx	Base intersection	Required
type	Type of measure	Required
naval	NA value of the measure	Required
defagg	Default aggregation	Required

Sample usage:

```
<register-measure name="MeasureX" intx="WEEK_STR_DAY_" type="int"  defagg="total"
naval="0"/>
```

2.8 unregister-measure

Unregisters a measure registered by rpac.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<unregister-measure name="MeasureX" />
```

3. Domain Operation Tags

This section provides details about domain operation tags.

3.1 Assertions

This section provides details about assertions.

3.1.1 assert-dimension-contain Assert a dimension contains or does not contain a position name. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Dimension name	Required
posname	List of comma-separated position names	Required
containflag	If flag is true, then position names are checked to see if they are contained in the dimension.	Required

Sample usage:

```
<assert-dimension-contain name="DEPT" posname="dept1" containflag="true" />
```

3.1.2 assert-dimension-exists Assert a dimension name exists. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Dimension name	Required

Sample usage:

```
<assert-dimension-exists name="DEPT" />
```

3.1.3 assert-dimension-size Assert a dimension size is between a minimum and a maximum value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Dimension name	Required
min	Minimum value	Required
maz	Maximum value	Required

Sample usage:

```
<assert-dimension-size name="DEPT" min="8" max="32" />
```

3.1.4 assert-domain-measure-exists Assert a domain measure exists.

Attribute	Description	Category
name	Dimension name	Required

Sample usage:

```
<assert-domain-measure-exists name="r_ex_test" />
```

3.1.5 assert-domain-measure-intx Assert a domain measure base intersection matches an expected value.

Attribute	Description	Category
name	Dimension name	Required
intx	Intersection string to compare with	Required

Sample usage:

```
<assert-domain-measure-intx name="r_ex_test" intx="str_sku_week"/>
```

3.1.6 assert-domain-measure-value-eq Assert a domain measure value is equal to an expected value. This tag can be run in a number of ways:

Check against a loadmeasure file:

By specifying <path>, <start>, and <width>, you can validate that the contents of a measure are equal to the contents of a fixed-width loadmeasure file.

Check against specified positions:

By providing <keys> or <namedkeys>, you can verify that all of the positions in your specification are equal to the provided value. You can specify <keyint> to provide positions at an alternate rollup.

Check entire measure:

If you do not supply the <keys>, <namedkeys>, or <path> tag, then rpac will assert that every logical cell in the measure is equal to the value provided.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Sample usage (using keys):

```
<assert-domain-measure-value-eq name="bulylagtx" keyint="year" namedkeys=" 2007"
value="Test1"/>
```

Sample usage (using path):

```
<assert-domain-measure-value-eq name="bulylagtx" path="loadmeas.dat" start="100"
width="20" />
```

Sample usage (check all):

```
<assert-domain-measure-value-eq name="bulylagtx" value="Test1" />
```

3.1.7 assert-domain-measure-value-ge Assert a domain measure value is greater than or equal to an expected value. This tag can be run in a number of ways:

Check against a loadmeasure file:

By specifying <path>, <start>, and <width>, you can validate that the contents of a measure are equal to the contents of a fixed-width loadmeasure file.

Check against specified positions:

By providing <keys> or <namedkeys>, you can verify that all of the positions in your specification are equal to the provided value. You can specify <keyint> to provide positions at an alternate rollup.

Check entire measure:

If you do not supply the <keys>, <namedkeys>, or <path> tag, then rpac will assert that every logical cell in the measure is equal to the value provided.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Sample usage (using keys):

```
<assert-domain-measure-value-ge name="bulylagtx" keyint="year" namedkeys=" 2007"
value="Test1"/>
```

Sample usage (using path):

```
<assert-domain-measure-value-ge name="bulylagtx" path="loadmeas.dat" start="100"
width="20" />
```

Sample usage (check all):

```
<assert-domain-measure-value-ge name="bulylagtx" value="Test1" />
```

3.1.8 assert-domain-measure-value-gt Assert a domain measure value is greater than a an expected value. This tag can be run in a number of ways:

Check against a loadmeasure file:

By specifying <path>, <start>, and <width>, you can validate that the contents of a measure are equal to the contents of a fixed-width loadmeasure file.

Check against specified positions:

By providing <keys> or <namedkeys>, you can verify that all of the positions in your specification are equal to the provided value. You can specify <keyint> to provide positions at an alternate rollup.

Check entire measure:

If you do not supply the <keys>, <namedkeys>, or <path> tag, then rpac will assert that every logical cell in the measure is equal to the value provided.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Sample usage (using keys):

```
<assert-domain-measure-value-gt name="bulylagtx" keyint="year" namedkeys=" 2007"
value="Test1"/>
```

Sample usage (using path):

```
<assert-domain-measure-value-gt name="bulylagtx" path="loadmeas.dat" start="100"
width="20" />
```

Sample usage (check all):

```
<assert-domain-measure-value-gt name="bulylagtx" value="Test1" />
```

3.1.9 assert-domain-measure-value-ie Assert a domain measure value is less than or equal to an expected value. This tag can be run in a number of ways:

Check against a loadmeasure file:

By specifying <path>, <start>, and <width>, you can validate that the contents of a measure are equal to the contents of a fixed-width loadmeasure file.

Check against specified positions:

By providing <keys> or <namedkeys>, you can verify that all of the positions in your specification are equal to the provided value. You can specify <keyint> to provide

positions at an alternate rollup.

Check entire measure:

If you do not supply the <keys>, <namedkeys>, or <path> tag, then rpac will assert that every logical cell in the measure is equal to the value provided.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Sample usage (using keys):

```
<assert-domain-measure-value-le name="bulylagtx" keyint="year" namedkeys=" 2007"
value="Test1"/>
```

Sample usage (using path):

```
<assert-domain-measure-value-le name="bulylagtx" path="loadmeas.dat" start="100"
width="20" />
```

Sample usage (check all):

```
<assert-domain-measure-value-le name="bulylagtx" value="Test1" />
```

3.1.10 assert-domain-measure-value-lt Assert a domain measure value is less than an expected value. This tag can be run in a number of ways:

Check against a loadmeasure file:

By specifying <path>, <start>, and <width>, you can validate that the contents of a

measure are equal to the contents of a fixed-width loadmeasure file.

Check against specified positions:

By providing <keys> or <namedkeys>, you can verify that all of the positions in your specification are equal to the provided value. You can specify <keyint> to provide positions at an alternate rollup.

Check entire measure:

If you do not supply the <keys>, <namedkeys>, or <path> tag, then rpac will assert that every logical cell in the measure is equal to the value provided.

Attribute	Description	Category
name	Measure name	Required

Attribute	Description	Category
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Sample usage (using keys):

```
<assert-domain-measure-value-lt name="bulylagtx" keyint="year" namedkeys=" 2007"
value="Test1"/>
```

Sample usage (using path):

```
<assert-domain-measure-value-lt name="bulylagtx" path="loadmeas.dat" start="100"
width="20" />
```

Sample usage (check all):

```
<assert-domain-measure-value-lt name="bulylagtx" value="Test1" />
```

3.1.11 assert-domain-measure-value-ne Assert a domain measure value is not equal to an expected value. This tag can be run in a number of ways:

Check against a loadmeasure file:

By specifying <path>, <start>, and <width>, you can validate that the contents of a measure are equal to the contents of a fixed-width loadmeasure file.

Check against specified positions:

By providing <keys> or <namedkeys>, you can verify that all of the positions in your

specification are equal to the provided value. You can specify <keyint> to provide positions at an alternate rollup.

Check entire measure:

If you do not supply the <keys>, <namedkeys>, or <path> tag, then rpac will assert that every logical cell in the measure is equal to the value provided.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional

Attribute	Description	Category
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Sample usage (using keys):

```
<assert-domain-measure-value-ne name="bulylagtx" keyint="year" namedkeys=" 2007"
value="Test1"/>
```

Sample usage (using path):

```
<assert-domain-measure-value-ne name="bulylagtx" path="loadmeas.dat" start="100"
width="20" />
```

Sample usage (check all):

```
<assert-domain-measure-value-ne name="bulylagtx" value="Test1" />
```

3.1.12 assert-domain-popcount-eq Assert a domain measure popcount matches an expected value.

Attribute	Description	Category
name	Measure name	Required
pop-count	Expected popcount	Required

Sample usage:

```
<assert-domain-popcount-eq name=" Ipopappenbb" pop-count="100" />
```

3.1.13 assert-domain-popcount-eq-zero Assert a domain measure is not populated.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<assert-domain-popcount-eq-zero name=" Ipopappenbb" />
```

3.1.14 assert-domain-popcount-ne-zero Assert a domain measure is populated.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<assert-domain-popcount-ne-zero name=" Ipopappenbb" />
```

3.1.15 assert-bitsize-value-eq Assert the dimInfo bitsize or dimregistry bitsize value of a dimension. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	List of comma-separated dimension names	Required
category	Bit Size Category (dimInfo or dimRegistry)	Required
value	comma-separated Bit Size Values	Required

Sample usage:

```
<assert-bitsize-value-eq name="week,skup," category="dimInfo" value="12,14" />
```

3.1.16 assert-dim-registry-version Assert the value for dimension registry version. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
value	Registry version	Required

Sample usage:

```
<assert-dim-registry-version value="1" />
```

3.1.17 assert-reindex-threshold Assert the threshold value of the dimension.

Attribute	Description	Category
name	List of comma-separated dimension names	Required
value	List of comma-separated threshold values	Required

Sample usage:

```
<assert-reindex-threshold name="scls,week" value="25,103" />
```

3.1.18 assert-reindex-required Assert whether Diminfo Bitsize is equal to DimRegistry Bitsize.

Attribute	Description	Category
bitsize-flag	Returns true if a Bitsize mismatch occurs	Required
name	Dimension name	Required

Sample usage:

```
<assert-reindex-required name="week" bitsize-flag="false" />
```

3.1.19 assert-reindex-in-progress Assert the status of reindex with the user input.

Attribute	Description	Category
flag	Flag to indicate whether reindexing Domain is in progress.	Required

Sample usage:

```
<assert-reindex-in-progress flag="false" />
```

3.1.20 set-current-domain Set the current domain to either the master or a sub-domain.

Attribute	Description	Category
index	Domain index	Required

Sample usage:

```
< set-current-domain index="0" />
```

3.1.21 backup-domain-measures Back up a list of measures.

Attribute	Description	Category
names	List of measure names	Required

Sample usage:

```
<backup-domain-measures  names="UTDyDst03U,UTDyDst05U,tdwpunelapbopb,bucpgmpp"/>
```

3.1.22 restore-domain-measures Restore a list of measures.

Attribute	Description	Category
names	List of measure names	Required

Sample usage:

```
<restore-domain-measures names="UTDyDst03U,UTDyDst05U,tdwpunelapbopb,bucpgmpp"/>
```

3.1.23 assert-domain-hier-exists Assert a hierarchy exists in domain or not.

Attribute	Description	Category
name	Hierarchy name	Required
exists	If flag is true, then check if hierarchy exists in domain.	Required

Sample usage:

```
<assert-domain-hier-exists name="clnd" exists="true"/>
```

3.2 Domain Data Edit Operations

This section provides details about domain data edit operations.

3.2.1 clear-all-measures Clears all measures used in the following tags:
<edit-workbook-measure>, <set-workbook-measure>, <set-domain-measure>.

This sets all the contents of these measures to the NA value.

3.2.2 clear-domain-measure Clear a domain measure of all data. This sets all the contents of this measure to its NA value.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<clear-domain-measure name="r_ex_pick_real" />
```

3.2.3 clear-domain-measure-list Clear a list of domain measures of all data.

Attribute	Description	Category
path	Path to a file containing a new-line delimited list of measures	Required

Sample usage:

```
<clear-domain-measure-list path="measlist.txt" />
```

3.2.4 execute-expression Execute an expression.

Attribute	Description	Category
expression	The expression to execute/calculate. This is a full expression string,	Optional
incremental	True if this is to be an incremental evaluation.	Optional

Sample usage:

```
<execute-expression expression="a = b+c" incremental="false" />
```

3.2.5 set-domain-measure Set the value of a domain measure based on provided attributes.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

Attribute	Description	Category
incremental	True if evaluation is to be incremental	Optional

> **Note:** Note that for Date type measures, RPAC supports setting the array cell using either of the following two Date & Time formats - "%m/%d/%Y" or "%m/%d/%Y-%H:%M:%S" where m = 2 digit month, d = 2 digit date, Y = 4 digit year, H = 2 digit hour, M = 2 digit min, and S = 2 digit sec.

Sample usage:

```
<set-domain-measure name="r_ex_pick_real" keyint="dept" keys="1" value="150" type="real" />
```

Sample usage (using keys):

```
<set-domain-measure name="meas1" keys="0-2:5" value="0.0" />
```

Sample usage (using keyint):

```
<set-domain-measure-value-ne name="meas2" namedkeys="chnl1:week1" value="02/24/2012" />
```

Sample usage (load-measure file):

```
<set-domain-measure name="meas1" path="loadMeas1.dat" start="100" width="20"/>
```

Sample usage (set all):

```
<set-domain-measure name="meas1" value="2"
```

3.2.6 dump-domain-measure Dump a measure to a text file.

Attribute	Description	Category
name	Measure name	Required
path	File path to text file to be used	Required
noValues	Flag to disable dumping of cell values.	Optional

Sample usage:

```
<dump-domain-measure name="utdydst03u" path="test3.txt" noValues="true"/>
```

3.2.7 load-domain-measure Load a measure from a text file.

Attribute	Description	Category
name	Measure name	Required
path	File path to text file to be used	Required

Sample usage:

```
<load-domain-measure name="utdydst03u" path="test3.txt" />
```

4. Workbook Operation Tags

This section provides details about workbook operation tags.

4.1 Workbook Wizard Operations

This section provides details about workbook wizard operations.

