Oracle® Retail Merchandising Cloud Services

Administration Guide

Release 16.0.027

E96476-02

October 2018

ORACLE®

Oracle Retail Merchandising Cloud Services Administration Guide, Release 16.0.027

E96476-02

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

Send Us Your Comments

Oracle® Retail Merchandising Cloud Services Administration Guide, Release 16.0.027

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**.

x

Preface

This guide describes the administration tasks for Oracle Retail Merchandising Cloud Services.

Audience

This guide is intended for administrators.

This guide describes the administration tasks for Oracle Retail Merchandising Cloud Services.

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.

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 these documents through My Oracle Support.)

Conventions

The following text conventions are used in this document:

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

1

Administrative Tasks

This chapter describes the processes for maintaining users and roles as well as batch processes. For information regarding standard end user activities like creating and viewing reports, please see the *Oracle Retail Merchandising Cloud Services User Guide*.

Oracle Support

It is considered to be a best practice to have all Oracle Retail Merchandising Cloud Services support requests submitted through a single point of contact for that customer environment; the client designated administrator is usually designated to perform this role.

The link to use when submitting Service Requests (SR) is:

https://support.oracle.com

User Creation

Before users can access the Oracle Retail Merchandising Cloud Services applications it is necessary to provision each user access to the system, and assign roles to each user to control what functionality will be available to them. The access provisioning is done using Oracle Identity Management (OIM). The following steps explain how to define users, assign roles and revoke access for users when needed. The OIM Application URL and the login with the required administrator access would be needed to execute the below steps:

1. Log into the OIM application.

2. Under Administration, click **Users**.

Figure 1–1   Select Users
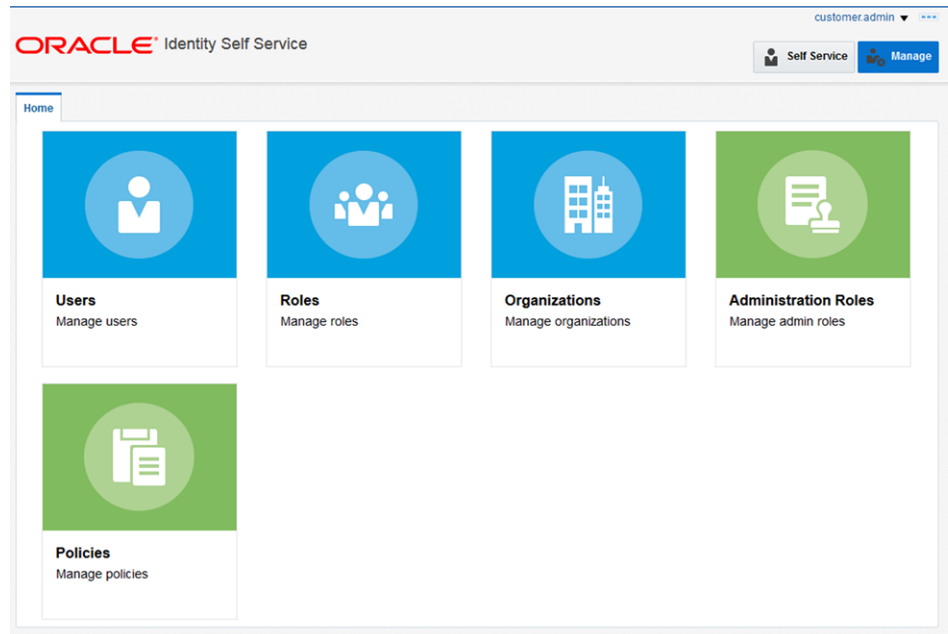
3. Under Actions, click **Create**.

Figure 1–2   Select Create

The Create User screen opens.

4. Under Basic Information, enter the following:

   ■ First Name

   ■ Last Name

   ■ For Organization, enter *Retail*

   ■ For User Type, enter *Full time employee*

   ■ For E-mail, enter the e-mail address of the employee

5. Under Account Settings, enter the following:

   ■ User Login: <firstname>.<lastname>

   ■ Password, enter a password

   ■ Confirm Password, reenter the password

Figure 1–3 Complete User Information

6. Click **Submit**.

Role Creation

To create a role, complete the following:

1. Log into the OIM application.

2. Click **Roles**.

Figure 1–4 Select Roles
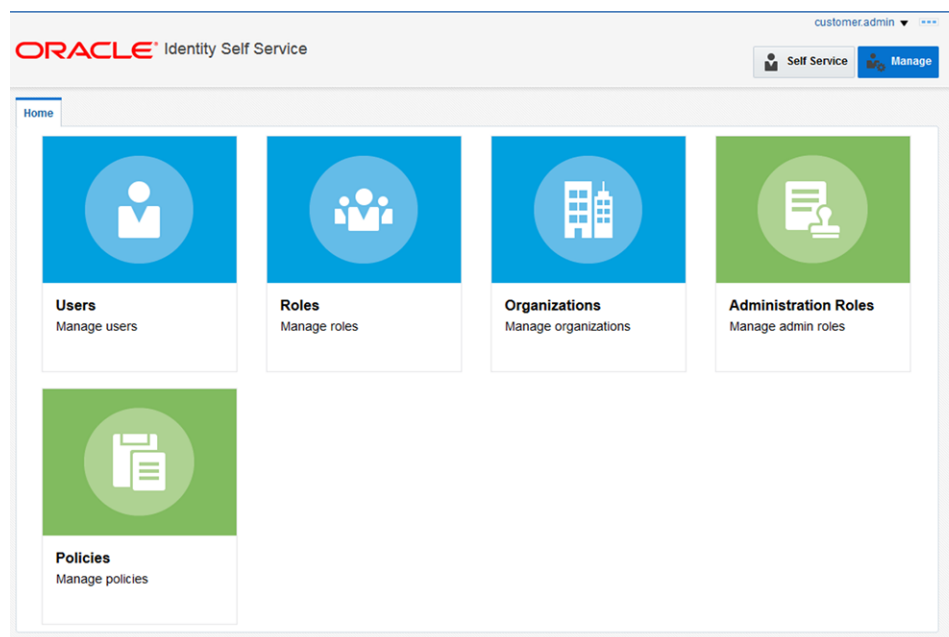
3. Under Actions, click **Create**.

Figure 1–5   Select Create
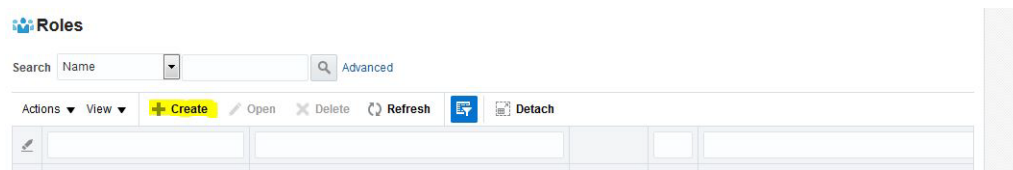
The Create Role screen opens.

4. Under General Role Information, enter the following:

    ■ Name: <RoleName>

    ■ Display Name: <RoleName>

Figure 1–6   Complete General Role Information
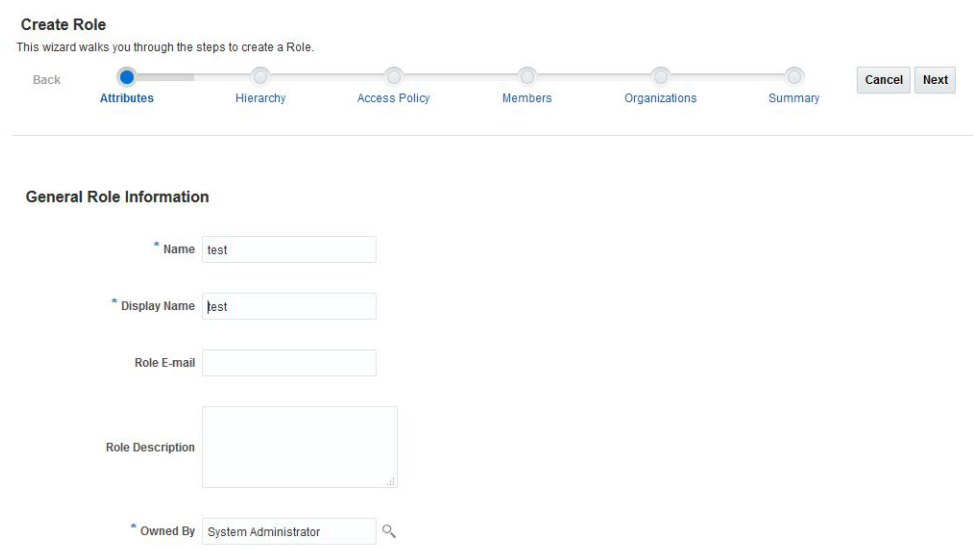
5. Click **Next** and provide the required details in the coming screens. For Organization, select **retail**. After reviewing the summary, click **Finish** to create the role.

Figure 1–7   Finish Creating Role
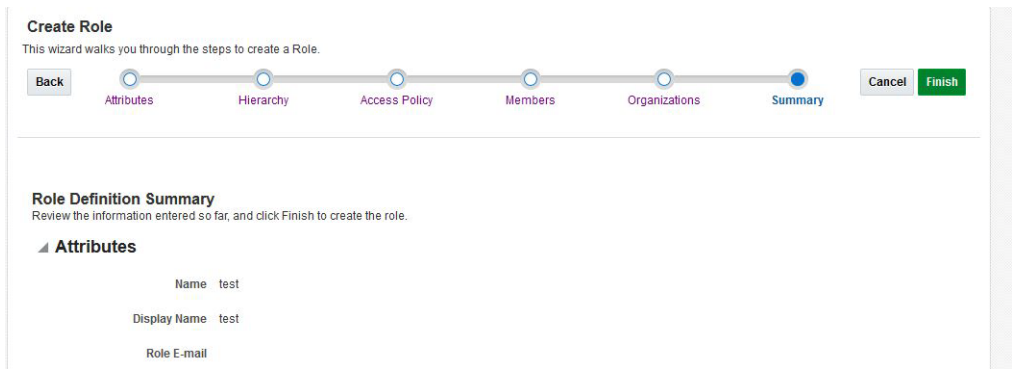
Assigning Users to a Role

To assign users to a role, complete the following:

1. Log into the OIM application.

2. Click **Users**.

Figure 1–8   Select Users
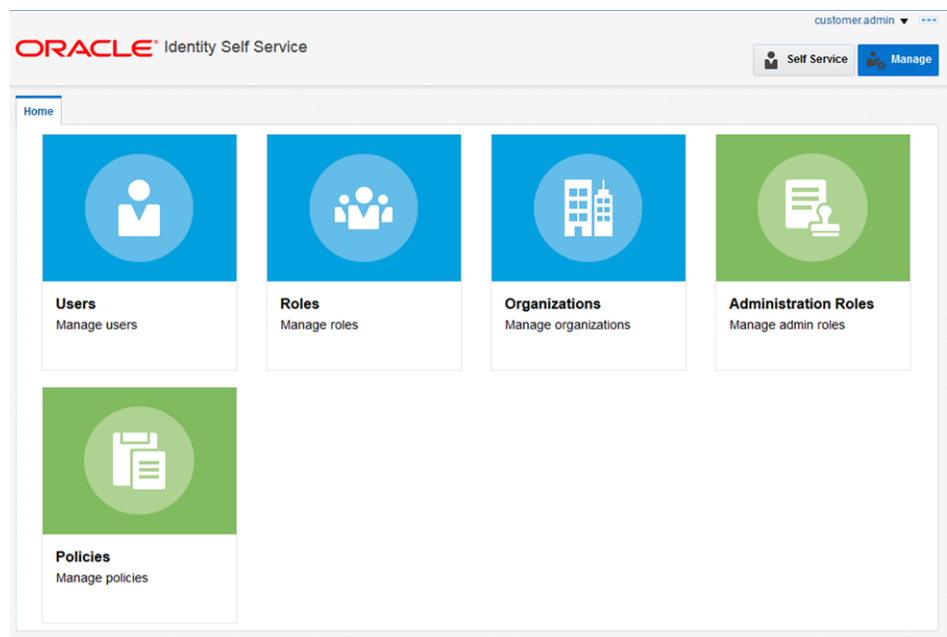
3. Click the oim.test user.

Figure 1–9   oim.test User

4. Click the Roles tab.

Figure 1–10   Roles Tab
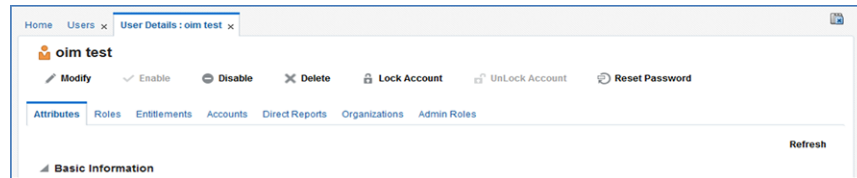
5.   Click the **Request Roles** button.

Figure 1–11   Request Roles Button
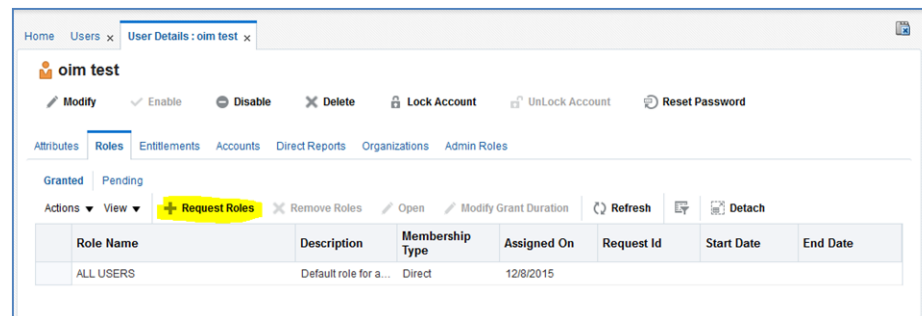
6.   Click the **Add to Cart** button next to the role you want to assign.

Figure 1–12   Adding Roles to the Cart
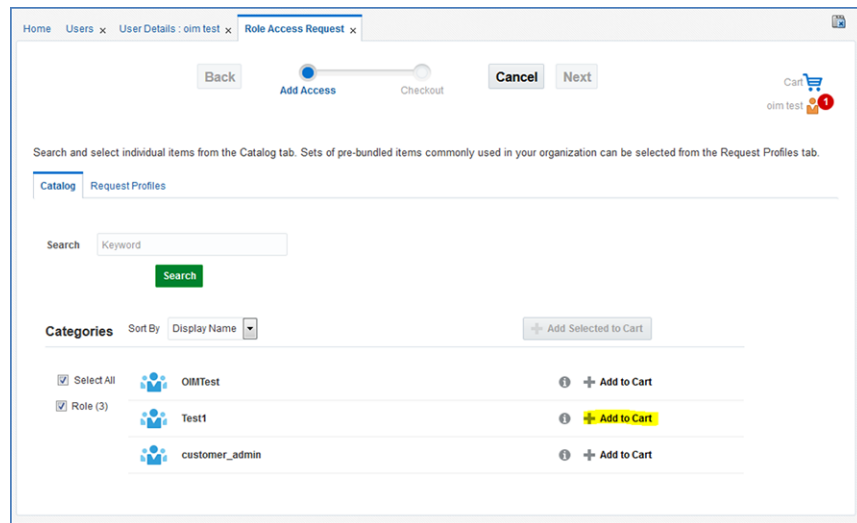
7.   Click **Next**.