4.1.1 build Builds workbook based on provided tag data.

Attribute	Description	Category
template-name	Template name	Required
label	Label for workbook	Required
user	User used to access workbook	Required

Sample usage:

```
<build template-name="MeasureAnalysis" label="MEAS AN" user="adm">
```

4.1.2 wizard-page-settings Contains the settings for a wizard page.

Attribute	Description	Category
id	ID use to match up settings with wizard pages. For configured templates, this follows the pattern: wiz_<wbt_name><base_dim> Example: For the EnableOp workbook, the wizard page that sets CHNL is: Fwiz_EnableOpchnl Note: For custom templates, obtain the name from the administrator.	Required

Sample usage:

```
<wizard-page-settings id="wiz_EnableOpchnl">
```

4.1.3 set-hier-selection Sets the selection of a SingleHier Select control of a custom wizard page.

Attribute	Description	Category
control-name	Name of the control for a wizard page. This can be obtained from the template.cfg file.	Required
dim-name	Name of the dimension containing the selection.	Required
selection	Name of the position selected.	Required

Sample usage:

```
<set-hier-selection control-name="loc_select" dim-name="chnl"  selection="1" >
```

4.1.4 set-selected Sets the value for a Boolean control on a custom wizard page.

Attribute	Description	Category
control-name	Name of the control for a wizard page. This can be obtained from the configuration	Required
selected	Set this to TRUE or FALSE	Required

Sample usage:

```
<set-selected control-name="UseExtraMeasures" value="TRUE">
```

4.1.5 set-selection Sets the selection of a drop-down list control on a custom wizard page.

Attribute	Description	Category
control-name	Name of the control for a wizard page. This can be obtained from the configuration.	Required
selection	Text for the selection operation.	Required

Sample usage:

```
<set-selection control-name="ForecastLevel" value="2">
```

4.1.6 set-selections Sets the selections of a list box or tree control on a custom wizard page. This tag should contain a space-delimited list of the selections.

Attribute	Description	Category
control-name	Name of the control for a wizard page. This can be obtained from the configuration.	Required

Sample usage:

```
<set-selections control-name="ExtraMeasures">
meas1 meas2 meas3 </set selections>
```

4.1.7 set-text Sets the contents of an edit control on a custom wizard page.

Attribute	Description	Category
control-name	Name of the control for a wizard page	Required
test	Text for the operation	Required

Sample usage:

```
<set-text control-name="MyText" value="abc">
```

4.1.8 set-tree-selections Sets the selections for a two-tree control on either a standard or custom wizard page.

Attribute	Description	Category
control-name	Name of the control. For two-tree controls, this should always be set to "tree1"	Required
rollup-name	The highest level of the wizard page.	Required
dim-name	Name of the dimension on which the selections are made.	Required

Sample usage:

```
<set-tree-selections control-name="tree1" rollup-name="chnl" dim-name="str">
str1,str2,str3
</set-tree-selections>
```

4.2 Workbook Operations

This section provides details about workbook operations.

4.2.1 Workbook-operations Contains the workbook operations to be executed.

4.2.2 Calc Does workbook calculations.

Attribute	Description	Category
template-name	Template name	Required
label	Label for workbook	Required
user	User used to access workbook	Required

Sample usage:

```
<calc template-name="RptAdmin" label="Reporting Administration" user="adm"/
```

4.2.3 custom-menu Checks for the existence of, or executes, a custom menu.

Attribute	Description	Category
template-name	Template name	Required
label	Label for workbook	Required
user	User used to access workbook	Required
menu-label	Menu label for this custom menu. This can be obtained from the configuration	Required
menu-exist	Set this to "true" to see if the menu item exists, or leave it blank to execute the custom menu.	Optional

Sample usage:

```
<custom-menu template-name="BottomUp" label="Botup" user="adm"
menu-label="menuItem83"  menu-exist="true"/>
```

4.2.4 refresh Does workbook refresh.

Attribute	Description	Category
template-name	Template name	Required
label	Label for workbook	Required
user	User used to access workbook	Required
rule-group	Set this to use a rule group other than the default refresh group.	Optional

Sample usage:

```
<refresh template-name="BottomUp" label="Botup" user="adm" rule-group="refresh_
alt"/>
```

4.2.5 Close Does workbook close. Arguments may be provided for debug purposes.

Attribute	Description	Category
template-name	Template name	Optional
label	Label for workbook	Optional
user	User used to access workbook	Optional

Sample usage:

```
<close template-name="BottomUp" label="Botup" user="adm"/>
```

4.2.6 Does workbook commit Does workbook commit.

Attribute	Description	Category
type	Commit type (now/later)	Required
template-name	Template name	Required
label	Label for workbook	Required
user	User used to access workbook	Required

Sample usage:

```
<commit type="now"  template-name="BottomUp" label="Bottom Up MFP Refresh"
user="adm" />
```

4.3 Workbook Assertions

This section provides details about workbook assertions.

4.3.1 assert-window-contain-measure Asserts that the measures provided are visible (or not visible) on the window.

Attribute	Description	Category
name	Name of the window in the workbook configuration. User-generated windows cannot be used.	Required
measure	Comma-separated list of measures names.	Optional

Attribute	Description	Category
containflag	If true, will check that measures provided are visible. If false, will check that the measures provided are not visible.	Optional

Sample usage:

```
<assert-window-contain-measure  name=" chnldeptssn__W"  measures =" Ipopappenbb"/>
```

4.3.2 assert-window-exists Asserts if a window exists in the current workbook.

Attribute	Description	Category
name	Window name of current workbook	Required

Sample usage:

```
<assert-window-exists name="chnldeptssn__W" />
```

4.3.3 assert-window-intersection Asserts window intersection matches provided intersection.

Attribute	Description	Category
name	Window name of current workbook.	Required
baseint	Intersection string to compare. Leave blank if checking for scalar.	Optional

Sample usage:

```
<assert-window-intersection name="chnldeptssn__W" baseint"chnldeptssn_" />
```

4.3.4 assert-window-not-exists Asserts if a window does not exist in the current workbook.

Attribute	Description	Category
name	Window name of current workbook	Required

Sample usage:

```
<assert-window-not-exists name="chnldeptssn__W" />
```

4.3.5 assert-window-measure-exists Asserts a measure exists.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<assert-measure-exists name=" Ipopappenbb" />
```

4.3.6 assert-workbook-measure-intx Asserts a workbook measure base intersection matches an expected value.

Attribute	Description	Category
name	Measure name	Required
intx	Intersection string to compare with	Required

Sample usage:

```
<assert-measure-intersection name=" Ipopappenbb" baseint"chnldeptssn_" />
```

4.3.7 assert-workbook-measure-value-eq Asserts that a workbook measure at the specified positions is equal to the provided value.

Parent tag(s): testcase

Child tag(s): None

Attribute	Description	Category
keys	The keys to be used to compare measure values. Can be either indices or position names. Each dimension spec in the keys-spec is delimited by a ":". Multiple position ranges for a dimension can be specified by using "," as separators.	Required
	When using indices, a range of positions can also be used.	
	Ex - "0-2:4-7,11-13:4" implies indices 0,1,2 for the first dimension, 4,5,6,7,11,12,13 for the second, and 4 for the third dimension of the measure.	
	Note: Because indices are based on the workbook selections, they may not be consistent across all test cases. Care must be exercised to ensure that you are using the correct indices for your specific test case.	
namedkeys	Position names used to get keys of the measure array. Using indices instead of position names or positions that do not exist in the MeasureStore (Workbook) will be flagged as an error. Similar to "keys" above, but using position names only.	Optional
path	Path of the flat file used for comparison against the Measure array. This looks like pointing to a pre-set environment variable. "path" cannot be used along with "keys" or "namedkeys". "path" is used only to compare the measure data against a pre-existing flat file. If none of "path","keys" and "namedkeys" are specified, the comparison occurs for all the keys in the measure array of the Measure Store (Workbook).	Optional

Attribute	Description	Category
start	The field start value. If this does not exist in the XML, the value will be taken from the measure's properties. Used in flat-file comparison.	Optional
width	The field start value. If this does not exist in the XML, the value will be taken from the measure's properties. Used in flat-file comparison.	Optional
value	The expected value of the measure. Comparison on equality.	Optional

Sample usage: (indices)

```
<assert-workbook-measure-value-eq name="drwppthrsclsp" keys="0-2"  value="0.0" />
```

Sample usage (names):

```
<assert-workbook-measure-value-eq name="drwppthrsclsp" namedkeys="pos1,pos2,pos3"
value="0.0" />
```

4.3.8 assert-workbook-measure-value-ge Asserts a workbook measure value at the specified positions is greater than or equal to the provided value.

Attribute	Description	Category
keys	The keys to be used to compare measure values. Can be either indices or position names. Each dimension spec in the keys-spec is delimited by a ":". Multiple position ranges for a dimension can be specified by using "," as separators. When using indices, a range of positions can also be used. Ex - "0-2:4-7,11-13:4" implies indices 0,1,2 for the first dimension, 4,5,6,7,11,12,13 for the second, and 4 for the third dimension of the measure. **Note:** Because indices are based on the workbook selections, they may not be consistent across all test cases. Care must be exercised to ensure that you are using the correct indices for your specific test case.	Required
namedkeys	Position names used to get keys of the measure array. Using indices instead of position names or positions that do not exist in the MeasureStore (Workbook) will be flagged as an error. Similar to "keys" above, but using position names only.	Optional

Attribute	Description	Category
path	Path of the flat file used for comparison against the Measure array. This looks like pointing to a pre-set environment variable. "path" cannot be used along with "keys" or "namedkeys". "path" is used only to compare the measure data against a pre-existing flat file. If none of "path","keys" and "namedkeys" are specified, the comparison occurs for all the keys in the measure array of the Measure Store (Workbook).	Optional
start	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in flat-file comparison.	Optional
width	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in a flat-file comparison.	Optional
value	The expected value of the measure. Comparison on equality.	Optional

Sample usage (indices):

```
<assert-workbook-measure-value-ge name="drwppthrsclsp" keys="0-2"  value="0.0" />
```

Sample usage (names):

```
<assert-workbook-measure-value-ge name="drwppthrsclsp" namedkeys="pos1,pos2,pos3"
value="0.0" />
```

4.3.9 assert-workbook-measure-value-gt Asserts a workbook measure value is greater than a provided value.

Attribute	Description	Category
keys	The keys to be used to compare measure values. Can be either indices, or position names. Each dimension spec in the keys-spec is de-limited by a ":". Multiple position ranges for a dimension can be specified by using "," as separators.	Required
	When using indices, a range of positions can also be used.	
	Ex - "0-2:4-7,11-13:4" implies indices 0,1,2 for the first dimension, 4,5,6,7,11,12,13 for the second and 4 for the third dimension of the measure.	
	Note: Because indices are based on the workbook selections, they may not be consistent across all test cases. Care must be exercised to ensure that you are using the correct indices for your specific test case.	

Attribute	Description	Category
namedkeys	Position names used to get keys of the measure array. Using indices instead of position names or positions that do not exist in the MeasureStore (Workbook) will be flagged as an error. Similar to "keys" above, but using position names only.	Optional
path	Path of the flat file used for comparison against the Measure array. This looks like pointing to a pre-set environment variable. "path" cannot be used along with "keys" or "namedkeys". "path" is used only to compare the measure data against a pre-existing Flat file. If none of "path","keys" and "namedkeys" are specified, the comparison occurs for all the keys in the measure array of the Measure Store (Workbook).	Optional
start	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in a flat-file comparison.	Optional
width	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in flat-file comparison.	Optional
value	The expected value of the measure. Comparison on equality.	Optional

Sample usage (indices):

```
<assert-workbook-measure-value-gt name="drwppthrsclsp" keys="0-2"  value="0.0" />
```

Sample usage (names):

```
<assert-workbook-measure-value-gt name="drwppthrsclsp" namedkeys="pos1,pos2,pos3"
value="0.0" />
```

4.3.10 assert-workbook-measure-value-le Asserts a workbook measure value is greater than or equal to a provided value.

Attribute	Description	Category
keys	The keys to be used to compare measure values. Can be either indices, or position names. Each dimension spec in the keys-spec is delimited by a ":". Multiple position ranges for a dimension can be specified by using "," as separators. When using indices, a range of positions can also be used. Ex - "0-2:4-7,11-13:4" implies indices 0,1,2 for the first dimension, 4,5,6,7,11,12,13 for the second, and 4 for the third dimension of the measure. **Note:** Because indices are based on the workbook selections, they may not be consistent across all test cases. Care must be exercised to ensure that you are using the correct indices for your specific test case.	Required
namedkeys	Position names used to get keys of the measure array. Using indices instead of position names, or positions that do not exist in the MeasureStore (Workbook) will be flagged as an error. Similar to "keys" above, but using position names only.	Optional
path	Path of the flat file used for comparison against the Measure array. This looks like pointing to a pre-set environment variable. "path" cannot be used along with "keys" or "namedkeys". "path" is used only when we want to compare the measure data against a pre-existing flat file. If none of "path","keys" and "namedkeys" are specified, the comparison occurs for all the keys in the measure array of the Measure Store (Workbook).	Optional
start	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in a flat-file comparison.	Optional
width	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in flat-file comparison.	Optional
value	The expected value of the measure. Comparison on equality.	Optional

Sample usage (indices):

```
<assert-workbook-measure-value-le name="drwppthrsclsp" keys="0-2"  value="0.0" />
```

Sample usage (names):

```
<assert-workbook-measure-value-le name="drwppthrsclsp" namedkeys="pos1,pos2,pos3"
```

```
value="0.0" />
```

4.3.11 assert-workbook-measure-value-lt Asserts a workbook measure value is less than a provided value.

Attribute	Description	Category
keys	The keys to be used to compare measure values. Can be either indices, or position names. Each dimension spec in the keys-spec is de-limited by a ":". Multiple position ranges for a dimension can be specified by using "," as separators. When using indices, a range of positions can also be used. Ex - "0-2:4-7,11-13:4" implies indices 0,1,2 for the first dimension, 4,5,6,7,11,12,13 for the second, and 4 for the third dimension of the measure. **Note:** Because indices are based on the workbook selections, they may not be consistent across all test cases. Care must be exercised to ensure that you are using the correct indices for your specific test case.	Required
namedkeys	Position names used to get keys of the measure array. Using indices instead of position names, or positions that do not exist in the MeasureStore (Workbook) will be flagged as an error. Similar to "keys" above, but using position names only.	Optional
path	Path of the flat file used for comparison against the Measure array. This looks like pointing to a pre-set environment variable. "path" cannot be used along with "keys" or "namedkeys". "path" is used only when we want to compare the measure data against a pre-existing flat file. If none of "path","keys" and "namedkeys" are specified, the comparison occurs for all the keys in the measure array of the Measure Store (Workbook).	Optional
start	The field start value. If this does not exist in the XML, the value is taken from the measure's pProperties. Used in flat-file comparison.	Optional
width	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in a flat-file comparison.	Optional
value	The expected value of the measure. Comparison on equality.	Optional

Sample usage (indices):

```
<assert-workbook-measure-value-lt name="drwppthrsclsp" keys="0-2"  value="0.0" />
```

Sample usage (names):

```
<assert-workbook-measure-value-lt name="drwppthrsclsp" namedkeys="pos1,pos2,pos3"
value="0.0" />
```

4.3.12 assert-workbook-measure-value-ne Asserts a workbook measure value is not equal to a provided value.

Attribute	Description	Category
keys	The keys to be used to compare measure values. Can be either indices, or position names. Each dimension spec in the keys-spec is de-limited by a ":". Multiple position ranges for a dimension can be specified by using "," as separators.	Required
	When using indices, a range of positions can also be used.	
	Ex - "0-2:4-7,11-13:4" implies indices 0,1,2 for the first dimension, 4,5,6,7,11,12,13 for the second, and 4 for the third dimension of the measure.	
	Note: Because indices are based on the workbook selections, they may not be consistent across all test cases. Care must be exercised to ensure that you are using the correct indices for your specific test case.	
namedkeys	Position names used to get keys of the measure array. Using indices instead of position names, or positions that do not exist in the MeasureStore (Workbook) will be flagged as an error. Similar to "keys" above, but using position names only.	Optional
path	Path of the flat file used for comparison against the Measure array. This looks like pointing to a pre-set environment variable. "path" cannot be used along with "keys" or "namedkeys". "path" is used only to compare the measure data against a pre-existing flat file. If none of "path","keys" and "namedkeys" are specified, the comparison occurs for all the keys in the measure array of the Measure Store (Workbook).	Optional
start	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in a flat-file comparison.	Optional
width	The field start value. If this does not exist in the XML, the value is taken from the measure's properties. Used in flat-file comparison.	Optional

Attribute	Description	Category
value	The expected value of the measure. Comparison on equality.	Optional

Sample usage (indices):

```
<assert-workbook-measure-value-ne name="drwppthrsclsp" keys="0-2"  value="0.0" />
```

Sample usage (names):

```
<assert-workbook-measure-value-ne name="drwppthrsclsp" namedkeys="pos1,pos2,pos3"
value="0.0" />
```

4.3.13 assert-workbook-popcount-eq Asserts a workbook measure popcount matches an expected value.

Attribute	Description	Category
name	Measure name	Required
pop-count	Expected popcount	Required

Sample usage:

```
<assert-workbook-popcount-eq name=" Ipopappenbb" pop-count="100" />
```

4.3.14 assert-workbook-popcount-eq-zero Asserts a workbook measure is not populated.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<assert-workbook-popcount-eq-zero name=" Ipopappenbb"  />
```

4.3.15 assert-measure-popcount-ne-zero Asserts a workbook measure is populated.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<assert-workbook-popcount-ne-zero name=" Ipopappenbb"  />
```

4.4 Workbook Data Edit Operations

This section provides details about workbook data edit operations.

4.4.1 clear-workbook-list Clears a list of workbook measures of all data.

Attribute	Description	Category
path	Path to a file containing a new-line delimited list of measures	Required

Sample usage:

```
<clear-workbook-list path="my_list.txt"/>
```

4.4.2 clear-workbook-measure Clears a workbook measure of all data.

Attribute	Description	Category
name	Measure name	Required

Sample usage:

```
<clear-workbook-measure name=" Ipopappenbb/>
```

4.4.3 edit-workbook-measure Edits the value of a workbook measure based on provided attributes. This tag closely models the action of editing a cell from the Client, as opposed to set-workbook-measure, which does a direct update of the workbook arrays.

This tag can be used in a variety of ways:

Set by loadmeasure file:

By specifying <path>, <start>, and <width>, you can set the measure values according to a supplied loadmeasure file. This file must be in fixed-width format.

Set by positions:

You can set individual positions by providing a <keys> tag. If you want, you can provide the list of positions at an aggregate level by specifying the intersection using the <keyint> tag.

Set everything:

If you do not supply a <path> or <keys> tag, the entire measure will be set to the supplied value.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional

> **Note:** For Date type measures, RPAC supports setting the array cell using either of the following two Date & Time formats - "%m/%d/%Y" or "%m/%d/%Y-%H:%M:%S" where m = 2 digit month, d = 2 digit date, Y = 4 digit year, H = 2 digit hour, M = 2 digit min, and S = 2 digit sec.

Sample usage (using keys):

```
<edit-workbook-measure name="meas1" keys="0-2:5"  value="0.0" />
```

Sample usage (using keyint):

```
<assert-workbook-measure-value-ne name="meas2" namedkeys="chnl1:week1"
value="02/24/2012" />
```

Sample usage (load-measure file):

```
<edit-workbook-measure name="meas1" path="loadMeas1.dat" start="100" width="20"/>
```

Sample usage (set all):

```
<edit-workbook-measure name="meas1" value="2"
```

4.4.4 set-workbook-measure Sets the value of a workbook measure based on provided attributes. Unlike edit-workbook-measure, this performs a direct change of the workbook data and does not update the change arrays unless told to do so. This can have an impact on the calculation cycle since bypassing the change arrays results in a full evaluation rather than incremental one.

This tag can be used in a variety of ways:

Set by loadmeasure file:

By specifying <path>, <start>, and <width>, you can set the measure values according to a supplied loadmeasure file. This file must be in fixed-width format.

Set by positions:

You can set individual positions by providing a <keys> tag. If you want, you can provide the list of positions at an aggregate level by specifying the intersection using the <keyint> tag.

Set everything:

If you do not supply a <path> or <keys> tag, the entire measure will be set to the supplied value.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional
incremental	True if evaluation is to be incremental	Optional

> **Note:** For Date type measures, RPAC supports setting the array cell using either of the following two Date & Time formats - "%m/%d/%Y" or "%m/%d/%Y-%H:%M:%S" where m = 2 digit month, d = 2 digit date, Y = 4 digit year, H = 2 digit hour, M = 2 digit min, and S = 2 digit sec.

Sample usage (using keys):

```
<set-workbook-measure name="meas1" keys="0-2:5"  value="0.0" incremental="true"/>
```

Sample usage (using keyint):

```
<set-workbook-measure-value-ne name="meas2" namedkeys="chnl1:week1"
value="02/24/2012" />
```

Sample usage (load-measure file):

```
<set-workbook-measure name="meas1" path="loadMeas1.dat" start="100" width="20"/>
```

Sample usage (set all):

```
<set-workbook-measure name="meas1" value="2"
```

5. Dynamic Position Maintenance (DPM) Tags

This section provides details about dynamic position maintenance tags. Note that rpac tests for DPM are restricted to cloud deployments.

5.1 Assertions

This section provides details about assertions.

5.1.1 assert-position-label Assert if a position does not exist.

Attribute	Description	Category
levelname	Level name	Required
position	Position name	Required
label	Position label	Required
lang	Language	Required

Sample usage:

```
<assert-position-label levelname="clss" position="4000" label="4000 Casual"
lang="english" />
```

5.1.2 assert-position-not-exists Assert if a position does not exist.

Attribute	Description	Category
levelname	Level name	Required
position	Position name	Required

Sample usage:

```
<assert-position-not-exists levelname="clss" position="4000"/>
```

5.1.3 assert-position-rollup Assert if the given rollup is valid or not.

Attribute	Description	Category
rollup	The expected format for rollup is rollup =childLevel:childPosition, parentLevel:parentPosition	Required

Sample usage:

```
<assert-position-rollup rollup="clss:4000,dept:101" />
```

5.1.4 assert-position-status Assert the status of a position. Status is 1 for formal, 2 for informal, 0 for placeholder, and -1 for deleted.

Attribute	Description	Category
levelname	Level name	Required
position	Position name	Required
status	Status	Required

Sample usage:

```
<assert-position-status levelname="str" position="str_dpm0000020" status="2"/>
```

5.2 Update Informal Position Operation

This section contains details about update informal position operations.

5.2.1 update-informal-position Contains the dpm tags to add or modify dpm positions.

5.2.2 dpmposinfo tag is the child tag of update-informal-position.

Attribute	Description	Category
lang	Language	Required

Sample usage:

```
<dpmposinfo lang="english"> </dpmposinfo>
```

5.2.3 dpmposentry This is the child tag of dpmposinfo.

Attribute	Description	Category
levelname	Level name	Required
position	Position name	Required
isnewpos	Boolean value: True for new position and False for modify.	Required
label	Position label	Optional
oldposition	Old position name	Optional

Attribute	Description	Category
type	Integer value. Can be one of the following	Optional
	0, REGULAR: one single position	
	1, MULTIPLE_PER_PARENT: multiple positions per parent dim	
	2, ONE_PER_CHILD: one position on rollup dim per child -	
	3, ONE_FOR_ALL_CHILDREN: one pos on rollup dim for all children	
	4, GEN_NAME_ONLY: only generating names/labels without applying changes	
count	position count for MULTIPLE_PER_ PARENT type	Optional

Sample usage:

```
<update-informal-positions>
    <dpmposinfo lang="english">
        <dpmposentry levelname="dstr" isnewpos="false" position="801" label="Pos
label" oldposition="801" type="0" count="1"/>
    </dpmposinfo>
</update-informal-positions>
```

5.3 Delete Informal Position Operation

This section provides details about the delete informal position operation.

5.3.1 delete-informal-position This tag deletes an informal position.

Attribute	Description	Category
levelname	Level name	Required
position	Position name	Required

Sample usage:

```
<delete-informal-position levelname="str" position="strdpm00000005"/>
```

6. HSA Tags

This section provides details about HSA tags.

6.1 assert-fact-group

Assert a fact exists or not in the fact group. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
factgroup	Fact group name	Required
factexists	If flag is true, then fact is checked to see if it exists in the fact group.	Required

Sample usage:

```
<assert-fact-group name="UtDyScSrc01U" factgroup="utscalar" factexists="true"/>
```

6.2 assert-fact-table

Assert a fact table exists or not. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
factgroup	Fact group name	Required
exists	If flag is true, then check if the fact table exists.	Required

Sample usage:

```
<assert-fact-table factgroup="utscalar" exists="true"/>
```

6.3 assert-fact-value-eq

Assert a fact value is equal to an expected value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional
precision	The number of decimal places to be considered for assertion	Optional

Sample usage:

```
<assert-fact-value-eqname="UtDyDstLoU"namedkeys="20070121:30000:5"value="10"/>
```

6.4 assert-fact-value-ge

Assert a fact value is greater than or equal to an expected value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional

Attribute	Description	Category
path	Path to load measure type file to compare with	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional
precision	The number of decimal places to be considered for assertion	Optional

Sample usage:

```
<assert-fact-value-ge name="UtDyDstLoU" namedkeys="20070121:30000:5" value="10"/>
```

6.5 assert-fact-value-gt

Assert a fact value is greater than an expected value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional
precision	The number of decimal places to be considered for assertion	Optional

Sample usage:

```
<assert-fact-value-gt name="UtDyDstLoU" namedkeys="20070121:30000:5" value="10"/>
```

6.6 assert-fact-value-le

Assert a fact value is less than or equal to an expected value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional

Attribute	Description	Category
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional
precision	The number of decimal places to be considered for assertion	Optional

Sample usage:

```
<assert-fact-value-le name="UtDyDstLoU" namedkeys="20070121:30000:5" value="10"/>
```

6.7 assert-fact-value-lt

Assert a fact value is less than an expected value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional
width	Width of measure field	Optional
precision	The number of decimal places to be considered for assertion	Optional

Sample usage:

```
<assert-fact-value-lt name="UtDyDstLoU" namedkeys="20070121:30000:5" value="10"/>
```

6.8 assert-fact-value-ne

Assert a fact value is not equal to an expected value. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
value	Expected value	Optional
start	Starting column of measure field	Optional

Attribute	Description	Category
width	Width of measure field	Optional
precision	The number of decimal places to be considered for assertion	Optional

Sample usage:

```
<assert-fact-value-ne name="UtDyDstLoU" namedkeys="20070121:30000:5" value="10"/>
```

6.9 assert-level-contain

Assert a level contains or does not contain a position name. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Level name	Required
positions	List of comma-separated position names	Required
exists	If flag is true, then position names are checked to see if they are contained in the level.	Required

Sample usage:

```
<assert-level-contain name="skup" positions="6200006" exists="T"/>
```

6.10 clear-rdm-status

Clears the RDM status code.

Attribute	Description	Category
status_code	RDM status code	Required
domain_id	Domain ID	Required
info	Information	Required

Sample usage:

```
<clear-rdm-status status_code="1004" domain_id="" info=""/>
```

6.11 set-rdm-status

Sets the RDM status code.

Attribute	Description	Category
status_code	RDM status code	Required
domain_id	Domain ID	Required
info	Information	Required

Sample usage:

```
<set-rdm-status status_code="2001" domain_id="0" info="scls"/>
```

6.12 set-fact-value

Set the value of a fact based on provided attributes.

Attribute	Description	Category
name	Measure name	Required
keys	List of comma-separated position names	Optional
namedkeys	List of comma-separated position names	Optional
path	Path to load measure type file to compare with	Optional
keyint	Intersection string	Optional
value	Value	Optional
start	Starting column of measure field	Optional
Width	Width of measure field	Optional

Sample usage:

```
<set-fact-value name="UtDyDst02U"  namedkeys="w01_2007:30018:5" value="0"/>
```

6.13 reset-seq-generator

Reset a sequence generator to a specific value.

Attribute	Description	Category
name	Name of the generator.	Required
central	Indicates whether the reset is for centralized generator. True/False.	Optional
value	Reset to this value. Any integer >=0. Default 0.	Optional

Sample usage:

```
<reset-seq-generator name="dpm_str"  value="1000"/>
```

6.14 assert-rdm-hier-exists

Assert a hierarchy exists in RDM or not.

Attribute	Description	Category
name	Hierarchy name	Required
exists	If flag is true, then check if hierarchy exist in RDM.	Required

Sample usage:

```
<assert-rdm-hier-exists name="clnd" exists="true"/>
```

6.15 assert-level-exists

Assert a level exists in RDM or not. Note that this tag is restricted to cloud deployments.

Attribute	Description	Category
name	Level name	Required
exists	If flag is true, then check if hierarchy exist in RDM.	Required

Sample usage:

```
<assert-level-exists name="day" exists="true"/>
```

RPAC Restricted Tags for Cloud Deployment Domain

The following tags are currently restricted in the Cloud.

Table 12–2   RPAC Restricted Tags

RPAC Tag	Description
dpmposentry	DPM position Entry
assert-bitsize-value-eq	Assert the dimInfo bitsize or dimregistry bitsize value of a dimension
assert-dim-registry-version	Assert the value for dimension registry version
assert-dimension-contain	Assert a dimension contains or does not contain a position name
assert-dimension-exists	Assert a dimension name exists
assert-dimension-size	Assert a dimension size is between a minimum and a maximum value
assert-position-label	Asserts the Updated DPM label
assert-position-not-exists	Asserts the position does not exist
assert-position-rollup	Asserts the roll up for DPM data
assert-position-status	Asserts the status of DPM data
assert-reindex-in-progress	Asserts the status of reindex with the user input
assert-reindex-required	Assert whether Diminfo Bitsize is equal to DimRegistry Bitsize
assert-reindex-threshold	Assert the Threshold value of the dimension
assert-rpas-user-eq	Assert the user is equal to an expected value
assert-rpas-user-ge	Assert the user is greater than or equal to an expected value
assert-rpas-user-gt	Assert the user is greater than an expected value
assert-rpas-user-le	Assert the user is less than or equal to an expected value
assert-rpas-user-lt	Assert the user is less than an expected value
assert-rpas-user-ne	Assert the user is not equal to an expected value
backup-domain-measures	Backup a list of measures
copy-domain	Does a simple copy of a domain to a new location.
delete-informal-position	Deletes the DPM data
dim-registry-version	Save the current dimension registry version

Table 12–2   (Cont.) RPAC Restricted Tags

RPAC Tag	Description
dump-domain-measure	Dump a measure to a text file.
load-domain-measure	Load a measure from a text file.
process-stop	Stop RPAC created process
process-wait	Wait for RPAC created process to exit
reset-seq-generator	Reset a sequence generator to a specific value.
restore-domain-measures	Restore a list of measures
rpasce_run	Start a rpasce run script with the argsments provided as the value of this tag
shell	Start a shell process with the command provided as the value of this tag
update-informal-positions	Updates the informal positions
dpmposinfo	DPM position info
update-dpm	Updates the DPM data

13

Additional Configuration

This chapter provides details about additional configuration for RPASCE.

- Notifications
- Images and Contextual Help
- Chart Data Points

Notifications

A notification app code is assigned to the RPASCE Client application during installation. In the 17.0 version, this is hard-coded to MFP. Starting with 17.1, the installer user can select the notification app code.

Selecting the Notification App Code

During RPASCE Client installation, starting with 17.1, you must choose a notification app code. It is typically the code corresponding to the RPAS application that you are installing. Table 13–1 lists some typical RPAS applications.

Table 13–1   Typical RPAS Applications

Application Name	Application Code
MFP(Retail) Cloud Service	MFPRCS
MFP(Cost) Cloud Service	MFPCCS
MFP Enterprise Edition Cloud Service	MFPEECS
Item Planning Cloud Service	IPCS
Assortment Planning and Optimization(Fashion) Cloud Service	APOFSLCS
Assortment Planning Enterprise Edition Cloud Service	APEECS

Whichever option you pick, the corresponding app code is inserted into a property in <RPASCE-Client-Config-Dir>/config.properties.

For example if you select MFP(Retail) Cloud Service, then you have the following property setting:

platform.appcode=MFPRCS

> **Note:**   This is a 17.1.* property. The 17.0.* versions do not have this property (since it is hard-coded to MFP and is not configurable).

LDIP Setup

This section describes the LDIP setup process.

Change directory to $LDIP_HOME/config. Open the notif.properties file. Edit it as follows:

hostname of server where application is deployed. It matches the client install

property "input.appserver.host"

host=burrg41101v

port number of WebLogic Server. It matches client install property "input.admin.server.port".

port=17000

Leave this as is

userAlias=notif_user

the notification app code. Take it from the platform.appcode config property mentioned above.

appName=MFPRCS

Set Up the Credential Store Wallet

Notifications are sent by LDIP/RPAS using web service calls, which require authentication. The corresponding userid and password are stored in a wallet file.

The userid and password are those of any user who exists in the identity store used by the RPASCE application. If you are using SSO, then the user is present in the LDAP directory inside the SSO infrastructure. If you are directly logging into the application using BASIC authentication, then the user is present in the WebLogic embedded LDAP directory.

> **Note:** This user does not have to be in RPAS (but can be).

1. Use the following command to create the wallet. (First delete the $LDIP_HOME/wallet folder that came with the server install).

   java com.oracle.retail.integration.common.security.credential.Main -a dev_user -u <userid> -l $LDIP_HOME/wallet

2. You will be prompted to enter the password.

   For example:

   java com.oracle.retail.integration.common.security.credential.Main -a dev_user -u adm -l $LDIP_HOME/wallet

   Enter password: Admin001

The notifications setup is complete, and every user should be able to log into the RPASCE application and see his or her notifications displayed in the UI.

Purging Old Notifications

Over time, the notifications can accumulate. Notifications older than a certain number of days may no longer be of interest. Users can delete their own old notifications from the UI, using the Notifications table. Alternatively, all notifications older than a certain

number of days can be deleted. This requires executing a PL/SQL function in the notification schema called RAF_NOTIFICATION_TASK_PKG.DEL_NOTIF_PAST_ RETENTION.

To invoke this, connect to the notification database schema using any SQL client, such as SQL Developer or sqlplus. Then, execute an anonymous PL/SQL block. For example:

declare

  v_result varchar2(32000);

  v_num NUMBER;

begin

  v_num := RAF_NOTIFICATION_TASK_PKG.DEL_NOTIF_PAST_RETENTION(v_ result, 'MFPRCS',57);

  dbms_output.put_line(v_num);

  dbms_output.put_line('Message : ' || v_result);

end;

The above script deletes all notifications older than 57 days for the notification app code MRPRCS. (In 17.0 it is MFP).

To explore the notifications from the database perspective, start with the RAF_ NOTIFICATIONS table. Each notification takes up a single row. The recipients are stored in the RAF_NOTIFICATION_RECIPIENTS table.

Images and Contextual Help

Valid Hosts: Users can specify a web URL as the source of an image(s) (typically the content management server) or source for contextual help. Due to security reasons, the RPASCE client will only use images or contextual help links from these configured hosts. Users are only allowed to enter URLs from the hosts defined by this setting in rpasConfig.properties:

```
#List of valid hostnames for images and contextual help
valid.url.hosts=xyz.com
```

These should be separated by commas. Enter only hostname and not the http protocol. For example, xyz.com. Note that the hostname is not case sensitive. The dimension for which the images are to be displayed should have a dimension attribute measure registered which of type string and has "UI Type" property set to media. For example consider the name of the dimension attribute measure is addvskupimgt for dimension skup. The measure should be loaded with the content similar to below. Each record in the input file is a position id followed by the image URL wrapped in <image> tag. The example input file below is in the csv format and hence named addvskupimgt.csv.ovr.

```
   70063172,"<image id=""main"" label=""Front""><url
size=""thumb"">http://127.0.0.1:7001/contentserver/imgfetch/1000396_Front_
full.jpg</url></image>"
   73137693,"<image id=""main"" label=""Front""><url
size=""thumb"">http://127.0.0.1:7001/contentserver/imgfetch/CardiganLS.jpg</url></
image>"
   96880516,"<image id=""main"" label=""Front""><url
size=""thumb"">http://127.0.0.1:7001/contentserver/imgfetch/1000399_Front_
full.jpg</url></image>"
   88805005,"<image id=""main"" label=""Front""><url
size=""thumb"">http://127.0.0.1:7001/contentserver/imgfetch/1000378_Front_
```