Figure 1–13   Add Access Request
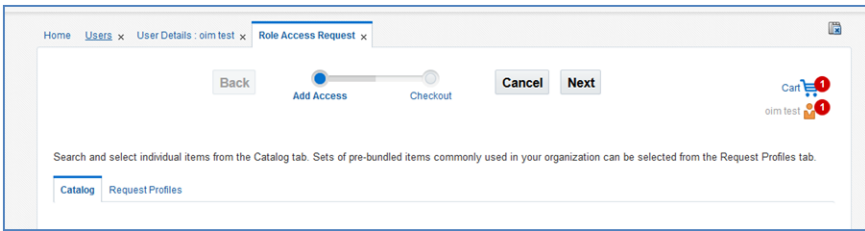
8. Click **Submit**.

Figure 1–14   Submit Access Request
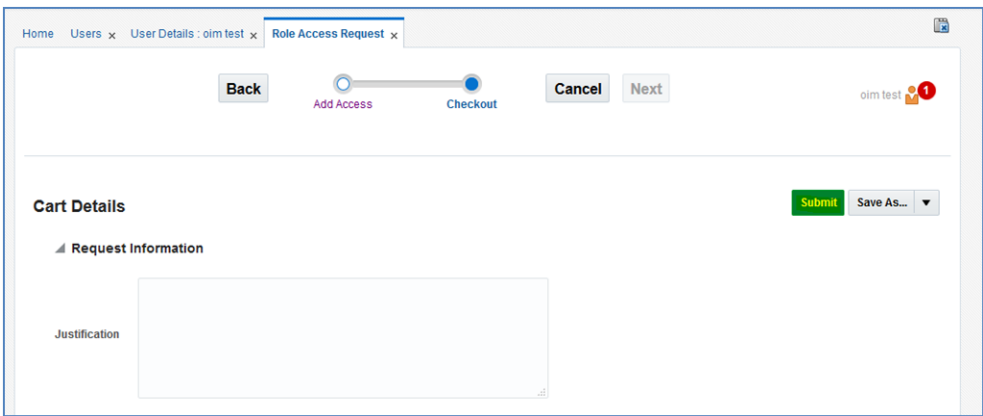
The role is now assigned to the User.

Figure 1–15   User Details

2
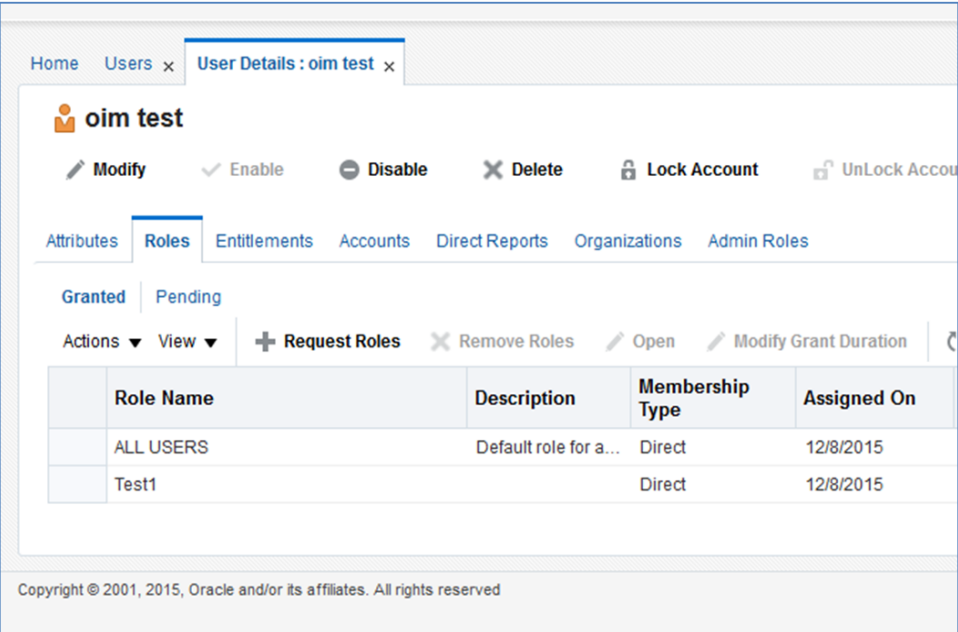
Manage Security Policies

As part of the Oracle Retail application security set up, default enterprise roles and their mappings to application roles are provided with every application. These enterprise roles/groups may not map one-to-one to retailers' security groups. The Manage Security Policies option in the Administration menu, accessed through the Settings icon on the task bar, provides a way for retailers to modify the default enterprise roles so they map to their security groups. It also provides a way for the retailers to change the mappings of the enterprise roles/groups to the application roles.

> **Note:** Merchandising maintains various policy stripes in the OPSS schema. The policy stripes are maintained to ensure that the changes done by the retailer on the application security setup are preserved on the subsequent patch installs. A policy patching screen is also provided where the retailers can merge the changes introduced in a patch with the changes done in the application policy setup.

Figure 2–1 Administration Menu
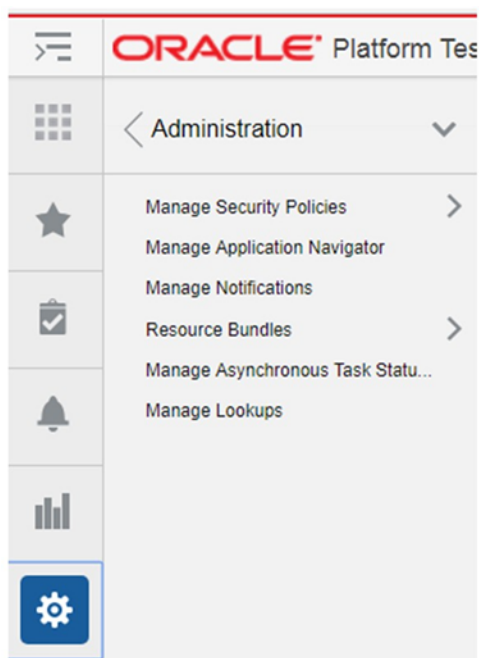
Navigation

Access to this functionality is available only for users with the following job roles:

Table 2–1   RAAC Security Privileges

Role	Duty	Privilege
ReIM Application Administrator	Admin Duty	Application Navigator Privilege
		Role Manager Privilege
		Notification Manage Privilege
		Resource Customization Privilege
Allocation Application Administrator	Admin Duty	Application Navigator Privilege
		Role Manager Privilege
		Notification Manage Privilege
		Resource Customization Privilege
Pricing Application Administrator	Admin Duty	Application Navigator Privilege
		Role Manager Privilege
		Notification Manage Privilege
		Resource Customization Privilege
ReSA Application Administrator	Admin Duty	Role Manager Privilege
		Application Navigator Privilege
		Notification Manage Privilege
		Resource Customization Privilege
RMS Application Administrator	Admin Duty	Role Manager Privilege
		Application Navigator Privilege
		Notification Manage Privilege
		Resource Customization Privilege

Accessing RAAC

RAAC is accessed through either the links in the Administration menu of the Settings menu or through the RAAC link in the administrator's user menu on the Global (top) bar of the screen.

1.   Click the RAAC application link in the user menu.

Figure 2–2   RAAC User Menu
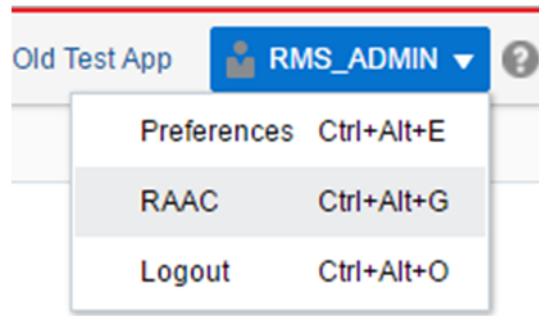
A new tab window or browser window is launched (depending on your browser settings).

2. From the Retail applications' login screen (for non-single sign-on), enter your username and password.

Roles Explained

The following is an explanation of Enterprise (or Job) Roles and Application (or Duty and Privilege) roles:

Job Roles

Job roles are called as such because they closely map to the jobs commonly found in most retailer organizations.

Naming Convention: All retail Job role names end with' _JOB'.

Example: ALLOCATOR_JOB.

Duty Roles

Duties are tasks or responsibilities one must do on a job.

Duty roles are roles that are associated with a specific duty or a logical grouping of tasks. Generally, the list of duties for a job is a good indicator of what duty roles should be defined.

Duties that you create should be self-contained and pluggable into any existing or new job role.

Naming Convention: All retail duty role names end with' _DUTY'.

Example: ALC_ALLOC_POLICY_MAINTENANCE_MANAGEMENT_DUTY.

Privilege Roles

Privilege is the logical collection of permissions. A privilege can be associated with any number of UI components. Privileges are expressed as application roles.

Naming Convention: All retail privilege role names end with' _PRIV'.

Example: ALC_ALLOC_SEARCH_PRIV

Security

RAAC uses ADF security to implement functional security. The enforcement of functional security is delegated to the Fusion Middleware.

RAAC allows retailers to tailor Oracle Retail applications' default security models to their enterprise business model. This tool also helps retailers manage their Retail applications' security metadata. RAAC provides:

- The ability to create new Application Roles (DUTY) through the Manage Roles Mapping task.

- The ability for a retailer to change the default roles mapping based on its business needs.

- The ability for clients to backup the roles mapping to a file in their local file system.

- The ability for a retailer to preserve the changes done to the default application polices beyond patch installs and merge the policy changes introduced in a patch with the policy changes done by the retailer.

- The ability for a retailer to backup the roles mapping to a backup OPSS Stripe. The backup stripe can be used to download the backup policies at any time or to restore the application policies using the backup policies.

> **Note:** The Roles mapping changes in RAAC will only change the launching application's functional behavior and not the RAAC application itself.
>
> For additional information on changing the RAAC application Roles Mapping, see the Managing Policies with Fusion Middleware Control section in the *Securing Applications with Oracle Platform Security Services Guide*.

Task Flows

The Manage Security Policies area of RAAC consists of the following task flows to fulfill the above mentioned business requirements.

- Manage Duty Roles

- Manage Roles Mapping

- Manage Policy Patching

- Manage Backups

It is advisable that before starting to customize the Security Policies, the user should use the Manage Backups feature to make a backup of the current policies.

Manage Duty Roles

Figure 2–3   Manage Duty Roles Screen

The Manage Duty Role task flow allows users to create or delete duties. Users can create a new duty role to map according to the retailers enterprise business model.

When a user clicks the manage duty role task flow from the regional area, it opens a new tab in the local area. A table is displayed with all the duties associated with the application. The table toolbar has the following action menu:

- Add

- Edit

- Delete

- Export to Excel

A quick search component is provided on the table toolbar to quickly look up any desired record.

Add

The Add action is enabled at all times and allows you to add a new duty role.

Add a Duty Role

1. From the Actions menu, select Add. The Add Duty Role dialog is displayed.

Figure 2–4   Add Duty Role Dialog

2. Enter a duty name and (optionally) a description for the new duty role and click OK.

Upon entering a name and moving to the Description field or when you click OK, RAAC performs a validation to prevent name duplication. If the name entered already exists, the following error message is displayed; A Duty with this name

already exists. Enter a unique name. If the duty name is unique, the new duty is created.

All newly created duties are reflected in the Policy Store, allowing them to be mapped to a role.
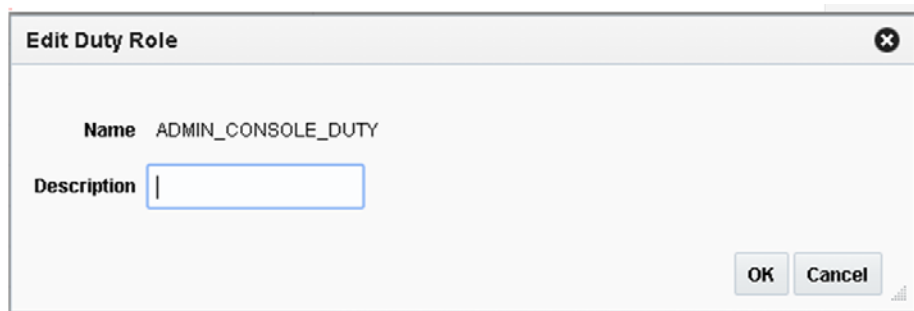
Edit

The Edit action is enabled when a duty role is selected. The Edit action is used to add or modify the description of an existing duty role.

Edit a Duty Role

1. Select the duty role you want to edit.

2. From the Actions menu, select Edit. The Edit Duty Role dialog is displayed.

Figure 2–5 Edit Duty Role Dialog

3. Add or modify the description of the duty role and click OK. Any updates to the description are updated to the mapping that the duty is mapped against.

Delete

The Delete action is enabled when you select a level that can be deleted from the database.

Only empty duty roles that do not contain any child duty or child privilege roles can be deleted. This is because a child privilege is associated with application code. Therefore, to delete the duty role, you must first remove all mappings between the duty and the child duties and privileges. If you attempt to delete a duty role without first removing the mapping, the following error message is displayed; You cannot delete duty because it has duties or privileges associated with it. Remove these mappings using Manage Roles Mapping task.

Delete a Duty Role

1. Select the duty role you want to delete.

2. From the Actions menu, select Delete. The Confirm dialog is displayed.

Figure 2–6   Delete Duty Role Confirmation Dialog
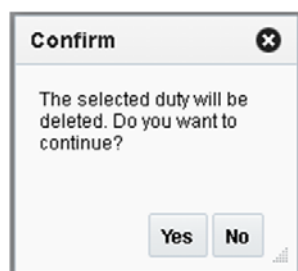
3. Click Yes to delete the selected duty role.

Export to Excel

The Export to Excel action is enabled whenever you have the Manage Duty Roles window open. This action allows you to export the Manage Duty Roles list to a Microsoft Excel spreadsheet.

Manage Roles Mapping

Figure 2–7   Manage Roles Mapping Screen

The Manage Roles Mapping task allows a user to add or remove duties or privileges associated with a job role.

When the user clicks on the Manage Roles Mapping link, a new tab is opened in the local area that displays a tree table with the already associated roles of the respective application (that is, the source application from which RAAC was launched). You can then choose to perform the following actions:

- Duplicate
- Delete
- Export to Excel
- Select and Add
- Remap

A quick search component is provided on the table toolbar to quickly look up any desired record.

Duplicate

The Duplicate action allows you to copy and paste existing job roles to create a new role. For example:

Within a retailer enterprise security system there are two security groups called Allocator_xyz and Assistant_ Allocator_xyz. By default, only the Allocator role is provided. The Assistant Allocator role is a trimmed down version of the Allocator role with fewer privileges. You can use the Duplicate action to copy the Allocator role and then edit the privileges of the new role.

Duplicate a Job Role

1. Select the job role you want to duplicate.

2. From the Actions menu, select Duplicate. The Duplicate Role dialog is displayed.

Figure 2–8   Duplicate Role Dialog

3. Enter a name for the duplicated role and click OK.

   Upon clicking OK or tabbing out of the field, the New name is validated. The validation checks for a valid name in the enterprise security solution of the retailer. If it finds a match, then it allows the user to proceed. If the name is not valid, an error is displayed.

Delete

The Delete action is enabled when a duty or privilege role is selected.

Duty Role: When the user selects a duty and clicks Delete, the user is prompted with a warning message. Click 'Yes' and the selected duty role mapping will be removed from the corresponding Job role. All other Job roles mappings with that duty role will not be affected.

Privilege Role: When the user selects a privilege and clicks Delete, the user is prompted with a warning message. Click 'Yes' and the selected privilege role mapping will be removed from the corresponding duty role. All the Job roles having that duty role mapping will lose that privilege.

> **Note:** The Delete action does not delete the role from the application policies, it only removes the mapping.

Export to Excel

The Export to Excel action is enabled whenever you have the Manage Roles Mapping window open. This action allows you to export the Manage Roles Mapping list to a Microsoft Excel spreadsheet.

Select and Add

Select and Add is enabled when a job or a duty is selected. Select and Add allows you to do the following:

- Add duty roles to job roles

- Add duty and privilege roles to duty roles

Select and Add a Duty

1. Select the job role to which you want to add a duty role.

2. From the Actions menu, select Select and Add. The Select and Add dialog is displayed.

Figure 2–9   Select and Add Dialog
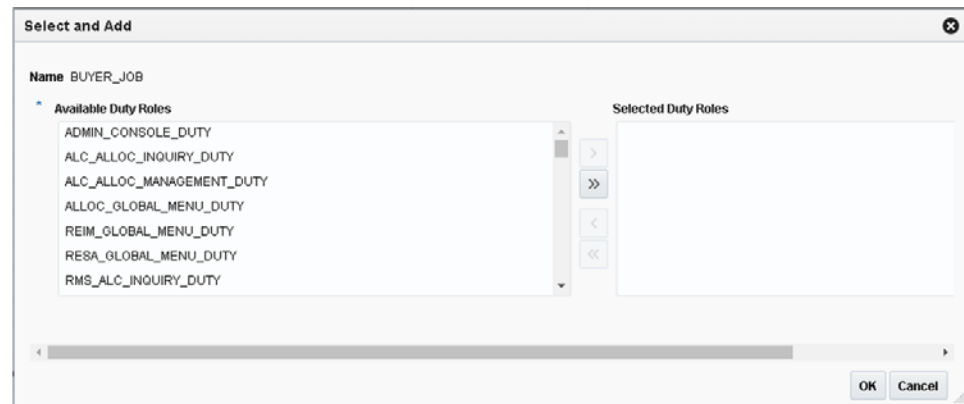
The Select and Add dialog has selected job or duty name displayed and a shuttle component for the user to select from all the available duties or privileges for the applications.

3. Select one or multiple duties and privileges and click OK to select and add them to the job role.

    If you select a duty level then all the associated privileges with it are inherited to the job role. If a privilege is selected then only that privilege is added under the selected duty role.

    > **Note:**   To add a new job role, the user should use the Duplicate feature.
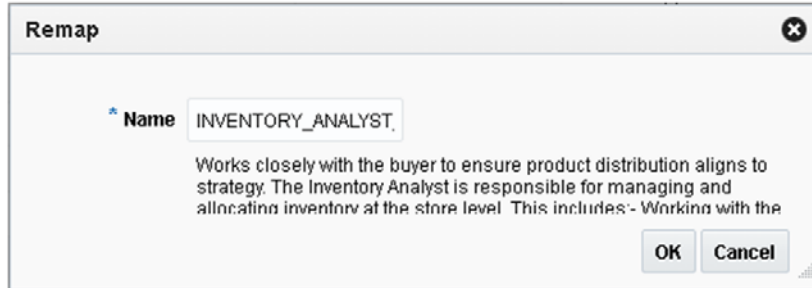
Remap

The Remap action is enabled when a job or duty is selected. The Remap action is used to move mapping from one role to another role. During this process, a new role is created and all the associated roles beneath the previous role are moved into the new role, leaving the old role as an orphan or with other roles associated with it.

Remap a Role

1. Select the role you want to remap.

2. From the Actions menu, select Remap. The Remap dialog is displayed.

Figure 2–10   Remap Dialog
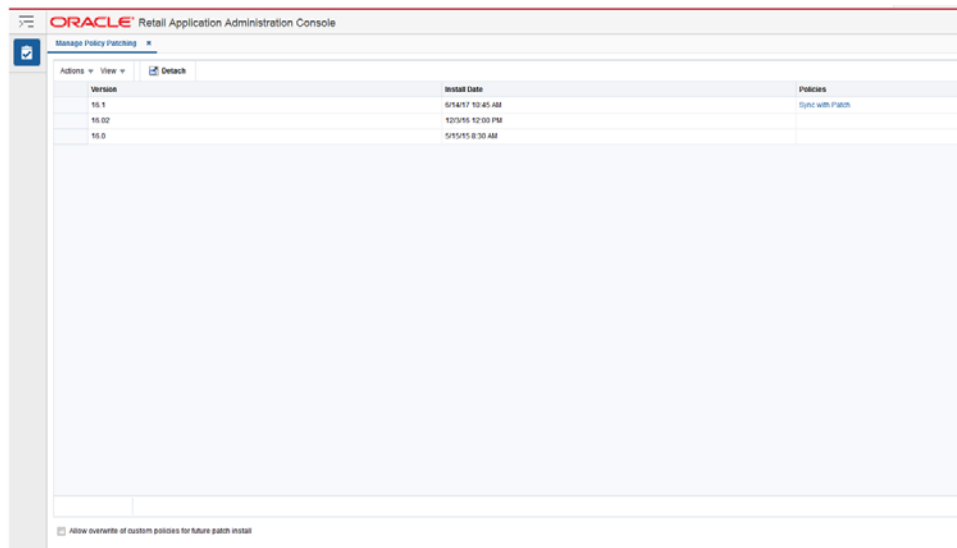
3. Enter a Name and (optionally) a Description for the remapped role.

   Upon clicking OK, the New name is validated. The validation checks for a valid name in the enterprise security solution of the retailer. If it finds a match, then it allows the user to proceed. If the name is not valid, an error is displayed.

   Job roles are handled in a different manner. When a job role is selected and remapped it will just change the current mapping to the new mapping without creating any new job role to be left as an orphan. This validation should be done once the user enters the new name and clicks OK. If there is no matching record found an error is displayed.

Manage Policy Patching

Figure 2–11   Manage Policy Patching Screen Patch History View

The Manage Policy Patching screen displays two different views. The first view is called the Patch History view. The Patch History view displays the list of patches applied to the application. The latest patch provides a link to synchronize the changes

introduced in the patch with the changes done by the retailer on the applications policies. The Patch History view also provides a check box to allow overwrite of the application policies for future patch installs. When this check box is unchecked, the application installer will not overwrite any changes done by the retailer on the application policies. The retailer should merge the changes introduced in the patch with the changes done on the application policies using the Sync with Patch view. When the checkbox is checked, the application installer will overwrite the application policies with the ones in the patch. The installer backs up the custom policies before overwriting application policies. The backups can be managed using the Manage Backups screen.

Figure 2–12   Manage Policy Patching Screen Sync With Patch View
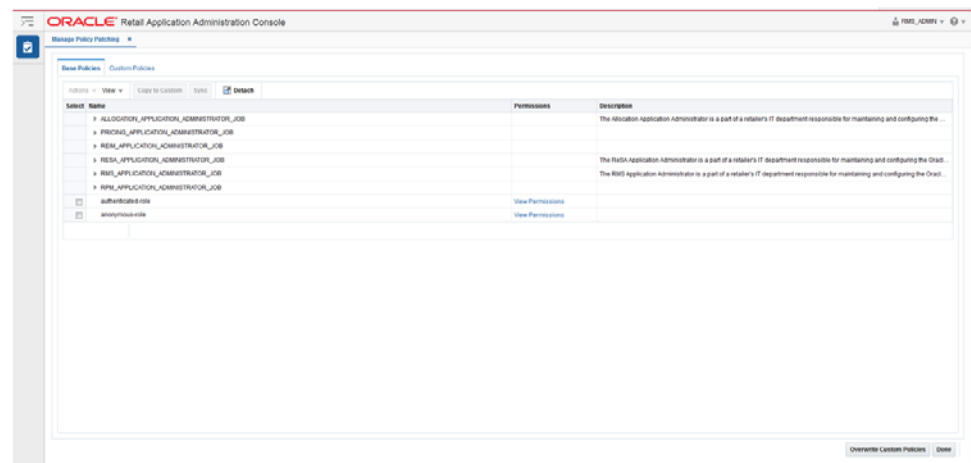
The Manage Policy Patching Screen Sync with Patch view provides a way for the retailer to synchronize the changes introduced in a patch with the changes done on the application policy setup. The Sync with Patch view displays two different tabs. The first tab called Base Policies displays the application policies that came in the patched application. The second tab called Custom Policies displays the application policies that the retailer has changed as per its business needs. The retailer can merge the changes introduced in the patch with the custom policy change using this view. In the current version of RAAC, the retailer can make use of the patch documented policy changes to help guide what changes to look for and merge.

The Sync with Patch view in the Manage Policy Patching screen provides the following actions

- Copy to Custom
- Sync
- View Permissions
- Overwrite Custom Policies
- Done
- Refresh

Copy to Custom

The Copy to Custom action is enabled when a privilege is selected and the privilege does not exist in the custom application policies. The Copy to Custom action copies

the privilege to the selected duty in the custom application policy setup.

Copy Privilege to Duty

1. Select the privilege in the base policies.

2. If the privilege does not exist in the custom policies, the Copy to Custom action will be enabled.

3. Click on Copy to Custom action to display the Copy to Custom Dialog.

Figure 2–13   Copy to Custom Dialog
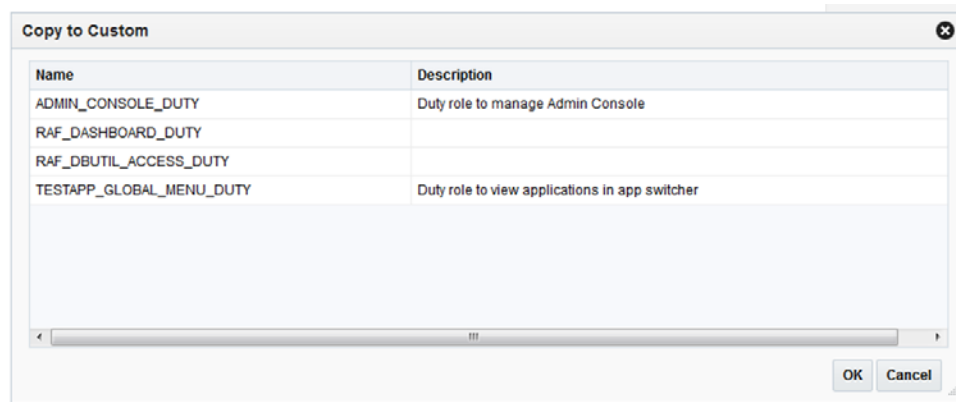
4. Select the duty where the privilege should be copied.

5. Click OK to copy the privilege to selected duty.

6. Successful copy will display the success dialog.

Figure 2–14   Copy to Custom Success Dialog
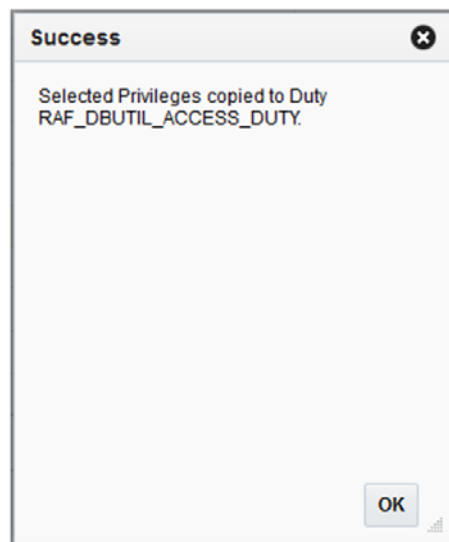
Sync

The Sync action is enabled when a privilege is selected and the privilege exists in the custom application policies. The Sync action synchronizes the permissions in the selected privilege with the same privilege in the custom application policy setup.

Synchronize Permissions in a Privilege

1. Select privilege in base policies.

2. If the privilege exists in the custom policies then the Sync action will be enabled.

3. Click on Sync to synchronize the permissions with the same privilege in the custom policy setup.

4. Successful sync will display the success dialog.
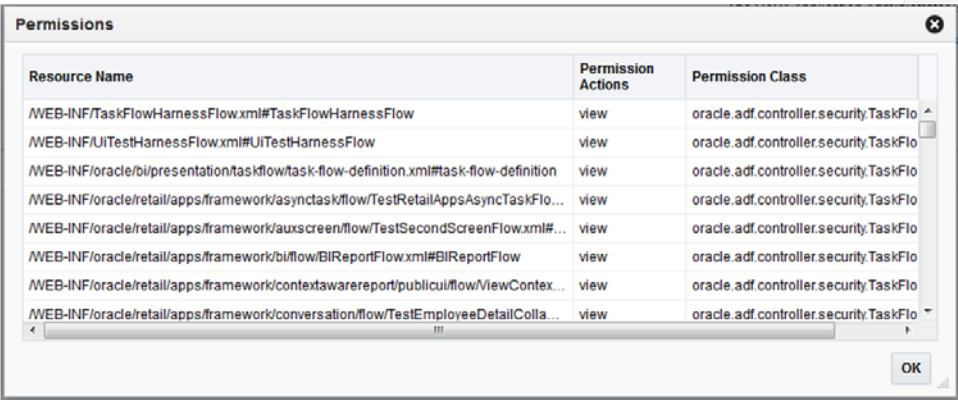
Figure 2–15   Sync Success Dialog
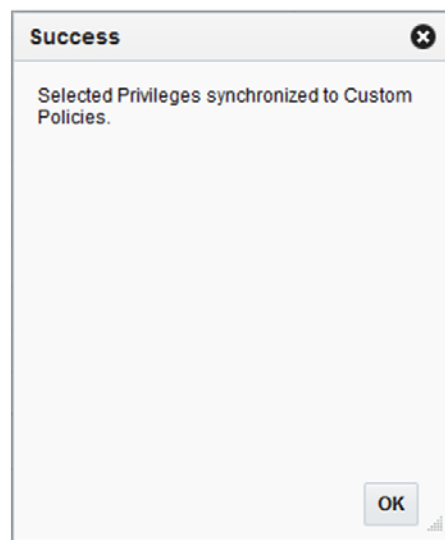
View Permissions

The View Permissions link is used to display the permissions associated with a privilege. This link opens up a popup that displays the Resource Name, Permission Actions and Permission Class for permissions.