```
full.jpg</url></image>"
   72939751,"<image id=""main"" label=""Front""><url
size=""thumb"">http://127.0.0.1:7001/contentserver/imgfetch/1000439_Front_
full.jpg</url></image>"
   96666402,"<image id=""main"" label=""Front""><url
size=""thumb"">http://127.0.0.1:7001/contentserver/imgfetch/1000440_Back_
full.jpg</url></image>"
```

The use of the double double quotes in the lines above is the standard way of escaping double quotes in any RPASCE measure's csv input files.

Chart Data Points

Users can specify the maximum number of data points on the chart representation of data by setting them in rpasConfig.properties. Note that setting the maximum values higher will degrade charting performance.

charts.maxSeries: Enter the maximum number of series. The default value is 100.

charts.maxGroups: Enter the maximum number of groups. The default value is 100.

14

Performance Diagnostic Tool

This chapter describes the RPASCE Performance Diagnostic Tool.

Overview

The Performance Diagnostic Tool is a set of administrative reports that an application administrator or system integrator can use to monitor, identify, and review any RPASCE performance issues. In this way, the system administrator or system integrator has access to the information necessary to take action. The Performance Diagnostic Tool Template is accessed in the System Administration Tasks tab, and is available to administration users. It provides details about potential performance trouble areas, focusing on common workbook and batch operations, including workbook open, commit, and calculations. The data can be viewed at aggregate levels, or down at the user level, if an administrator must work with a specific end user who has expressed concerns regarding RPASCE system performance. For the batch, it provides visibility into the more common batch operations that require a long time to complete, such as measure and hierarchy load and rule groups calculated during the batch. Based on the information provided in the Performance Diagnostic Tool, the administration user has a clearer understanding of where the issue is, which can help the user address the issue.

Collecting and Managing Performance Metrics

The Performance Diagnostic Tool monitors and analyzes two kinds of performance metrics: operational performance data and system metrics. RPASCE provides various means to collect the required data.

Operational performance data includes elapsed time for various workbook and batch operations. Workbook operations include Workbook Build, Open, Refresh, Commit, Custom Menu, and Calc. For Workbook Build, Refresh, Commit, and Custom Menu, elapsed time is also collected for each expression. Batch Operations include measure load, hierarchy load, and evaluating rule groups using the mace service.

RPASCE collects operation performance data in real time with minimal impact. The collection of operational performance data is built into the RPASCE core and does not need to be turned on or off. Data collection is available.

System metrics collected include Dimension Size, Measure Size, Workbook Size, and the Domain's Disk Usage. RPASCE collects the system metrics using a set of batch scripts provided as part of the release. The administrator can set up repeating jobs to execute these scripts on a daily or weekly basis. These scripts generate data files that are named: YYYYMMDD[metric].csv, where YYYYMMDD is the timestamp such as

20180101 and metric is the metric name, such as workbook_size, item_size, measure_size, and dimension_size. The data is stored as plain text in .csv format.

For support, both operational performance data and system metrics are automatically copied during the domain cloning or the copying process. This process does not need to be turn on or off. Since these files are text files without outside dependency, to preserve disk space, it is recommended that batch jobs be set up on the server to remove files older than a pre-defined number of days, such as 15 days. The system only retains the performance data for the last 15 days to preserve disk space. However, if any performance issues are observed for a particular period of time, it is recommended that the relevant performance data be preserved for analysis.

Analyzing Workbook Performance

The Performance Diagnostic Tool is available to administration users as a workbook template, located in the template group System Administration, available from the task flow menu under Administration. To access the template, the user must login as an administration user, go to the System Administration Task Flow, and select the Performance Diagnostic Tool, shown in Figure 14–1.

Figure 14–1 Select Performance Diagnostic Tool
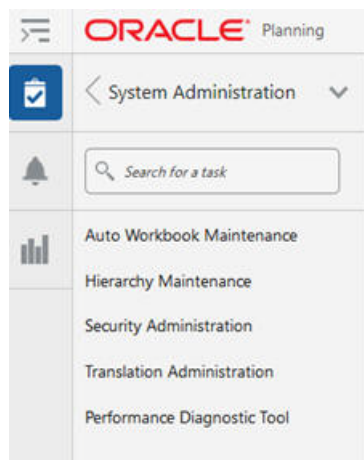


The Segment dialog box, which shows all previously defined segments, is displayed, as shown in Figure 14–2. This is used to select a predefined segment in order to modify it or copy it.

Figure 14–2 Segment Dialog Box
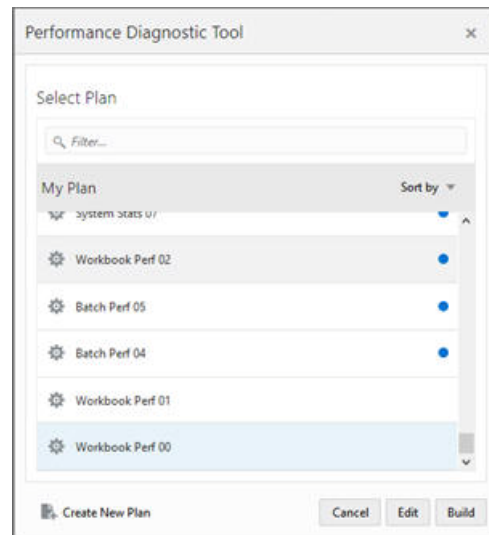
To define a new segment, select **Create New Plan**.The Enter Plan Label dialog box is displayed. Enter a label and click **OK**.

Figure 14–3 Enter Plan Label
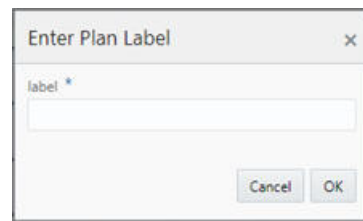
Then, select users from the list of users in the Levels and Positions tab shown in Figure 14–4. This contents of this list depends on whether or not performance data exists for this user in the domain. The administration user must pick at least one user to examine the performance data.

Figure 14–4 Levels and Positions
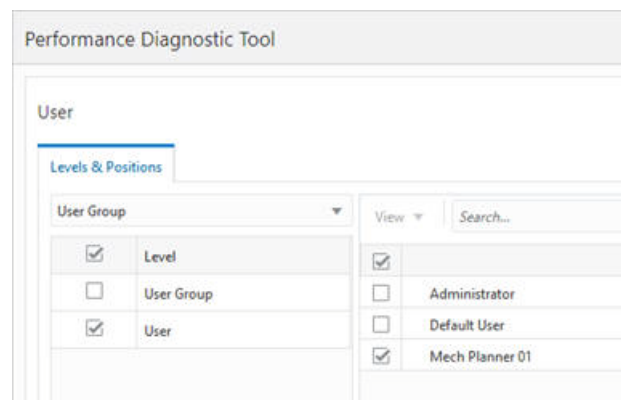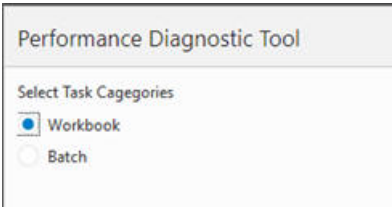

Select the calendar range, as shown in Figure 14–5. The calendar range is defined at the lowest calendar level of the domain, usually the day level. The days are arranged by the availability of performance data for the previously selected user.

Figure 14–5 Calendar Range
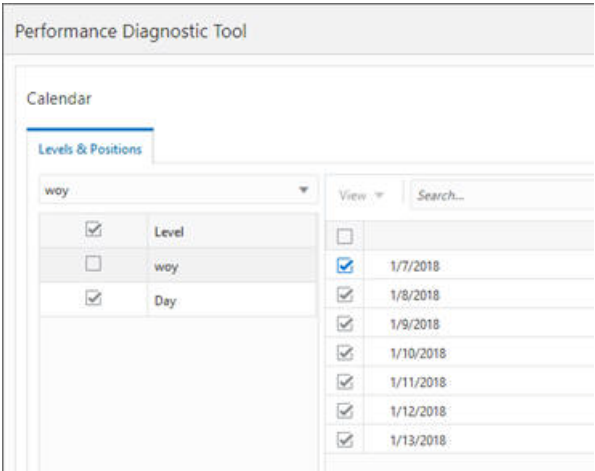
Select either Workbook or Batch, as shown in Figure 14–6.

Figure 14–6 Select Workbook or Batch

The segment, once fully specified, it is submitted to the RPASCE Task Daemon to build the workbook. The status of the workbook build can be tracked using the Task Status Dashboard or the Notification tab. When the workbook is being built, the performance data is loaded using a set of RPAS rules that create the customized hierarchies for examining the performance data and the actual performance measure that holds the performance data. Once the workbook is successfully built, it can be opened via the Notification tab or from the segment window. By default, the workbook opens as shown in Figure 14–7.

Figure 14–7   Operations High Level View
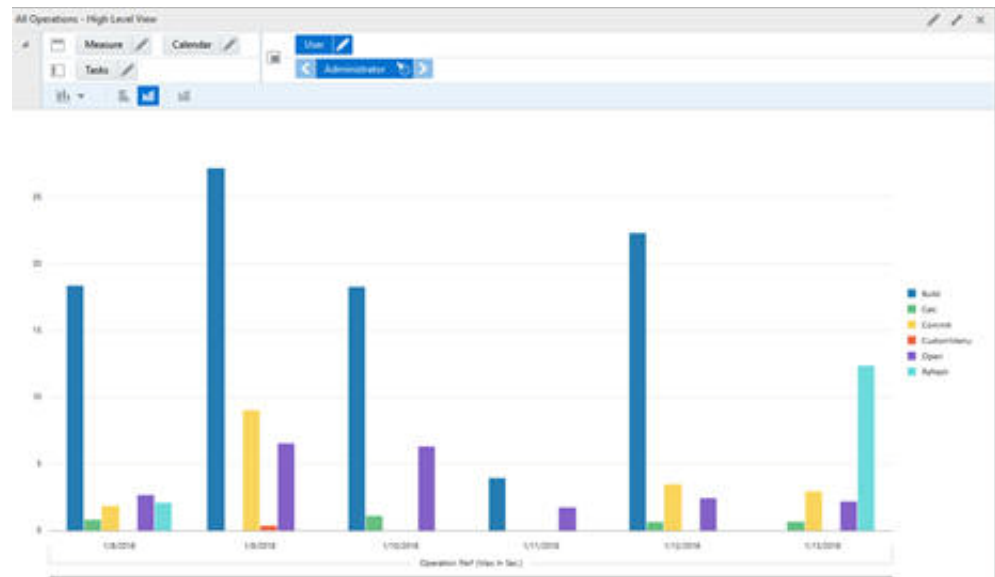