Figure 2–16   View Permissions Dialog

Overwrite Custom Policies

The Overwrite Custom Policies action overwrites the current application policy setup with the application policies available in the patched application. The action will cause the loss of any changes done by the retailer on the application policies. The action displays a warning popup before overwriting application policies with the patched policies. The action also backs up the application policies before overwriting and can be retrieved using the Manage Backups screen.

Figure 2–17   Overwrite Custom Policies Dialog

Done

The Done action takes the user back to the Patch History view of the Manage Policy Patching screen.

Refresh

The Refresh action is only available in the Custom Policies tab of the Sync with Patch view of the Manage Policy Patching screen. The action is used to refresh the custom

application policies. The action can be used to verify the changes in the custom policies after a successful Sync or Copy to Custom action from the base policies.

> **Note:** Note
>
> The Sync with Patch screen can be used for the synchronization of the privilege and its permissions from the patched application policies to the custom application policies. If a new duty role has been introduced in a patch then the Manage Duty Role screen should be used to create a duty role and Manage Role Mapping screen should be used to assign it to the appropriate job role. Once the new duty role has been created and assigned to a job role, the Copy to Custom action can be used to assign privileges to that duty.

Manage Backups

Figure 2–18   Manage Backups

The Manage Backups screen is used to maintain the application policy backups. The backups can be used to restore the application policies back to the original state if something goes wrong with the application policy setup. The backups can be created by the following actions

■   Installer creates a backup before overwriting the application polices during a patch install. The installer created backup is prefixed with the name SYS_BACKUP. The installer overwrites the application policies and creates a backup only if the 'Allow Overwrite of Custom Policies' flag is checked in the Manage Policy Patching screen.

■   A user can create a backup using the Create action on the Manage Backups screen. The user created backup is prefixed with the name USER_BACKUP.

■   A user can initiate a backup using the 'Overwrite Custom Policies' action on the Manage Policy Patching screen. The Overwrite Custom Policies action creates a backup before overwriting the custom policies. The backup created by the Overwrite Custom Policies action is prefixed with the name SYS_BACKUP.

The Manage Backups screen provides the following actions:

■   Create

■   Delete

■   Download

■   Restore

- Refresh

Create

The Create action is used to create a backup of the current application policies. The backup stripe name will be prefixed with the text USER_BACKUP. The create action opens up a popup where the user can enter the comments for the backup.

The RAAC application allows only one user initiated backup. If a backup was already created by an administrator for the current application policies, it will be overwritten.

Figure 2–19   Create Backup Dialog
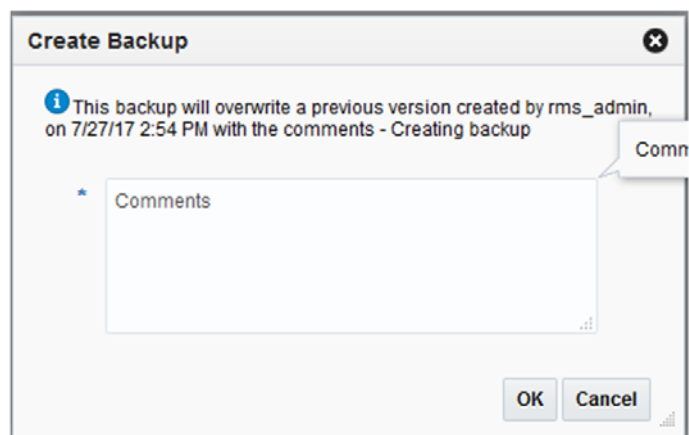
Delete

The delete action is used to delete the selected backup stripe. A confirmation dialog is displayed before deleting the backup.

Figure 2–20   Delete Backup Confirmation Dialog
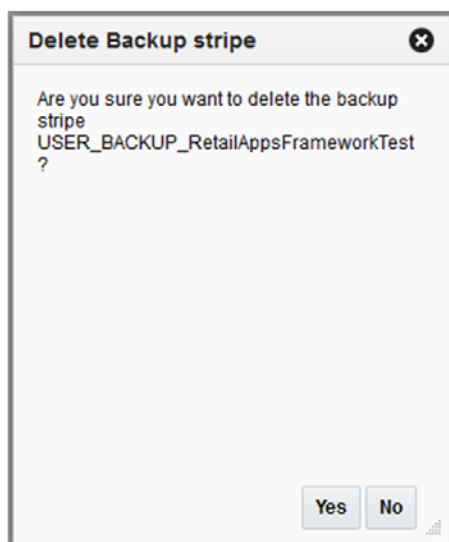
Download

The download action is used to download the selected backup stripe in an xml format. It will generate a file called backup.xml which can be stored on the device where the browser is running. The xml file can be opened in JDeveloper to look at the changes in the backup.

Restore

The Restore action will overwrite the current application policy setup with the application policies available in the backup. A confirmation dialog is displayed to the user before restoring.

Figure 2–21   Restore from Backup Dialog
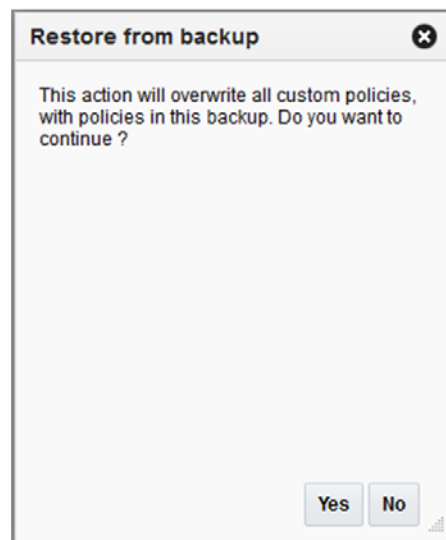
Refresh

The Refresh action will refresh the backup table.

3

Data Security/Filtering

Data Security/Filtering Implementation

If Data Level Security is enabled in Merchandising, then for the users to have access to the data, the below must be setup through the Data Upload Utility available in Merchandising. This is used both in the core Merchandising solution, as well as Allocation, Invoice Match, Pricing, and Sales Audit.

- Security Groups

- Security User and Security User Role

- Security User and Security Group mapping information

- Merchandise Hierarchy LOV filtering access information

- Purchase Order Approval Limits

Setup Security Groups

Perform the following procedure to define a Security Group in the system.

1. Navigate to Foundation Data > Data Loading > Download.

2. In the Download Data screen, select Template Type as 'Security' and Template as 'Security Groups'.

Figure 3–1 Download Security Groups Template Spreadsheet
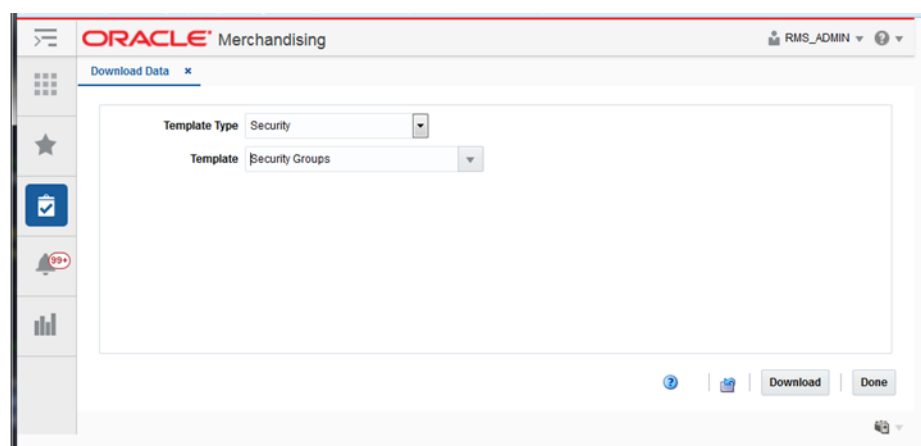
3. Click the **Download** button.

4. Save File to a local directory location when prompted.

Figure 3–2   Save File
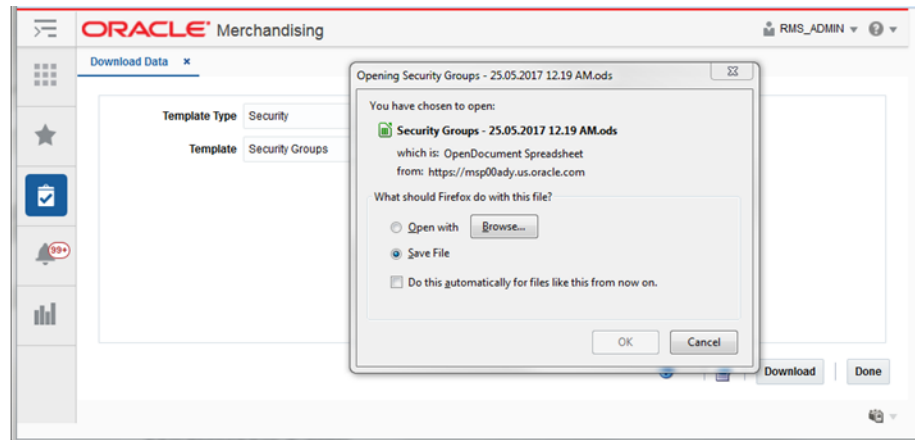
5. Click the **Done** button.

6. Open the downloaded file.

Figure 3–3   Open File
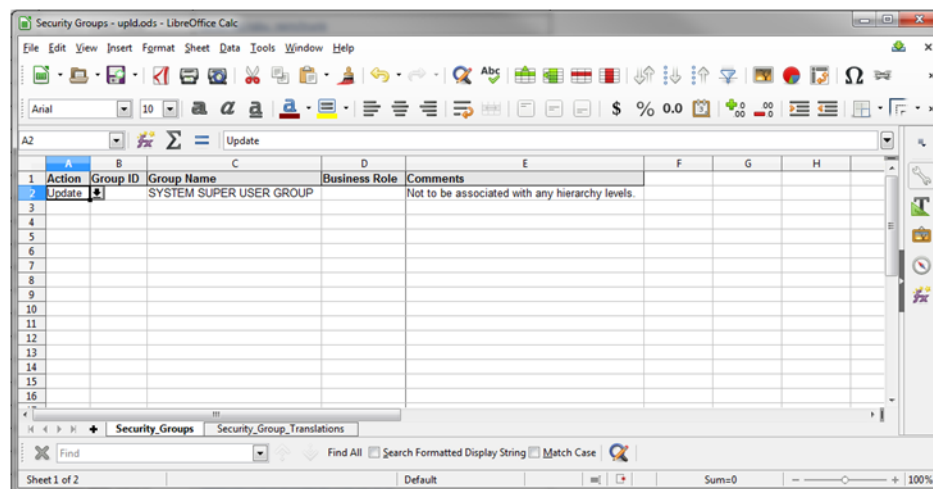
7. Save As < file name>.

Figure 3–4   Save as <File Name>
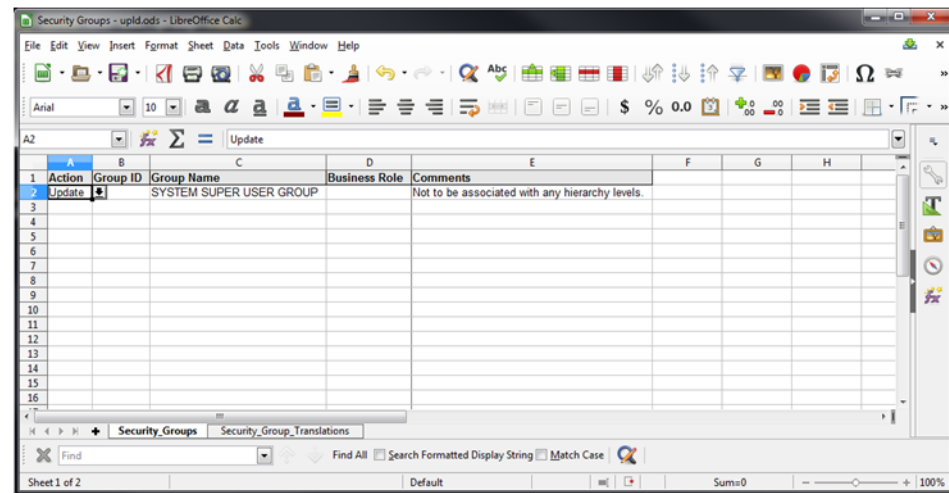
8. In the Security_Groups tab, enter/select the following:

    ■   Action: 'Create'

    ■   Group ID: <Group ID>

    ■   Group Name: <Group Name>

    ■   Business Role: <role> (optional)

    ■   Comments: <comments> (optional)

9. In the Security_Groups_Translations tab, enter the translated Security Group descriptions (optional).

Figure 3–5   Enter Security Groups information
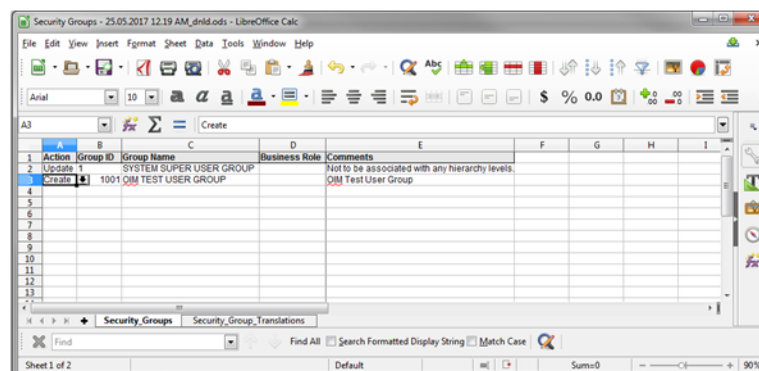
10. Save and Close the file.

11. In the RMS Application, navigate to Foundation Data > Data Loading > Upload.

12. In the Upload Data screen, select:

    ■   Template Type as 'Security'

    ■   Template as 'Security Groups'

    ■   Enter new Process Description or retain as is

    ■   Browse and select the Source file that was created in Step 10

Figure 3–6   Upload Security Groups Information

13. Click the **Upload** button.

Figure 3–7   Upload Security Group Information Complete

14. Click the **Done** button.

15. View the newly created Security Group by downloading the Security Groups spreadsheet (Steps 1 - 6).

Figure 3–8   View newly created Security Group

> **Note:** The system generated Group ID. This Group ID should be
> mapped to the Security Users.

Setup Security User and Security User Role

The Application (LDAP) User ID used to login to the RMS application must be defined
as a Security User.

1. Navigate to Foundation Data > Data Loading > Download.

2. In the Download Data screen, select Template Type as 'Security' and Template as
    'Security Users'.

Figure 3–9   Download Security Users template spreadsheet

3. Click the **Download** button.

4. Save File to local directory location when prompted.

Figure 3–10   Save File
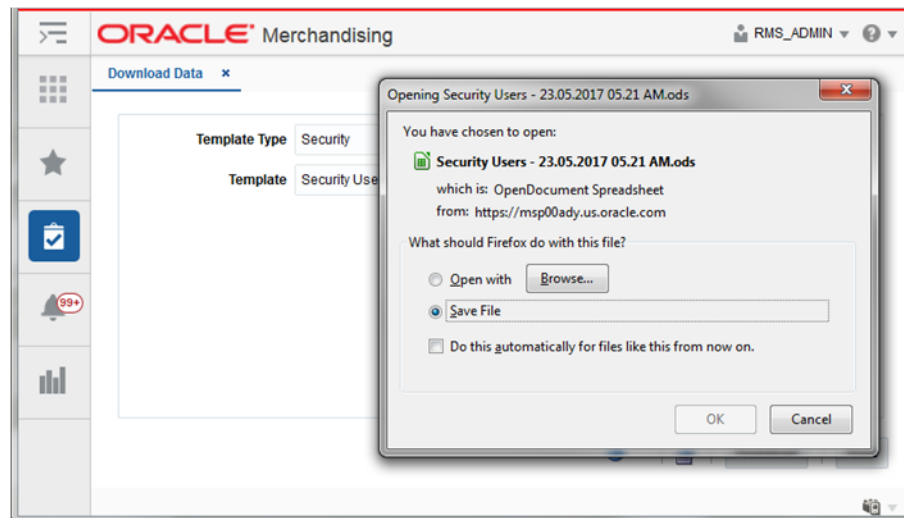
5. Click the **Done** button.

6. Open the downloaded file.

Figure 3–11   Open File
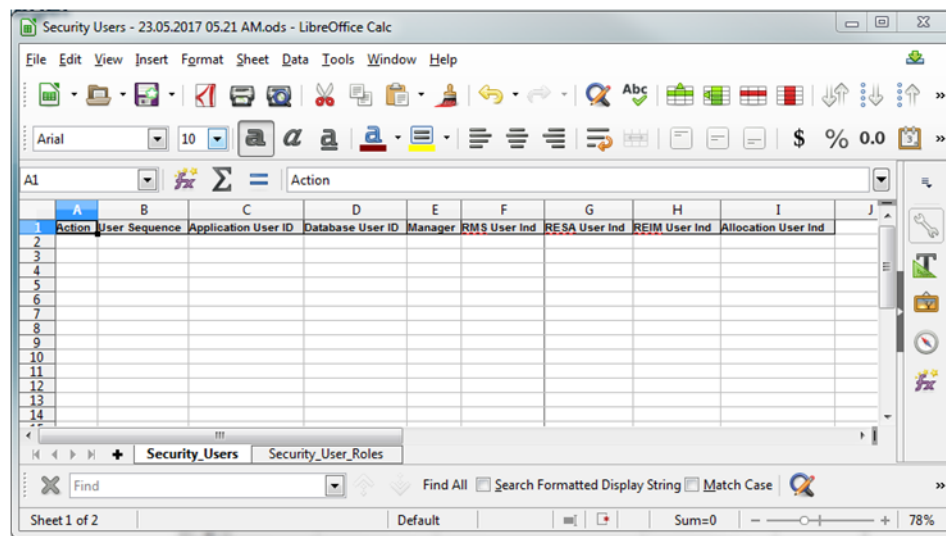
7. Save As < file name>.

Figure 3–12   Save as <File Name>
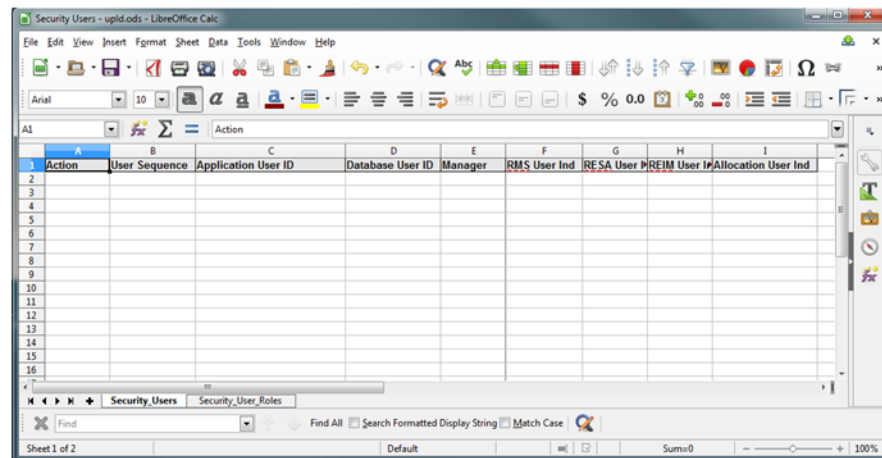
8. In the Security_Users tab, enter/select the following:

    ■   Action: 'Create'

    ■   User Sequence: <number>

    ■   Application User ID: <Application User ID>

    ■   RMS User Ind: Yes

    ■   ReSA User Ind: Yes/No

    ■   ReIM User Ind: Yes/No

    ■   Allocation User Ind: Yes/No

Figure 3–13   Security Users Information
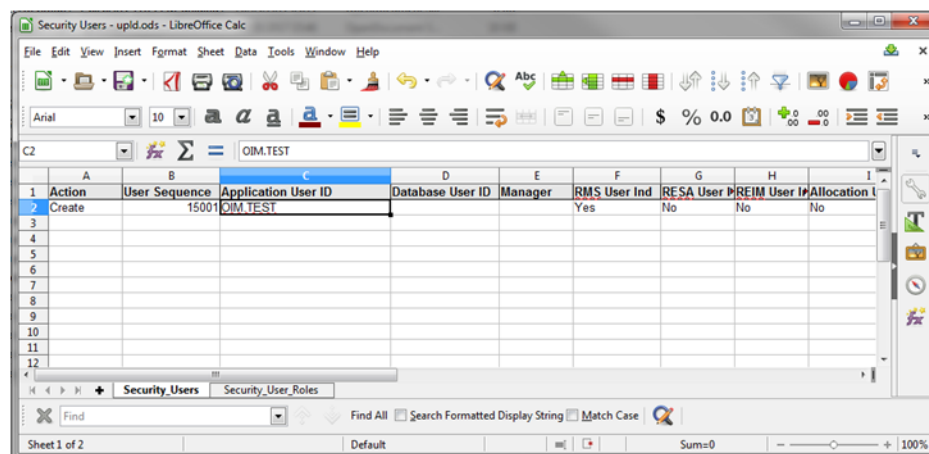
9. In the Security_User_Roles tab, enter/select the following:

    > **Note:**   Proceed to set up Security User Role, only if Role Privileges
    > have been setup.

    ■   Action: 'Create'

- User Sequence: <number>

Note: This must be the User Sequence provided in the Security_ Users tab.

- Role: <Role>

Figure 3–14   Security User Roles Information
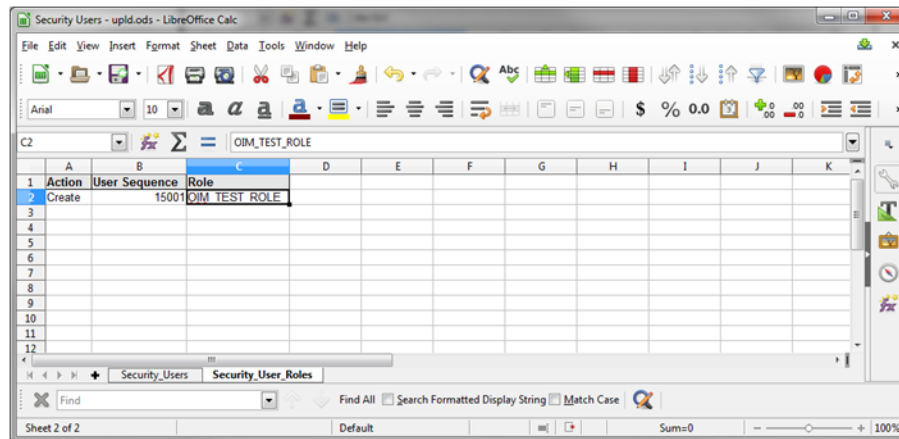
10. Save and Close the file.

11. In the RMS Application, navigate to Foundation Data > Data Loading > Upload.

12. In the Upload Data screen, select:

- Template Type as 'Security'

- Template as 'Security Users'

- Enter Process Description or retain as is

- Browse and select the Source file that was created in Step 10

Figure 3–15   Upload Security Users Information
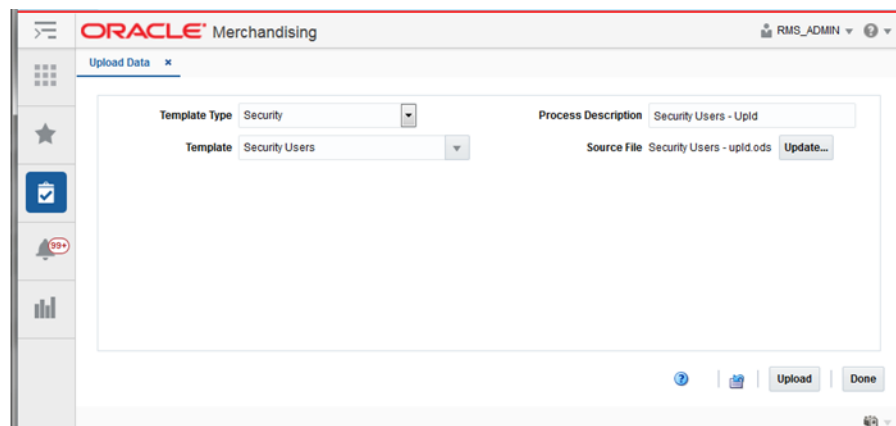
13. Click the **Upload** button.

Figure 3–16   Upload Security Users Information Complete
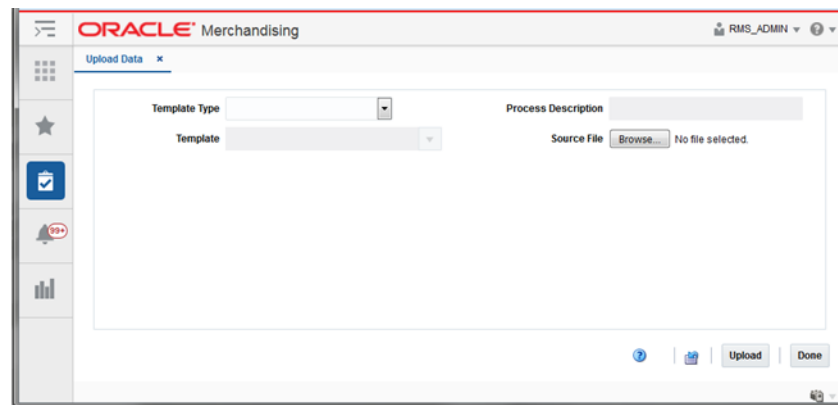
14. Click the **Done** button.

15. View the newly created Security User and Security User Role by downloading the Security Users spreadsheet (Steps 1 - 6).

Figure 3–17   View newly created Security User and Security Role Mapping information

Figure 3–18   View newly created Security User Role mapping information

Note the system generated User Sequence. This Security User (User Sequence) will be mapped to the Security Group.

Map Security User to Security Group

The security user must be assigned to a security group. This is achieved by associating the User Sequence assigned to the Application User ID with a Security User Group.

1. Navigate to Foundation Data > Data Loading > Download.

2. In the Download Data screen, select Template Type as 'Security' and Template as 'Associate Users to Groups'.

Figure 3–19   Download Associate User to Groups Template Spreadsheet


3. Click the **Download** button.

4. Save the file to a local directory location when prompted.

Figure 3–20   Save File
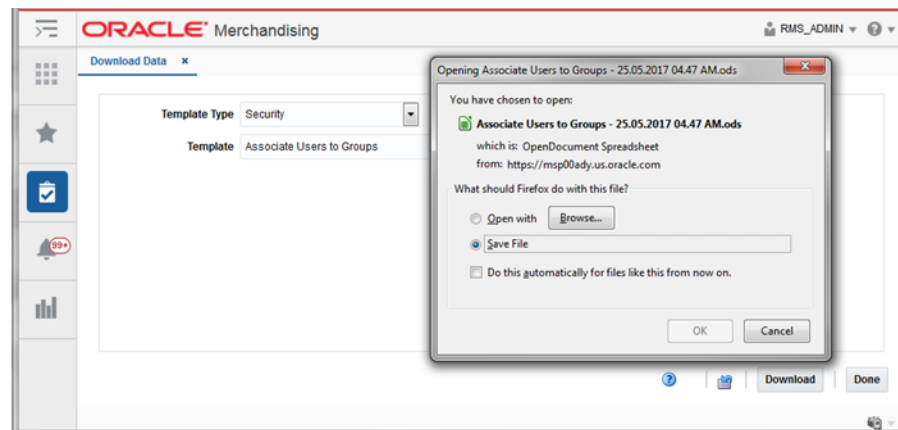
5.   Click the **Done** button.

6.   Open the downloaded file.

Figure 3–21   Open File
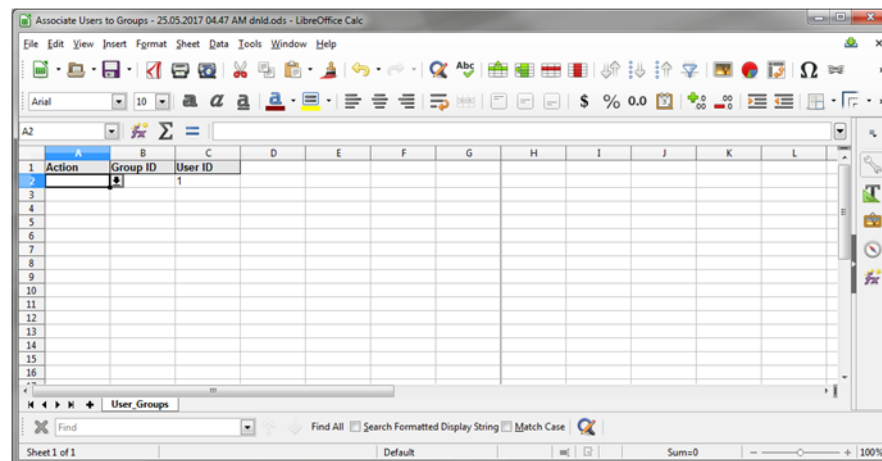
7.   Save As < file name>.