By default, this view displays the Maximum Running Time, in seconds, for different types of Workspace operations such as Build and Calc, along with the calendar dimension, in a bar chart. The maximum running time is shown at aggregated level. For example, the maximum build time for a particular day is the maximum build time for all the workbooks that the specific user built on that day. Using this view, administration users can identify the type of workbook operations that require the longest time and how that time changes from day to day.

Since the data is presented at a high level in this view, the administration user can use this as a starting point for an investigation to identify which workbook build is actually taking the time. For more detailed information, the administration user can view the more detailed All Operations view, for specific day/user combination, as shown in Figure 14–8.

Figure 14–8   All Operations View
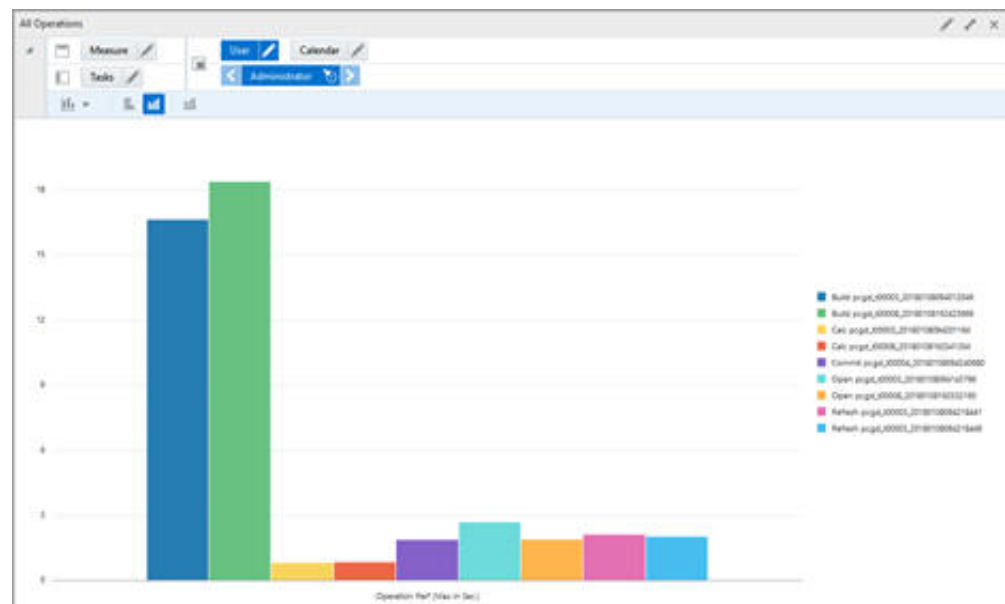
The All Operations bar chart shows the various workbook operations performed by the specific day/user combination, with the detailed workbook name.

Using the Performance Diagnostic Tool, an administration user can drill down to the execution time at the expression level. However, only the expression execution times for full mode evaluation (Workspace build, Commit, Refresh, and Custom Menu) are captured. The expression execution time for incremental mode evaluation by Workbook Calc operations are not captured, as the individual expression's incremental evaluation time is generally at the level of milliseconds. To examine the expression level performance data, the user can examine the All Operations - Expressions view, shown in Figure 14–9.

Figure 14–9   All Operations Expression
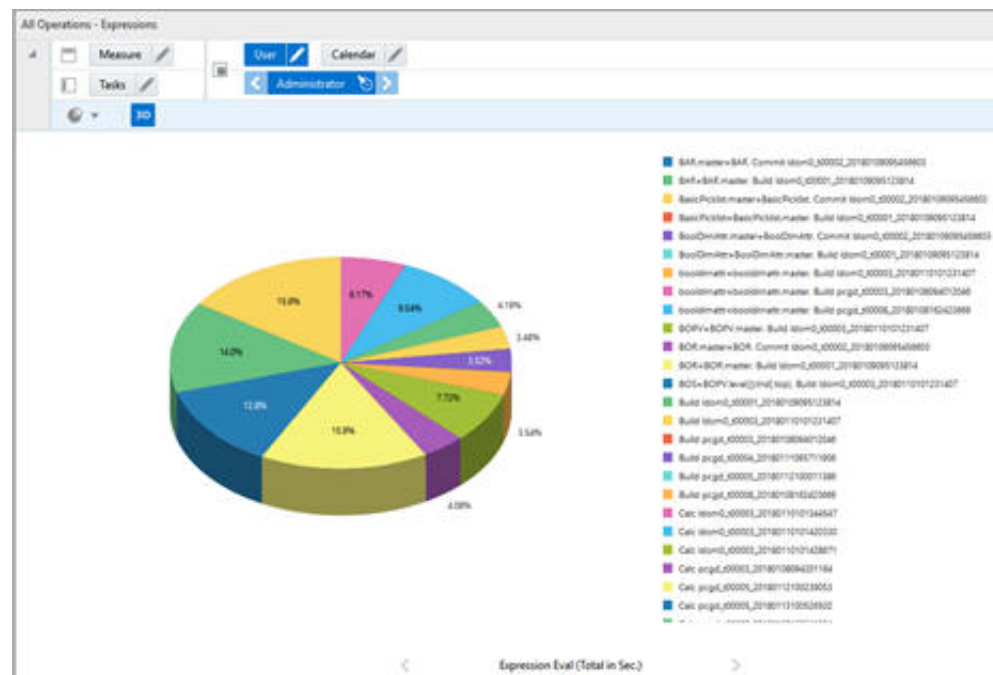
The expression level, by default, displays each expression's execution time for the user/day combination in a pie chart. This reveals the longest running expressions for a specific workbook operation. Mousing over the pie chart shows the actual expression for the slice.

From the execution time at the expression level to the overall maximum running time for the type of workbook operations, the level of granularity is controlled by a customized hierarchy called Tasks. The Tasks hierarchy is built on demand when the workbook is created, using the data in the performance log files. The user can arrange the Tasks hierarchy to drill down to operations belonging to a specific workbook template or to a specific workbook. The user can arrange and manipulate the Tasks hierarchy using the Tasks tab, via the Tasks Hierarchy dialog box shown inFigure 14–10.

Figure 14–10   Task Hierarchy
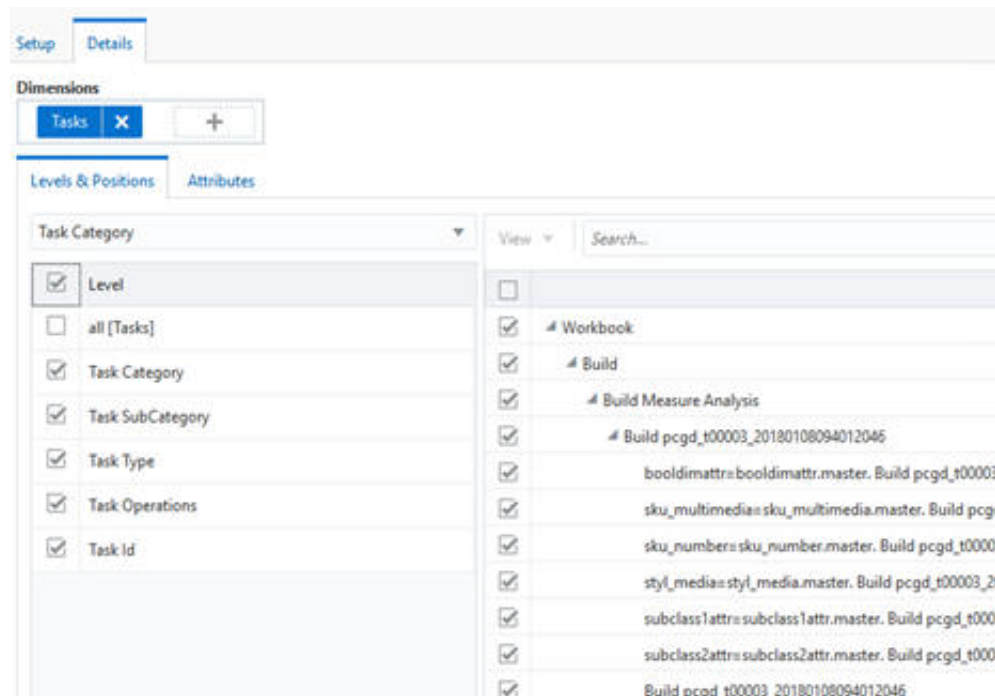
The Task hierarchy is made up of five levels.

- The lowest level is Task ID, which is used by the Expression level view. Each Task ID represents an individual Expression evaluation. Task ID aggregates to the Task Operation level, which maps to each specific workbook operation, such as Build Operation for workwork pcgd_t00003.

- The Task Operations level is the level shown in the All Operations view. Task Operations aggregates to the Task Type level, which maps to a specific workbook operation type/template name combination (for example, Build Operation for Measure Analysis template).

- The Task Type level aggregates to the Task Subcategory level, which is the level shown in the All Operations - High Level view.

- The Task Subcategory maps to each of the workbook operations such as Build, Calc, and Refresh.

- The top level of the Task Hierarchy is the Task Category, which has only two positions: Workbook and Batch. Only one of these can be shown for each Performance Diagnostic workbook. The level names in the Task Hierarchy are generic, such as Categories, IDs, or Types, because the hierarchy definition is shared between the Workbook and Batch operations.

Analyzing Batch Performance Data

To view performance data, the user selects the Batch Operation choice in the wizard. RPASCE collects batch performance data for hierarchy and measure loads and mace rule group evaluation through the mace service (rpasce_run mace). When the workbook is opened, the set of views available are the same, but the levels in the task hierarchy map to different batch operations. The All Operations - High Level View is shown in Figure 14–11.

*Figure 14–11   All Operations High Level View*
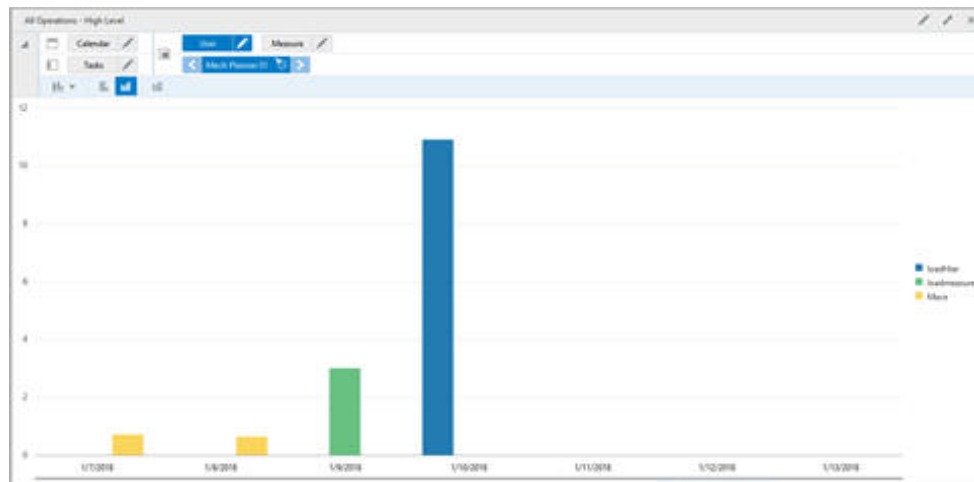
The High Level view shows the maximum running time for each type of batch operations for calendar. In this example, only one type of batch has been executed for each day: the mace service for the first two days, then the execution of load measure, and loadHier on the fourth day.

All Operations by default displays the maximum running time at the Task Operations level. For the mace service, the Task Operations displays the individual mace run for each local domain.

Figure 14–12   Mace Service
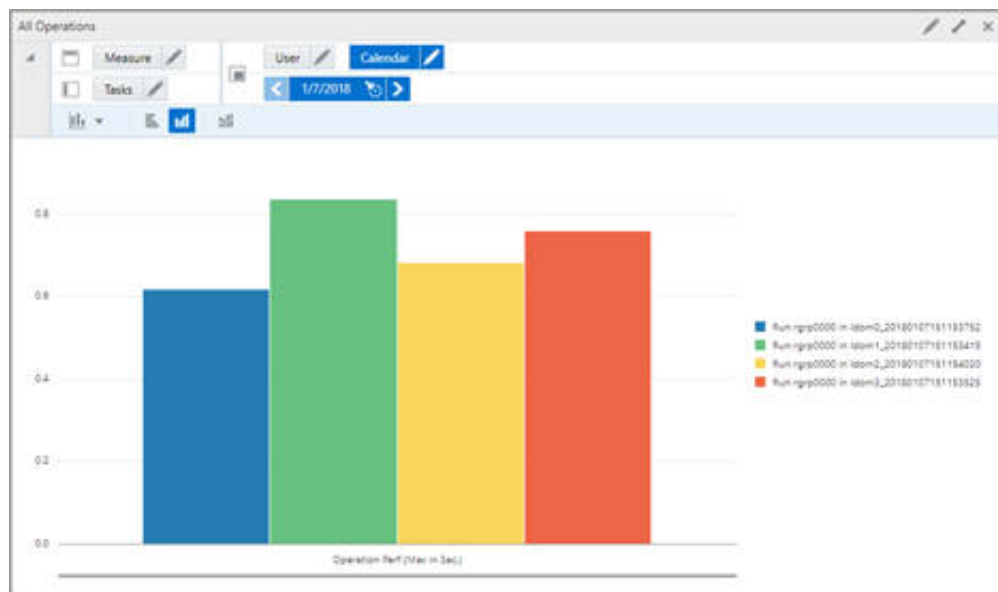
For the Expression Level view, similarly to Workbook Operations, the execution time for each expression is displayed, as shown in Figure 14–13. Data is only available for the mace service that evaluates the rule group, as there are no expressions to evaluate for load measure or load hierarchy data.

Figure 14–13   All Operations Expressions
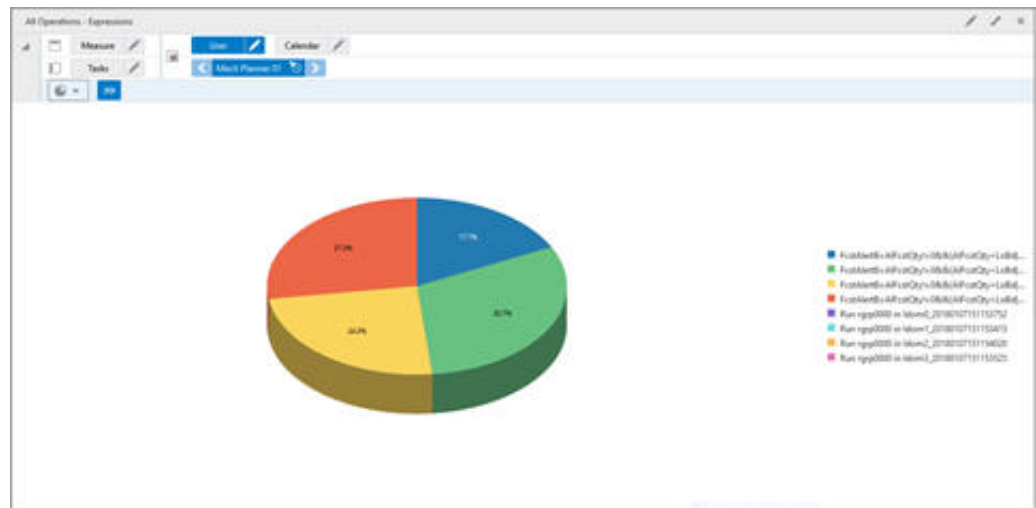
Compared to the Task Hierarchy for the Batch Operation views, the Hierarchy dialog box displays the same five Task Levels, but with a different meaning. For the mace run rule group, the Task Operations level displays the rule group evaluation in one local domain, identified by a time stamp. This helps to differentiate the same rule group being evaluated multiple times in the same day. The Task Type displays the Mace and Rule Group name being evaluated. The Task Subcategory displays the mace command. And the Task Category is batch.

Figure 14–14   Batch
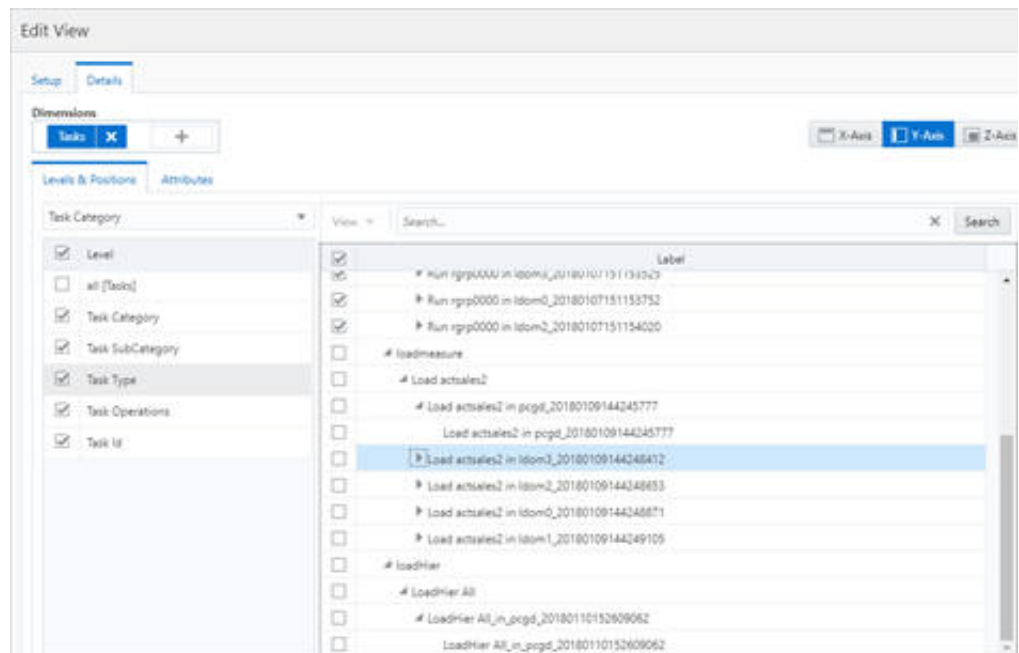
Figure 14–15   Load Measure
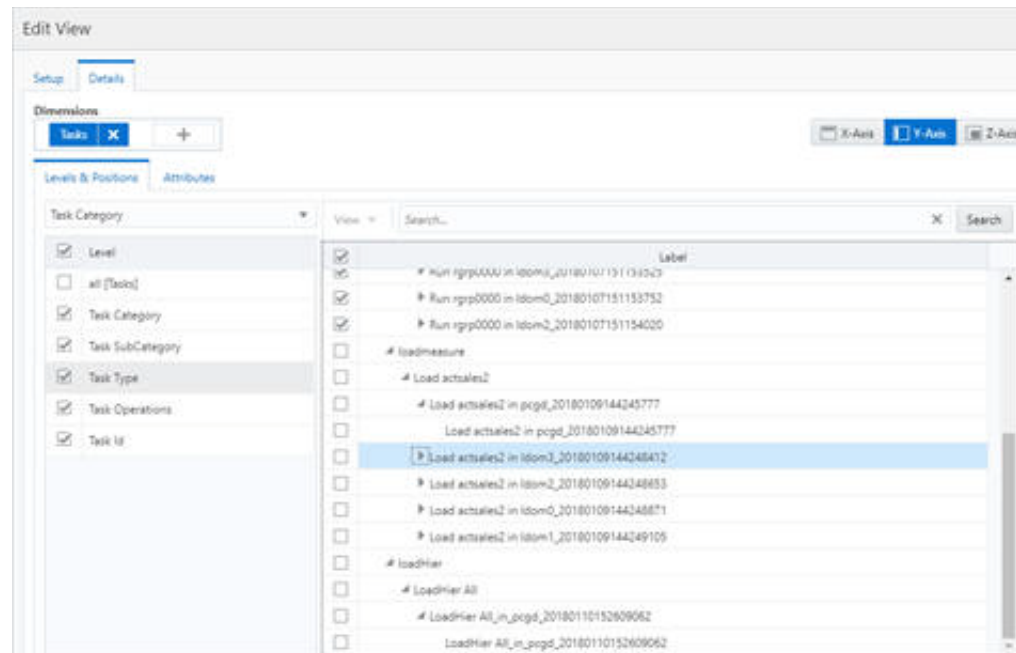
Neither loadHier and loadMeasure have any expression to evaluate, so the positions on the Task ID level are duplicates of the Task Operations, as RPASCE does not allow positions at an aggregated level without a position at the base level.

For loadHier:

- The Task Operations display the loadHier call in the master domain, the hierarchy name(s) being loaded, All in the example to signify that this is a loadHier -loadAll call, and the time stamp.