Figure 3–22   Save as <File Name>
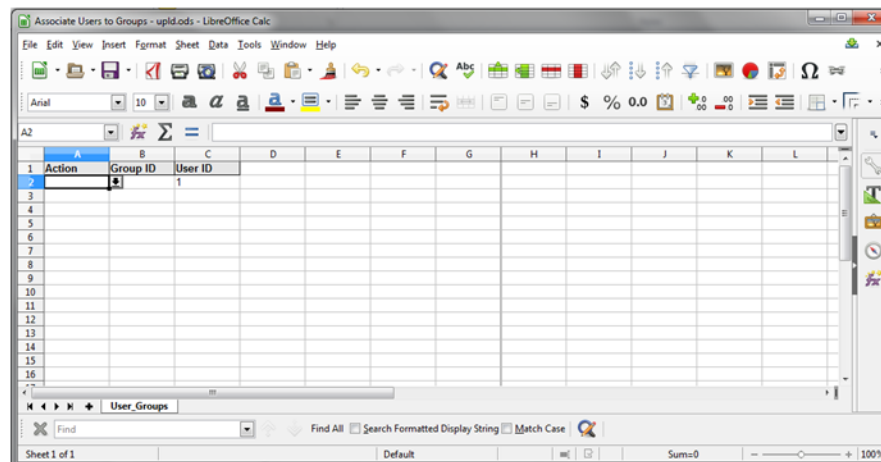
8. In the User_Groups tab, enter/select the following:

   - Action: 'Create'

   - Group ID: <Group ID>

   - User ID: <User ID>

Figure 3–23   Enter User Groups information
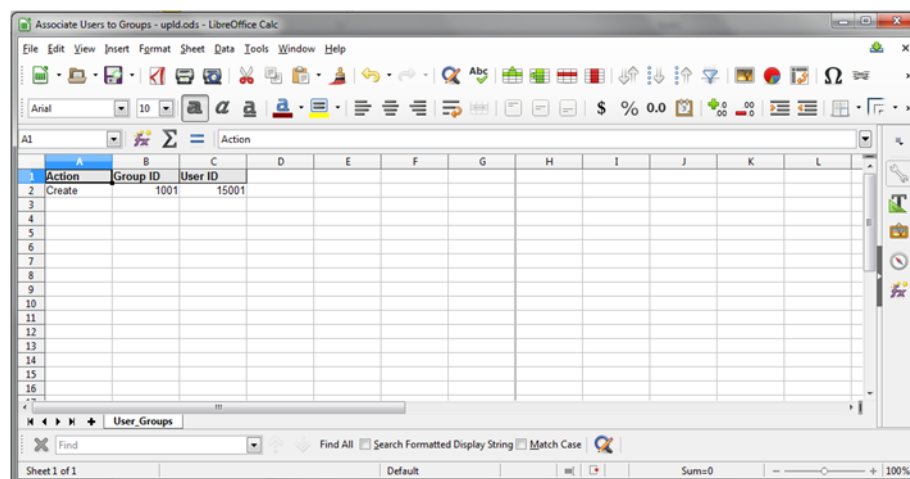
9. Save and Close the file.

10. In the RMS Application, navigate to Foundation Data > Data Loading > Upload.

11. In the Upload Data screen, select:

   - Template Type as 'Security'

   - Template as 'Associate Users to Groups'

   - Enter new Process Description or retain as is

   - Browse and select the Source file that was created in Step 9

Figure 3–24   Upload User Groups Information

12. Click the **Upload** button.

Figure 3–25   Upload User Groups Information Complete

13. Click the **Done** button.

14. View the newly created User Groups mapping by downloading the User Groups spreadsheet (Steps 1 - 6).

Figure 3–26   View Newly Created User Group Association
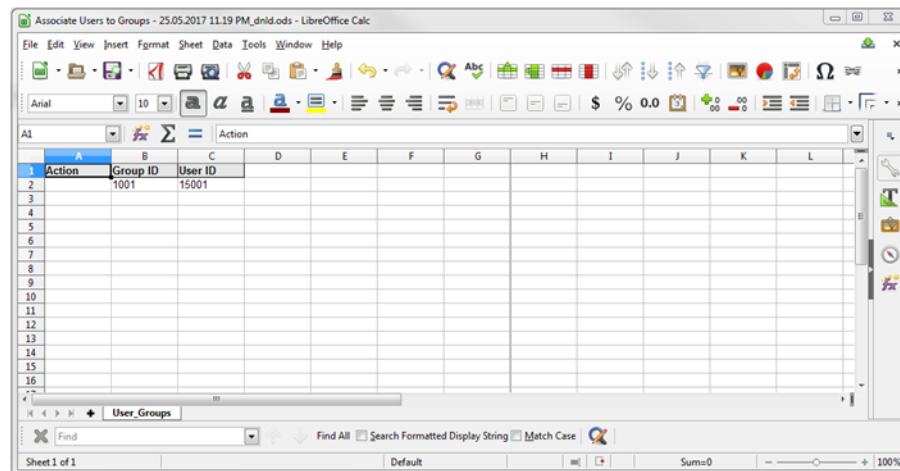
Assign Groups to Hierarchies

The security group can only access the merchandise hierarchies and organization hierarchies assigned to the user through Filter Groups. If a security group is not assigned to any Filter Group then the users in the group are considered 'super users' and will have access to all merchandise hierarchies or all organization hierarchies respectively.

1. Navigate to Foundation Data > Data Loading > Download.

2. In the Download Data screen, select Template Type as 'Security' and Template as 'Filter Groups'.

Figure 3–27   Download Filter Groups Template Spreadsheet
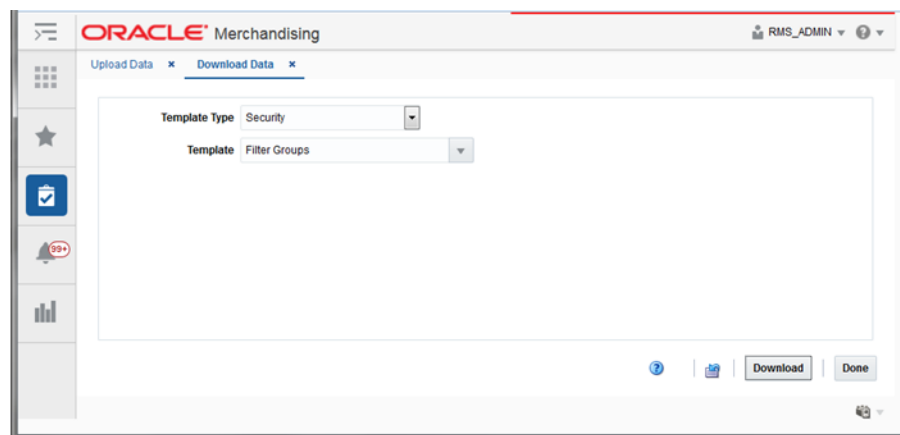
3. Click the **Download** button.

4. Save File to local directory location when prompted.

Figure 3–28   Save File
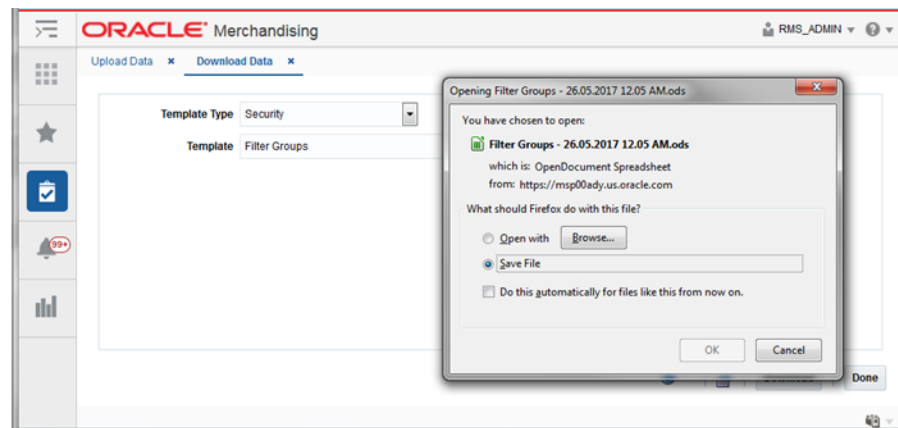
5.   Click the **Done** button.

6.   Open the downloaded file.

Figure 3–29   Open File
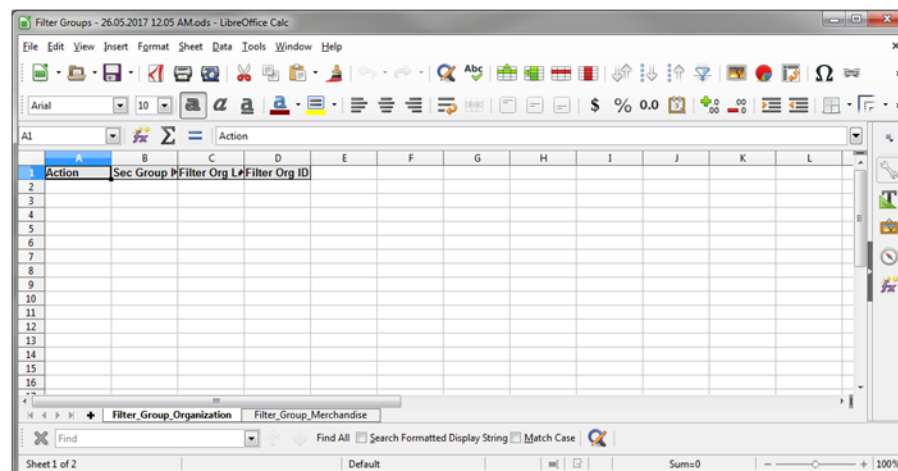
7.   Save As < file name>.

Figure 3–30   Save as <File Name>
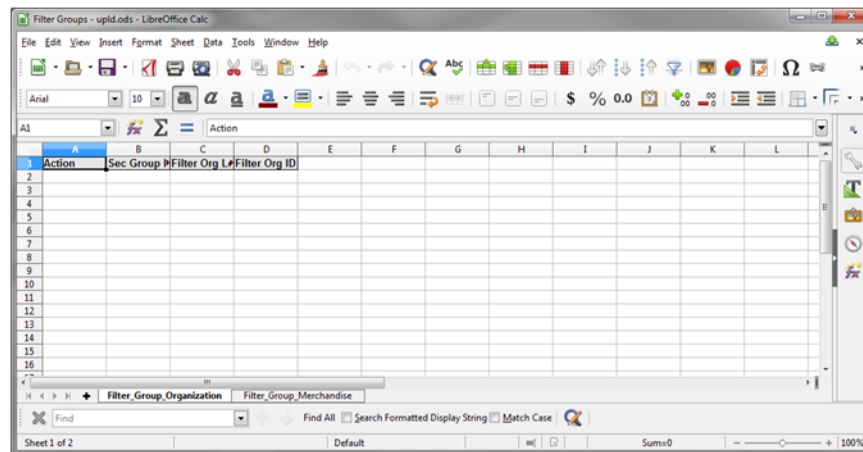
8. In the Filter_Group_Organization tab, enter/select the following:

    ■   Action: 'Create'

    ■   Sec Group ID: <User Security group ID>

    ■   Filter Org Level: <Organization hierarchy level>

    ■   Filter Org ID: <ID of the Organization hierarchy level>

Figure 3–31   Enter Filter Group Organization information
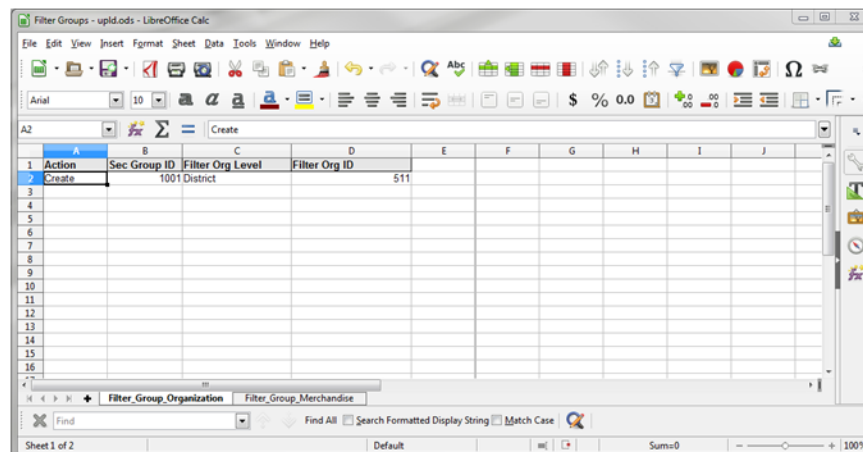
9. In the Filter_Group_Merchandise tab, enter/select the following:

    ■   Action: 'Create'

    ■   Sec Group ID: <User Security group ID>

    ■   Filter Merch Level: <Organization hierarchy level>

    ■   Filter Merch ID: <ID of the Merchandise hierarchy level>

    ■   Filter Merch ID Class: <Class ID of the Merchandise hierarchy level> (optional depending on the selected Filter Merch Level)

- Filter Merch ID Subclass: <Subclass ID of the Merchandise hierarchy level> (optional depending on the selected Filter Merch Level)

Figure 3–32   Enter Filter Group Merchandise information
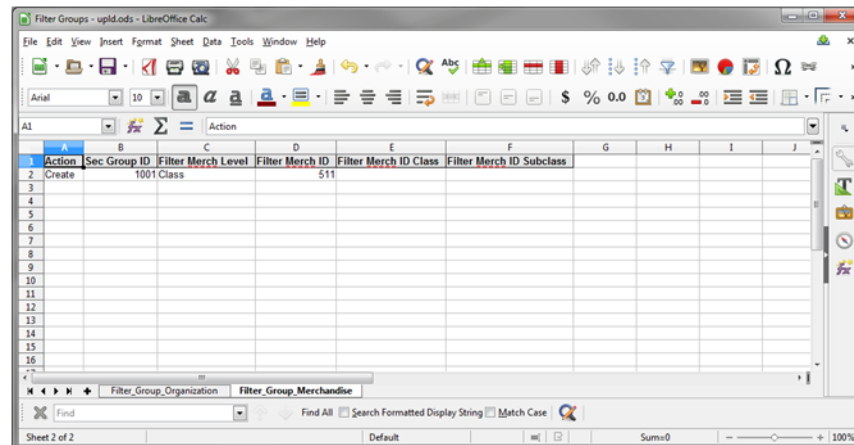
10. Save and close the file.

11. In the RMS Application, navigate to Foundation Data > Data Loading > Upload.

12. In the Upload Data screen, select:

- Template Type as 'Security'

- Template as 'Filter Groups'

- Enter new Process Description or retain as is

- Browse and select the Source file that was created in Step 10

Figure 3–33   Upload Filter Groups Information
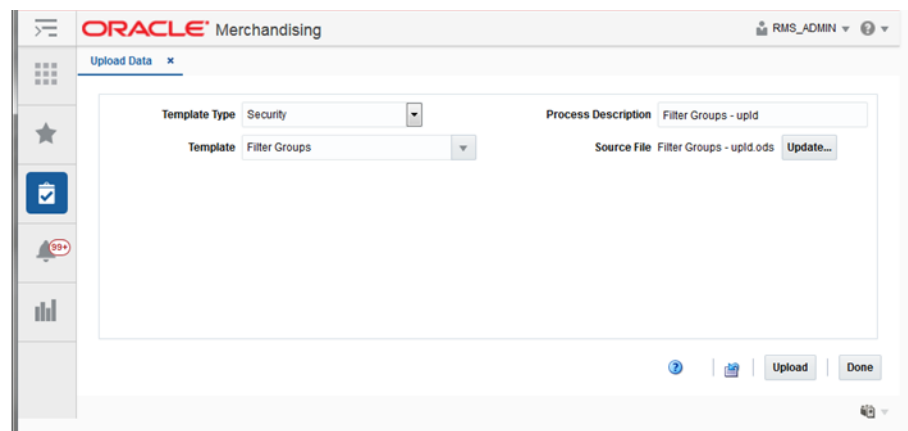
13. Click the **Upload** button.

Figure 3–34   Upload Filter Groups Information Complete
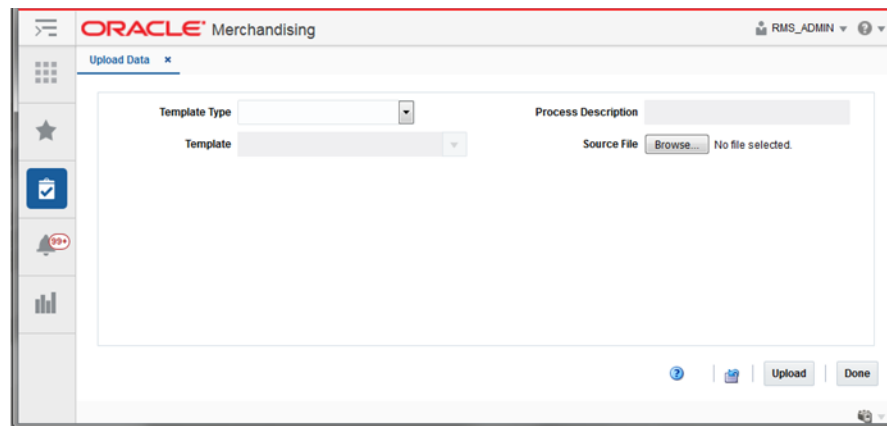
14. Click the **Done** button.

15. View the newly created Filter Groups mapping by downloading the Filter Groups spreadsheet (Steps 1 - 6).

Figure 3–35   View Newly Created Filter Groups

Figure 3–36   Newly Created Filter Groups
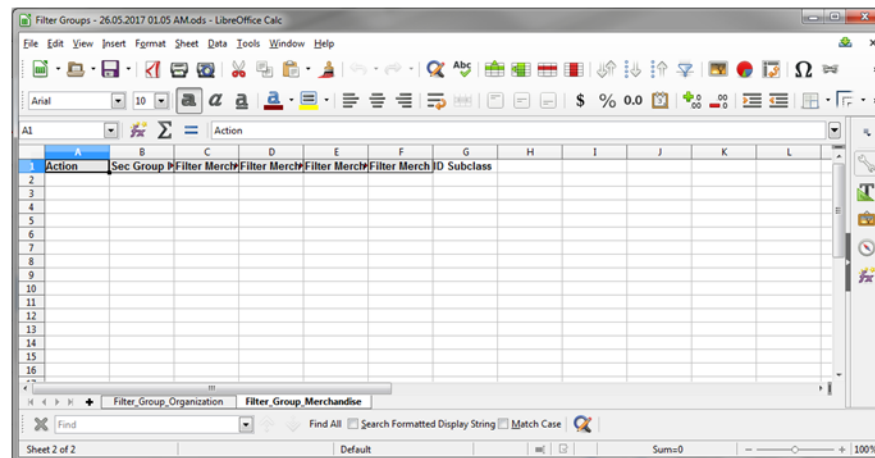
Setup Purchase Order Approval Limits

The setup of role privileges can be used to manage the size of a purchase order that a user can approve. The basis of this is either cost or retail based on the system option configuration. This setup is not required if user roles will not have an approval limit.

1. Log into the RMS Application.

2. Navigate to Foundation Data > Data Loading > Download.

3. In the Download Data screen, select:

   - Template Type as 'Security'

   - Template as 'Role Privileges'

Figure 3–37   Download Security Privileges Template Spreadsheet
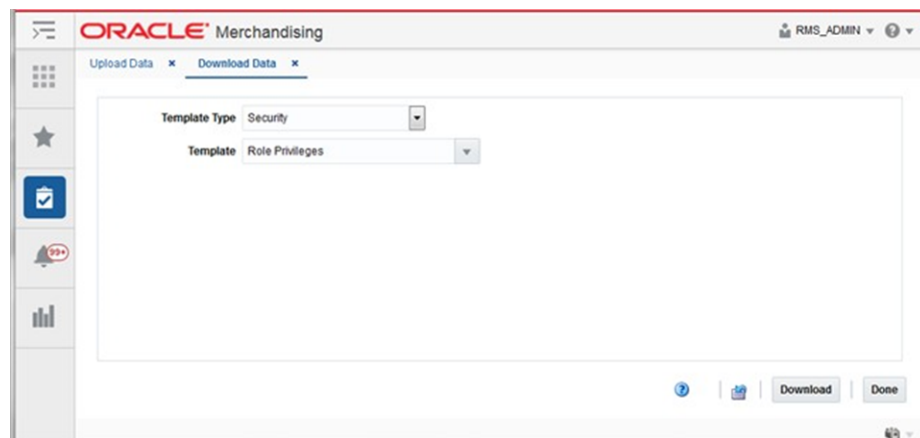
4. Click the **Download** button.

5. Save File to the local directory location when prompted.

Figure 3–38   Save File
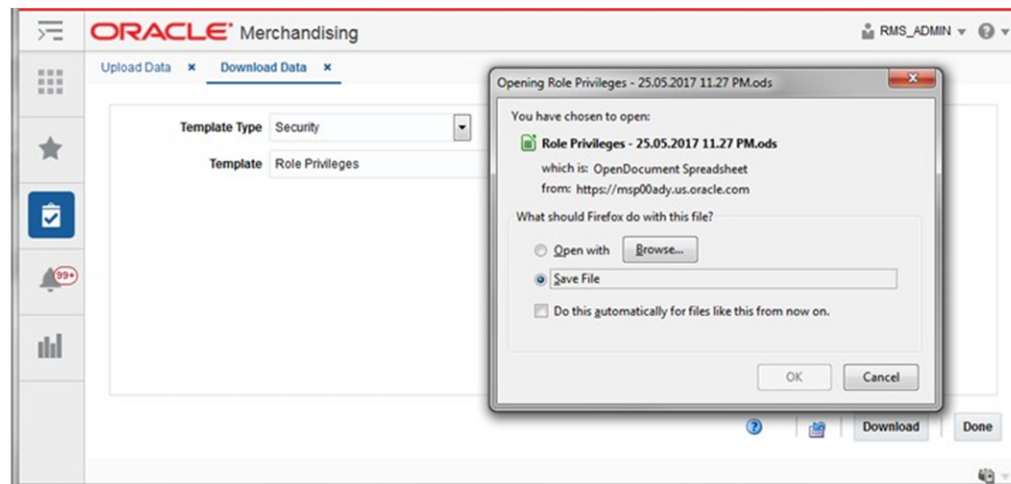
6. Click the **Done** button.

7. Open the downloaded file.

Figure 3–39   Open File
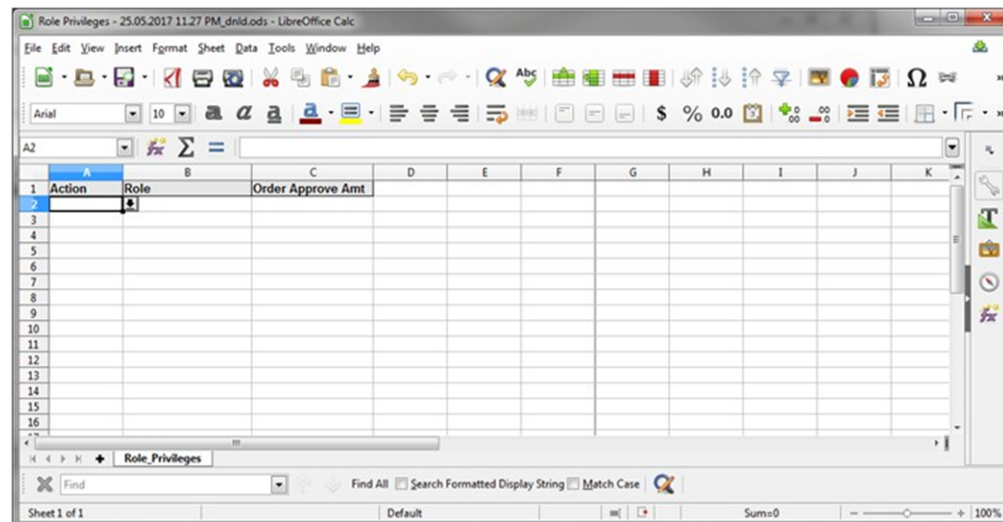
8. Save As < file name>.

Figure 3–40   Save as <File Name>
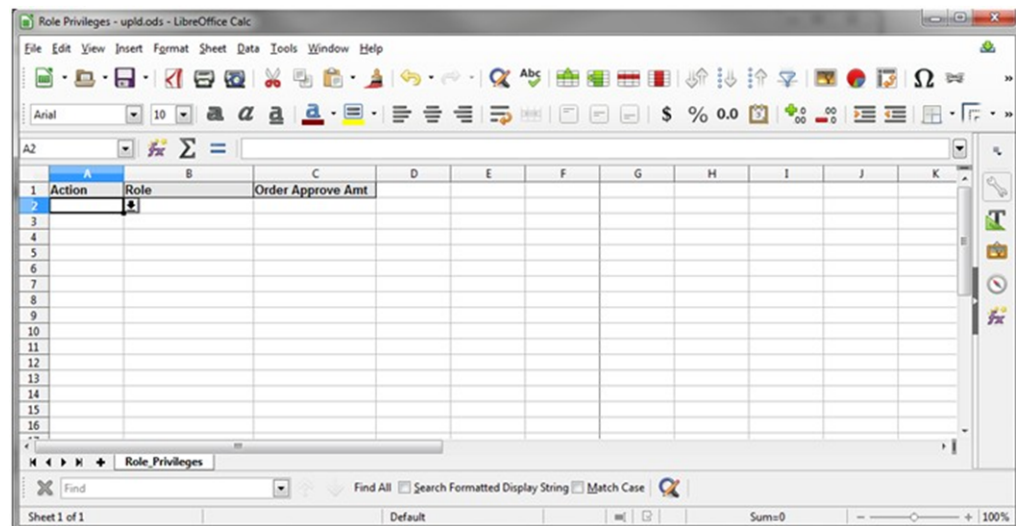
9. In the Role_Privileges tab, enter/select the following:

    ■   Action: 'Create'

    ■   Role: <Role>

    ■   Order Approve Amt: < Upper limit that the role will be able to approve on an order>

*Figure 3–41   Enter Role Privileges Information*
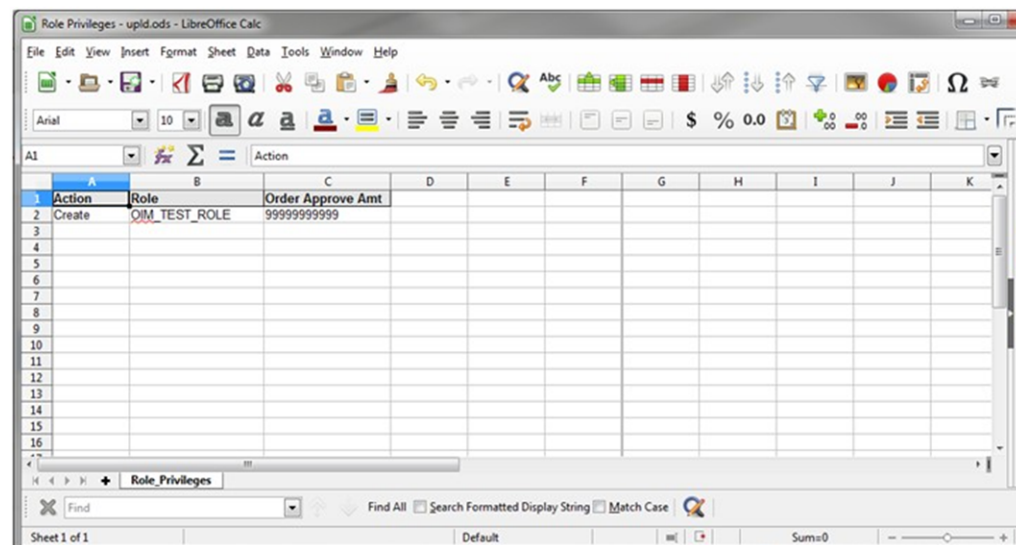
10. Save and Close the file.

11. In the RMS Application, navigate to Foundation Data > Data Loading > Upload.

12. In the Upload Data screen, select:

    ■   Template Type as 'Security'

    ■   Template as 'Role Privileges'

- Enter new Process Description or retain as is
- Browse and select the Source file that was created in Step 10

Figure 3–42   Upload Role Privileges Information

13. Click the **Upload** button.

Figure 3–43   Upload Role Privileges Information Complete

14. Click the **Done** button.

15. View the newly created Role Privileges by downloading the Role privileges spreadsheet (Steps 2 - 7).

Figure 3–44   View Newly Created Role Privilege

4

Application Customization

Application Customization is an option in the settings menu that can be used by an administrator to customize the different menus in the Merchandising solutions. The administrator can customize the task menu, reports menu, and the sidebar menu using the customization option.

Figure 4–1   Customization Menu
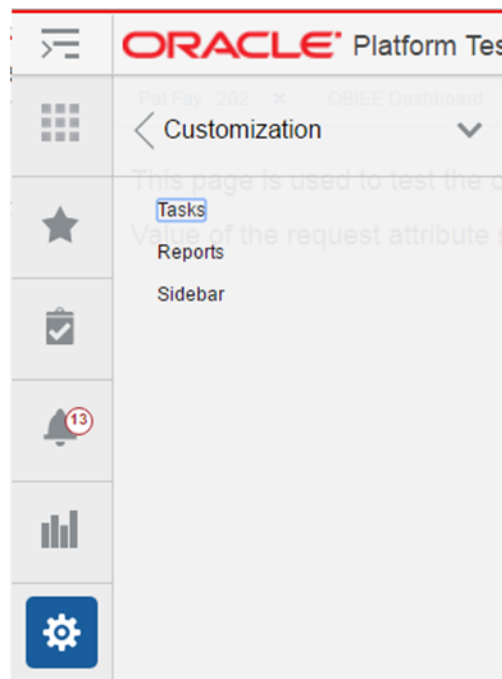
Task Menu Customization

The task menu model xml file uses the xml schema definition called NavigationModel.xsd for xml validation. The Navigation Model schema definition file is used to validate any changes done by the administrator on the task menu xml file. The Task Menu model file consists of a hierarchy of Item elements. Each element represents a menu item in the task menu. The Item element can either be a folder or has a URL, list of attributes and parameters that identify the task menu item.