- The Task Type is loadHier and the hierarchies being loaded, no timestamp. So if the user made multiple loadHier calls for the same set of hierarchies, there will be multiple Task Operations, one for each loadHier call, but only one Task Type position. If the user made two loadHier calls for the same day, two Task Types will be displayed for that day, one for the loadHier product and another for the loadHier location. Each Task Type also has a corresponding Task Operation with a timestamp.

- The Task SubCategory is loadHier.

- The Task Category is batch.

For loadMeasure, similarly to loadHier:

- The Task ID has same positions as Task Operation.

- Each loadMeasure call for a specific list of measures, plus a timestamp, constitutes a unique Task Operation.

- The Task Type is load Measure and the loaded measure list combination.

- The Task SubCategory is simply the LoadMeasure utility.

- The Task Category is batch.

Figure 14–16 Task ID
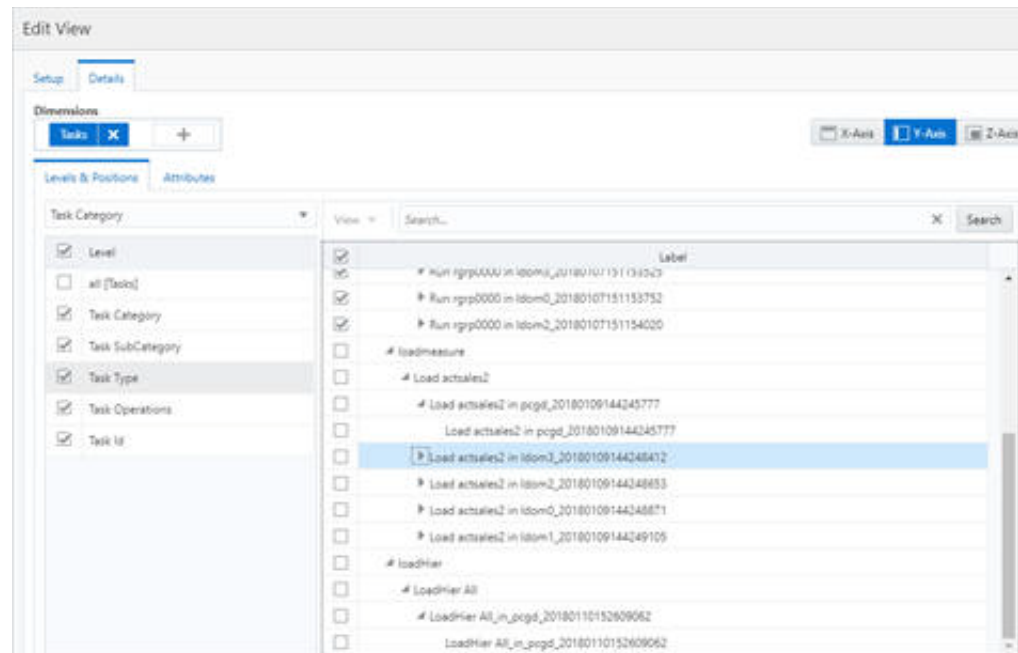
Analyzing Disk Usage

Using the Operations view, administration users can investigate performance issues down to the expression or utility level. A slow utility run or expression evaluation is usually linked to disk usage or to a large population count of measures. System metrics are always included in the Performance Diagnostic Tool workbook, regardless of whether the user selects Batch or Workbook Operations to build the workbook.

Administration users can use the Disk Usage view to analyze the overall disk utilization by RPASCE. By default, the Disk Usage view displays the total disk usage by MB for each sub-domain, the master domain, and the rpas home directory, along calendar dimension.

Figure 14–17   Disk Usage
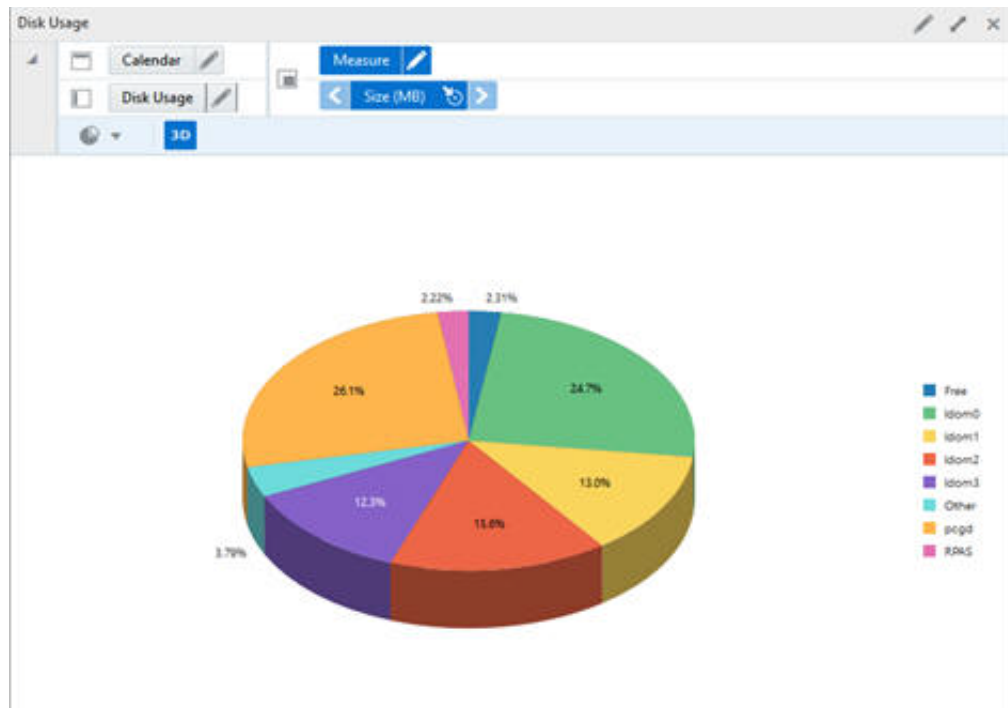
The disk usage is organized by the customized Disk Usage hierarchy. The user can explore and arrange the positions in this hierarchy via the Disk Usage tab, which displays the hierarchy dialog box.

Figure 14–18   Disk Usage Hierarchy
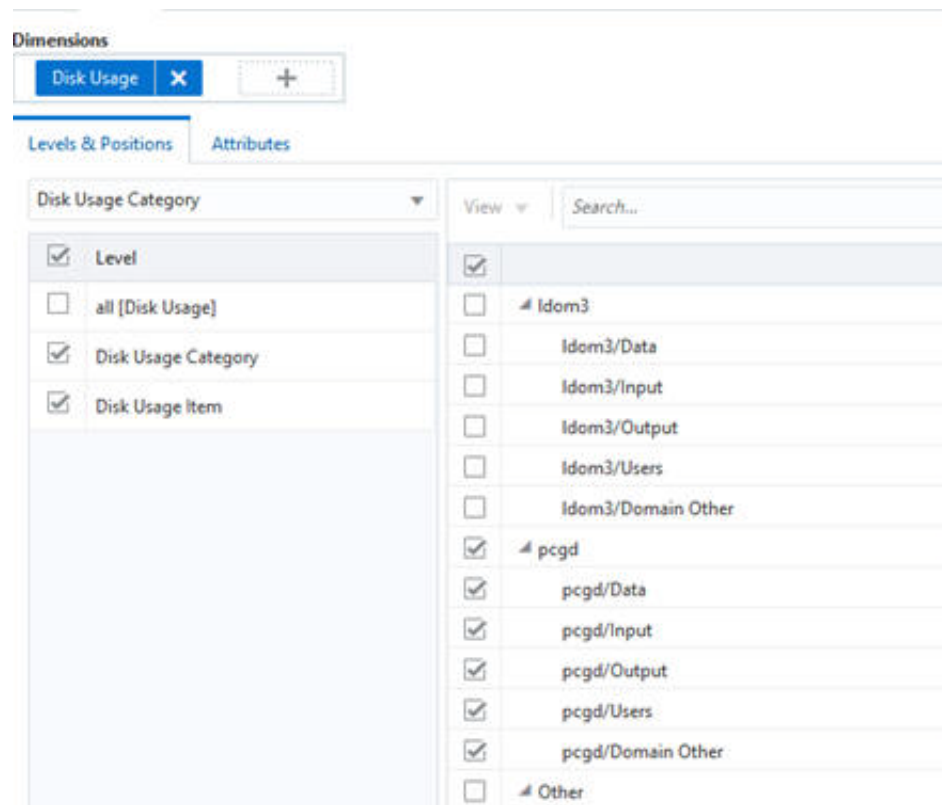
The Disk Usage Category level displays the domain level information. Each domain maps to a position at the Disk Usage Category level. The Disk Usage Item level maps to each sub-directory per domain. The user can view a particular domain and examine how the disk usage is distributed in the directories of that domain. Figure 14–19 displays the master domain pcgd's usage.

Figure 14–19 Disk Usage Category
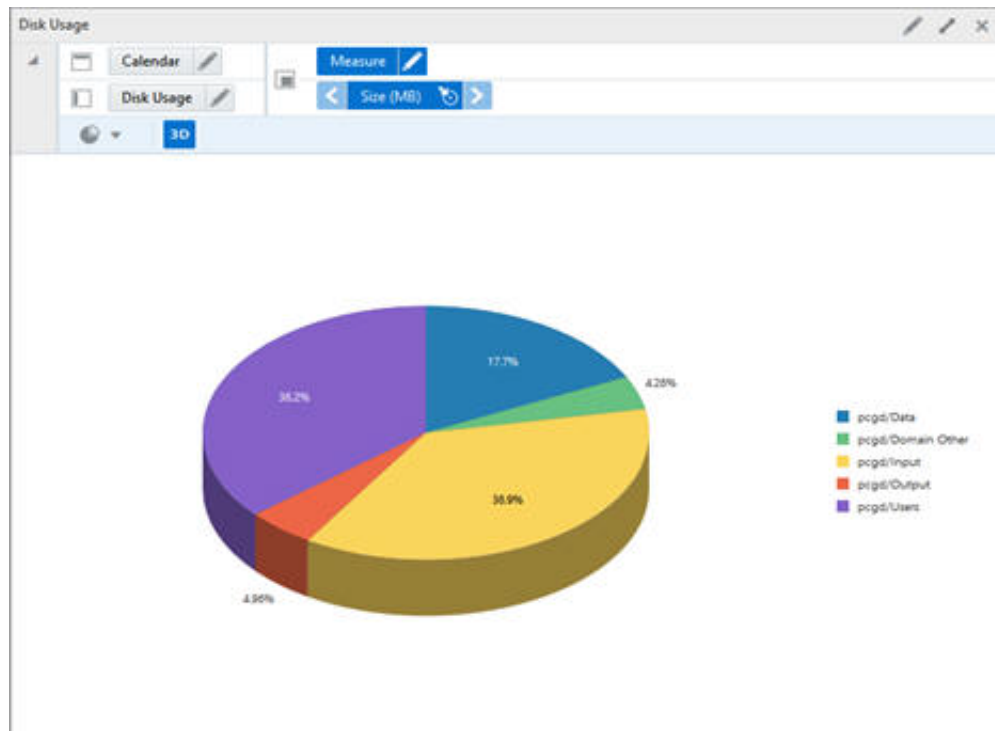
A high level of disk usage in the Users directory suggests a large number of saved workbooks. A large input directory may indicate that some large measure load files are remaining.

The user can also view how the disk usage changes over the calendar dimension by positions along the calendar. The change in disk usage over the calendar may be a result of data load, batch calculations run over night, or large workbook build/commits that occur during the day.

Analyzing Workbook Usage

The Workbook Size view displays the Workbook Size by MB, identified by the segment label. The default view is shown in Figure 14–20. Here, the user can identify the largest workbooks in each domain.

Figure 14–20   Workbook Size
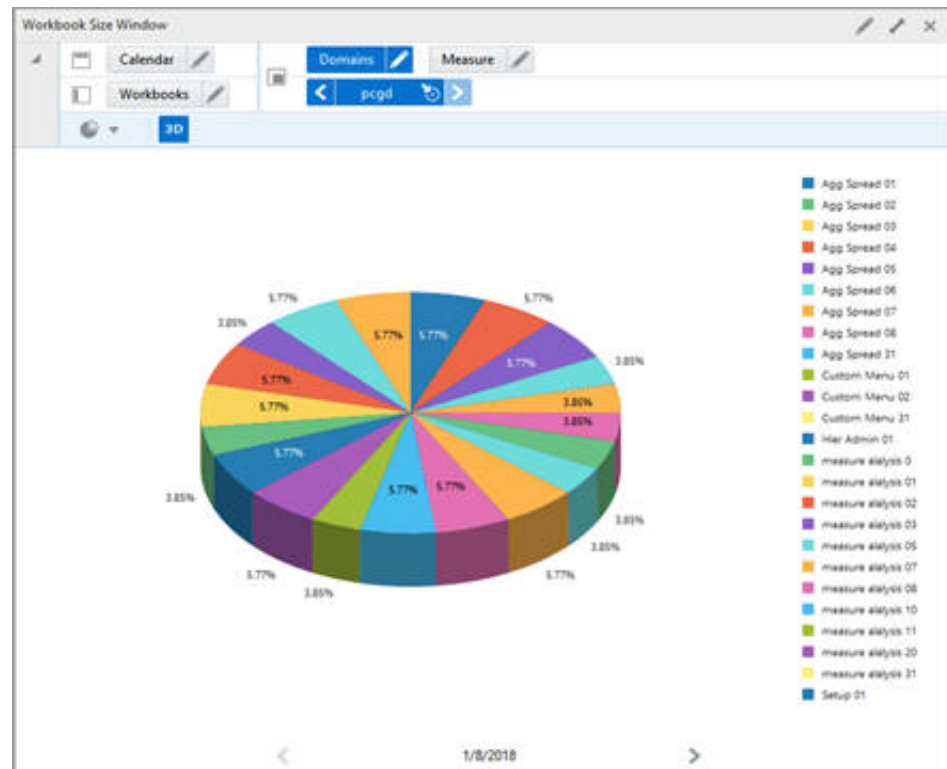
Workbook sizes are organized by the customized hierarchy Workbooks, which can be explored using the Hierarchy dialog box.

Figure 14–21   Workbook Size Hierarchy
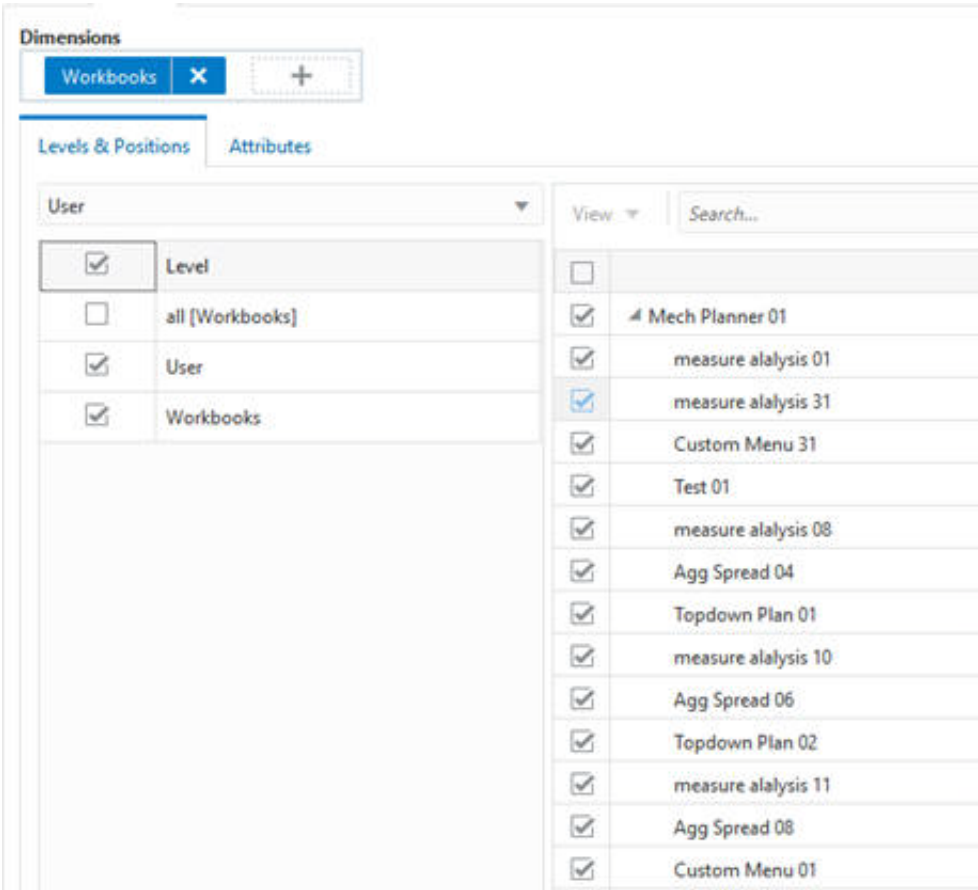
The customized Workbooks hierarchy has only two levels. The root level is the workbook itself identified by the segment label, and it rolls up to each user. The user can collapse the level to display the total workbook usage by each user within each domain.

Since the workbook size view is defined on the Calendar dimension, the user can roll over calendars to see how the workbook size changes, which may be caused by calculations that occurred in the workbook.

Analyze Measure Size

Measure Size, especially the populated size, is an important performance factor behind many operations, such as batch calculation and measure load. The user can explore both the logical and the populated size of measures using the Measure Size view. The Measure Size view displays the populated size across the global domain as shown in Figure 14–22, so that the user can identify the largest populated measures in the entire domain.

Figure 14–22   Measure Size
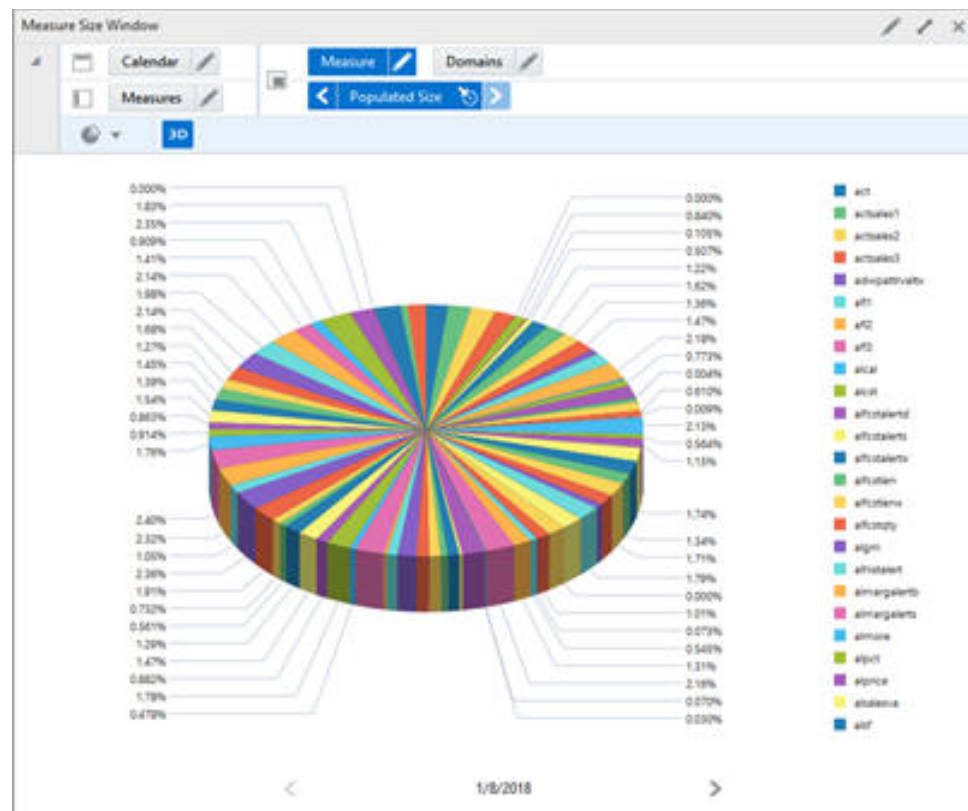
The user can explore the Domains dimension to find the largest measures within each sub-domain, or the master domain itself if the measure is HBI.

Figure 14–23   Domain Dimension
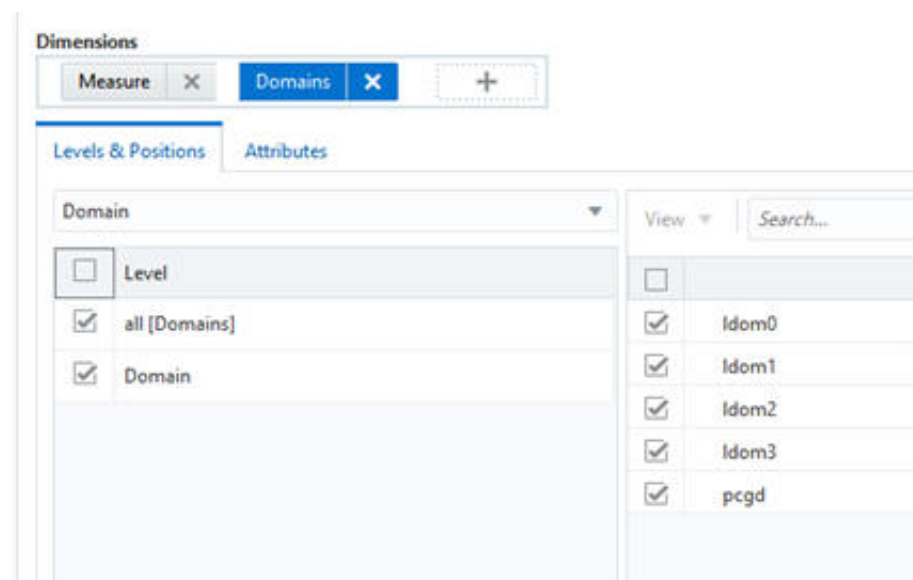
The user can also explore how the measure size changes over the calendar dimension by viewing the days. The measure size changes over the calendar either because of large data load during the batch run or due to the calculation or workbook commits.

Analyzing Dimension Size

Dimension size affects a measure's logical size. A large dimension may also affect to performance in loadHier or a batch calculation where the NOB (number of observation) iteration must be performed instead of the POP iteration. The user can use the Dimension Size view to explore the Dimension Size within the domain. By default, the Dimension Size Window displays the actual dimension size along the calendar dimension, so the user can identify the largest dimension and how its size changes over time, which is usually caused by a loadhier run.

Figure 14–24   Dimension Size

A
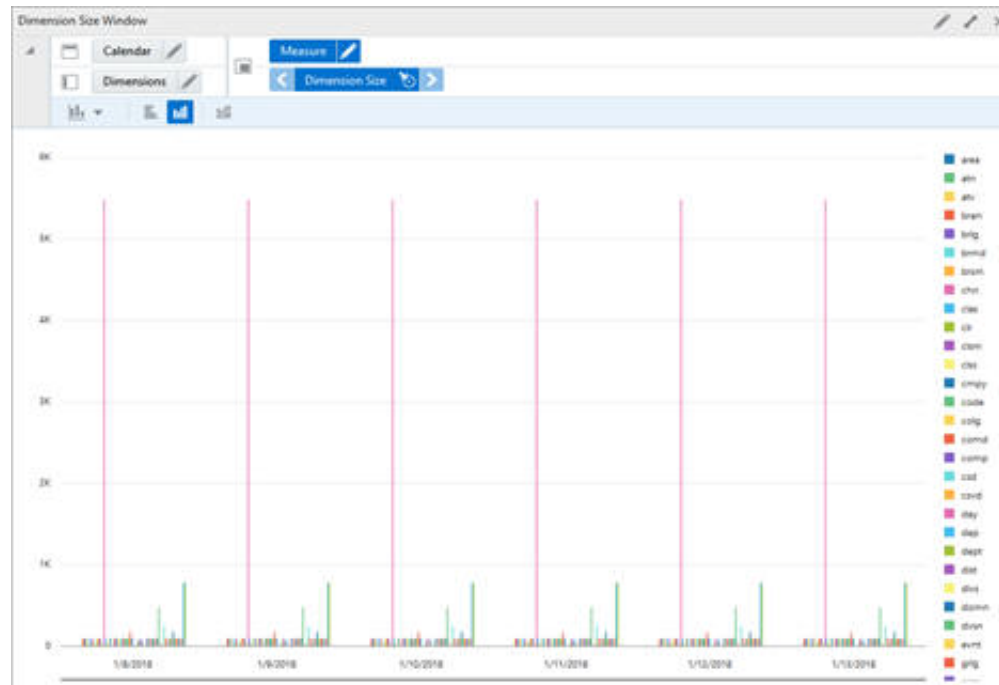
Appendix: RPASCE Client Logging Configuration

Logging in RPASCE Client application can be configured in the log4jconfig.xml. This file is located under the <RPASCE Client Install directory>/config folder. The configuration file provides the user with various levels of configuration options. Logging user actions generates information that is useful in troubleshooting application errors. Such logs may be requested by the RPASCE development group. It is not recommended to enable detailed logging by default.

Note the following:

- Logging can be configured at different levels and modes.

- Logging can be configured at different layers of the application. Different layers in the application can have different logging levels, as defined by a user's requirements.

Log Files

Log files can be specified by the customer. There are two default log files, both located in the log file path defined during the RPASCE client installation. It is <RPASCE Client install directory>/log by default. The main logging information goes to `Edge-<wls-server-name>.log`. The main log file is intended to capture data relating to user actions of significance. For debugging performance issues, another log file can be specified, called `perf-<wls-server-name>.log`. The name and location of each of these files is configurable.

You can change the file name by changing the value for the property in log4jconfig.xml, as follows:

```
<RollingFile name="EdgeAppender"
fileName="${log-path}/Edge-${wls-server-name}.log"
filePattern="${log-path}/Edge-${wls-server-name}-%d{yyyy-MM-dd}-%i.log" >
```

By default, the log file is a daily log. A new file is created each day. You can change the filePattern to yyyy-MM-dd-HH if hourly log files are desired. The use of the rolling file appender keeps the size of log files to a reasonable value. The file names are named <base-log-file-name>.<date>.

Log Levels

The RPASCE Client application supports different logging levels as part of the configuration. You can configure the log levels based on your needs.

Different levels and modes of logging in the application are listed in Table A–1 in order of severity. The warn level includes error, the info level includes warn and error, and so on.

Table A–1   Log Level Definitions

Log Level Name	Log Level Definition
error	Used to log errors. This is the default level that is set after installation. Critical errors and information are logged at this level.
warn	Used to show messages regarding potentially harmful situations. These situations do not affect the running of the RPASCE Client application but may not give the desired result. For this reason, it is preferable to warn the user about such situations.
info	Used to provide general information to the user (for example, if the login was successful or not). This level is used sparingly to provide information about main events. As mentioned in the *Oracle Retail Predictive Application Server Security Guide*, it is recommended that you set the security category to no lower than the info level, so that the users logging in and out are noted in the logs.
debug	Used for logging in debug mode. In this mode, much more logging occurs than in other modes. Debug information that can help in debugging is provided at this level. This must be used only for debugging purposes, as the number of logs generated may cause performance issues. When a customer wishes to report a defect, customer support can advise the customer to turn on logging at this level, run the problem scenario, capture the logs, and attach them to the trouble report.

Logging Based on Application Layers

For ease of configuration, the RPASCE Client provides the option of configuring different layers of the application at different log levels. The aim is to help the user concentrate on the layer that is of significance at any given point, while helping to reduce performance overhead by reducing logs that are not required.

The different layers in the configuration file log4jconfig.xml can be identified with the value of property in the xml file. Here is an extract from the xml file:

```
<Logger name="common.security" additivity="false" level="info" > - Layer Name  and
layer logging level :: change the level here
    <AppenderRef ref="EdgeAppender"/> - Where to log: choose where to log here
  </Logger>
```

Each layer has an appropriate entry in the xml file. You can change the parameters for level and appender, based on your requirements. The following layers are available:

- common.security
- common.view
- common.init
- common.services
- common.model
- common.data
- common.perf
- common.detailperf
- common.util

- common.i18n

- common.trans

- common.requests

- common.requestdetail

> **Note:** The appender can be the console or another appender that has been declared in the file. If you want to logs to be written to Edge_ .log, then use appender (see Log Files) in the <appender-ref> property.

Performance Logging

To generate performance logging, set the log level for both the perf and the perfdetail categories to debug and set the property perftiming.enabled to true in the config.properties file (located under the RPASCE Client config directory).

For more details on performance logging, see to Customer Performance Issue Report - Note 1493747.1 on My Oracle Support.