Task Menu Model XML Items

An Item in the task menu model renders as either a launchable link or a submenu in the Tasks menu of the Retail Application. A new Item can be added to add a new link or submenu. An existing Item can be modified to change an existing link or submenu. Below is the example of a task menu model Item.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NavigationDefinition … >
    <Items>
        <Item id="myFolder" title="My Folder"  type="folder">
            <Items>

                <Item id="myContent" type="taskflow"
                      title="My Content">
                    <url>

/WEB-INF/mycompany/CustomDashboardTaskFlow.xml#CustomDashboardTaskFlow
                    </url>
                    <Parameters>
                        <Parameter id="productName">"Acme"</Parameter>
                    </Parameters>
                </Item>
            </Items>
        </Item>
    </Items>
</NavigationDefinition>
```

Task Menu Model Item Attributes

The following attributes apply to an Item element in the task menu model xml file.

Table 4–1   Task Menu Model Item Attributes

Attribute	Description
visible	Indicates if the item should be rendered (visible) or not. It can be an EL Expression that evaluates to true or false. Or it can be a plain string value equal to "true" or "false". Defaults to "true".
type	Indicates the type of the item. The main values are "folder","taskflow","link". The type "folder" indicates that the item contains additional sub-navigation items. The type "taskflow" indicates it is a task flow, and the type "link" is used for URLs.
title	The attribute title is used to provide the title of the sidebar item. This attribute is a required attribute.
targetTitle	The attribute targetTitle is used to provide the title of the tab in the content area. This attribute is an optional attribute. If not provided, the value of the title attribute will be used as the title of the tab in the content area.
accessKey	The attribute accessKey is used to specify the keyboard keystroke or a group of keystrokes that are used to access the navigation item using the keyboard.
shortDesc	The attribute shortDesc is used to provide the description of the navigation item which will be displayed when the user hovers over a menu item.

Table 4–1   (Cont.) Task Menu Model Item Attributes

Attribute	Description
showCloseIcon	Mainly used when the content is being rendered in a popup, and indicates if the 'x' icon should show or not on the top right corner of the popup to facilitate manual closing of the popup. Takes the values "true" or "false". Defaults to "false".
defaultContent	Indicates whether the corresponding items from the model show up by default in tabs in the content area when the page template loads. Takes values "true" or "false", or may even take an EL Expression evaluating to "true" or "false".
reuseInstance	The attribute reuseInstance is used to specify whether the navigation items with the same title and url will use the same tab or not. When this attribute is set to false, the request to open the content for that navigation item will always use a new tab. When the attribute is true, the navigation items with the same title and url will share the same tab in the content area. It defaults to true.
keysList	Takes name value pairs separated by a semicolon. The attribute keysList provides a way to differentiate two navigation items with the same title and url. If keysList is not provided, then the two navigation items with the same title and url will always use the same tab. When the keysList is provided, it provides uniqueness to the navigation items with the same title and url enabling them to use different tabs. Example keysList="key1=value1;key2=value2"
urlRendererHeight, urlRendererWidth	Example values are shown below. These used in the case of a popup and indicate the height and width of the popup. urlRendererHeight="200px" urlRender-erWidth="200px"
reloadTab	When this attribute is set to true, an already opened tab will be reloaded with the new input parameters for the taskflow. When it is set to false, a previously opened tab will only be re-focused, but not reloaded with new input parameters for the taskflow.
refreshOnDisclosure	When the navigation item has refreshOnDisclosure attribute set to true then the tab displaying that item in the Content Area will be refreshed every time it's disclosed. The attribute can take either of the two values true or false. Default is false. The attribute is useful in the scenarios where we want to display to the user the latest information from the database every time he/she comes back to the tab. The attribute should be used with caution because if the data in that tab is not committed before leaving the tab then the uncommitted data will be lost upon coming back to the tab.
dynamicTabFocus	When a navigation item is invoked, the tab displaying that item will have its text focused. To override this behavior, set dynamicTabFocus to "false". This attribute defaults to "true".
popupId	Applicable only when target="_popup". Must be a number between 1 and 15. This attribute allows consuming applications to target a specific popup within the UI Shell. The framework provides 15 popups that consuming applications can take advantage of. In case this attribute has not been specified, a default popup will be used by the framework. This default popup will not store its dimensions in MDS.

Table 4–1   (Cont.) Task Menu Model Item Attributes

Attribute	Description
popupContentHeight	Applicable only when target="_popup".
	This attribute is used to provide the height in pixels of the resulting popup dialog window.
popupContentWidth	Applicable only when target="_popup".
	This attribute is used to provide the width in pixels for the resulting popup dialog window.
popupStretchChildren	Applicable only when target="_popup".
	This attribute is used to indicate the stretching behavior for the contents of the resulting popup window.
	The contents are referred to as child components.
	Valid values are:
	■ none - does not attempt to stretch any children (the default value and value you need to use if you have more than a single child; also the value you need to use if the child does not support being stretched).
	■ first - stretches the first child (not to be used if you have multiple children as such usage will produce unreliable results; also not to be used if the child does not support being stretched).
popupResize	Applicable only when target="_popup".
	This attribute is used to indicate the resulting popup window's resizing behavior. Valid values are:
	■ on - user can resize the dialog with their mouse by dragging any of the dialog edges.
	■ off - the dialog automatically sizes its contents if popupStretchChildren is set to "none"
popupHelpTopicId	Applicable only when target="_popup".
	This attribute is used to look up a topic in a help-Provider for the resulting popup window.
popupShortDesc	Applicable only when target="_popup".
	This attribute is used to provide short description of the resulting popup window.
contentListener	Specifies the fully qualified name of the class im-plementing the ContentListener interface.
	This allows applications to have the ability to inject any session or request values before opening tabs.
tabShortDesc	Specifies the text to be shown when the user hovers over the application tab. Using this attribute application team can keep the title short and the tabShortDesc as fully qualified tab name which can be shown as the tooltip of the tab. This attribute will be displayed as tab title in screen reader mode.

Task Menu Model Item Sub-elements

The following are the sub-elements of the Item element in the task menu model.

Table 4–2   Task Menu Model Item Sub-elements

Element	Description
url	The location of the item being launched.
	If the type is "taskflow" - then the URL element must contain the path to the task flow XML.
	If the type is "link" - then the URL of the external system must be indicated in this subelement. The entire URL must be marked as character data (e.g. enclosed in CDATA).
Parameter	The <Parameters> sub-element within <Item> should list all the parameters to the dashboard page if there are any. Each parameter is represented as a <Parameter> element inside <Parameters>.
	The <Parameter> id should be the actual parameter reference name recognized by the dashboard URL.
	The value of each <Parameter> is a string value. This is the only supported data type.

Securing Access to Items

To restrict access to menu items to specific security roles, set the visible property on the <Item> element for the URL to an Expression Language (EL) expression that calls ADF's securityContext API's isUserInRole method.   Example:

```
<Item id="myDashboard1"
        type="link"
        title="Profitability Dashboard"
        visible="#{securityContext.isUserInRole['BUYER_JOB']} ">
```

The parameter to the securiytContext.isUserInRole method is a logical security role that is configured for the Retail Application. The API returns true if the user is included in the specified security role. If the user is not authenticated or is not found in the role,   the API returns false.

Reports Menu Customization

The reports menu uses the same NavigationModel.xsd file used by the task menu. The attributes defined for the task menu apply to the reports menu as well. Please refer to the Task Menu Customization section above for the details on how to customize the reports menu.

Sidebar Menu Customization

The sidebar menu uses a schema called sidebarSchema.xsd for the validation of the sidebar items. The sidebar xml model file consists of Item elements similar to the Item element in the task menu. The Item element in the sidebar xml model has a different set of attributes.

Sidebar Menu Model XML Items

An Item element in the sidebar model xml file renders as a launchable icon in the sidebar. The sidebar has a fixed set of icons and a variable set of icons. Only the variable icons can be customized using the sidebar model xml file. The fixed set of icons cannot be customized. A new sidebar Item can be added to add a new icon on the sidebar. Below is the example of a sidebar menu model Item.

```
<?xml version="1.0" encoding="UTF-8" ?>
<SidebarModel id="raf_sidebar_items_root"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="urn:www.oracle.com:oracle.retail.apps.framework.sidebar
classpath:oracle/retail/apps/framework/uishell/common/sidebar/sidebarSchema.xsd"
            xmlns="urn:www.oracle.com:oracle.retail.apps.framework.sidebar">
    <Items>
        <Item id="quickCreateSidebarItem" title="Quick Create"
            iconUrl="/raf/retail-core/images/quick-create-gray.png"
            selectedIconUrl="/raf/retail-core/images/quick-create.png">

<taskFlow>/WEB-INF/oracle/retail/apps/framework/uishell/navigation/contentarea/flo
w/TestQuickCreateFlow.xml#TestQuickCreateFlow</taskFlow>
        </Item>
    </Items>
</SidebarModel>
```

Sidebar Menu Model Item Attributes

The following attributes apply to an Item element in the sidebar menu model xml file.

Table 4–3   Sidebar Menu Model Item Attributes

Attribute	Description
id	The id of the sidebar item.
title	The title of the sidebar icon.
visible	The visible attribute controls the visibility of the sidebar item.
iconUrl	The icon that is displayed for the sidebar item.
selectedIconUrl	The icon that is displayed when the sidebar item is selected.
disabledIconUrl	The icon that is displayed when the sidebar item is disabled.
hoverIconUrl	The icon that is displayed when the sidebar item is hovered on.
depressedIconUrl	The icon that is displayed when the sidebar item is depressed.

Sidebar Model Item Sub-elements

The following are the sub-elements of the Item element in the sidebar model.

Table 4–4   Sidebar Model Item Sub-elements

Element	Description
taskFlow	The URL of the taskflow being launched.
Parameter	The <Parameters> sub-element within <Item> should list all the parameters to the sidebar taskflow if there are any. Each parameter is represented as a <Parameter> element inside <Parameters>.
The <Parameter> id should be the actual parameter reference name recognized by the taskflow URL.
The value of each <Parameter> is a string value.   This is the only supported data type.

Application Configuration

Oracle Retail Application Configuration is an option in the settings menu that can be used by an administrator to configure the dashboards and contextual reports and manage application properties. The administrator can add or replace reports, change the dashboard and report layout using the configuration option.

Figure 4–2   Configuration Menu
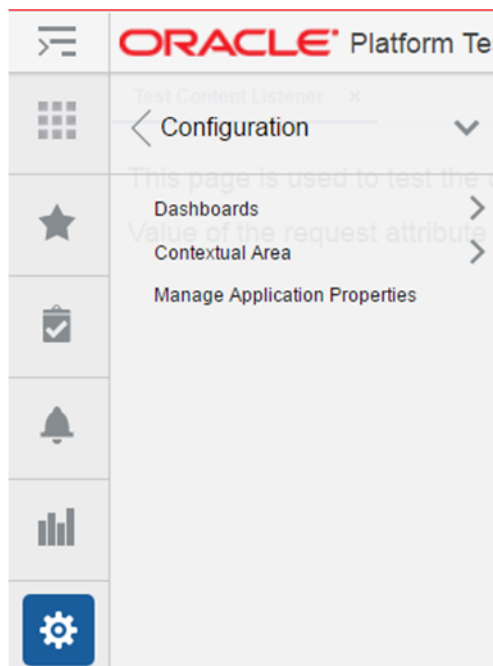
Dashboard Customization

Retail Dashboards display the status of metrics and key performance indicators relevant to the Retail Application. They are typically tailored for specific roles and they allow users to easily monitor the status of the current data within the application.

The Dashboard link displays the list of dashboards available in the application. When a dashboard name is clicked, it opens the dashboard xml file in an application tab. The dashboard reports can be added, modified or removed from the dashboard by changing the dashboard xml file. The dashboard layout can also be changed using the dashboard xml file.

Figure 4–3   Dashboards Menu
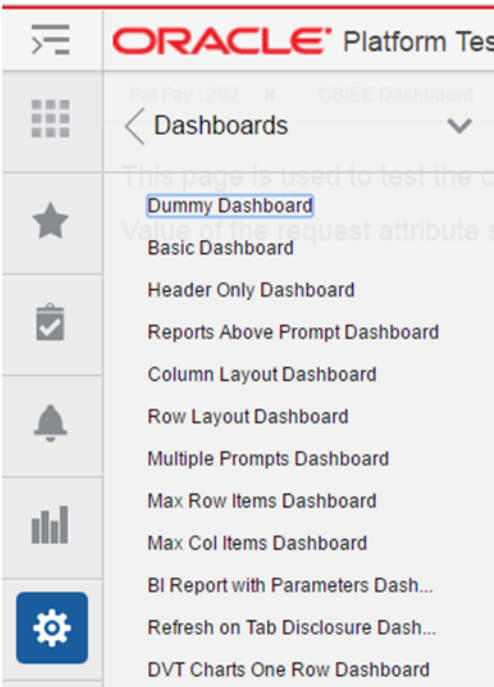
Understanding Dashboard Design Patterns

Depicted below is an example of a dashboard that might appear in a Retail Application:

Figure 4–4   Dashboard Example
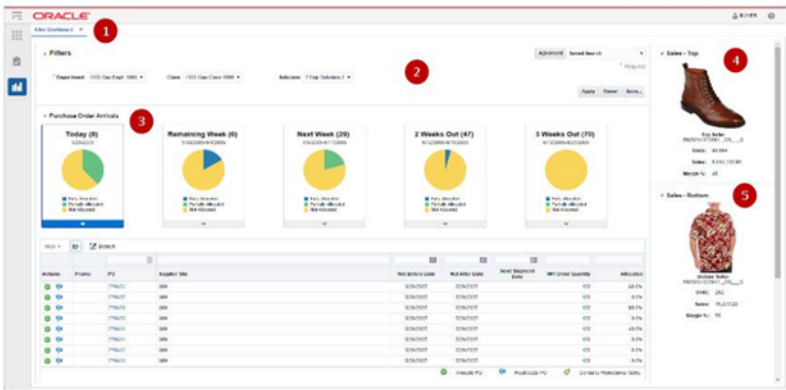
Table 4–5   Dashboard Region Descriptions

Number	Region	Description
1	Dashboard Page	A page rendering a dashboard. A dashboard is composed of 1 or more reports. The dashboard page is launched as a tab within the Retail Application

Table 4–5   (Cont.) Dashboard Region Descriptions

Number	Region	Description
2	Prompt Region	The prompt region contains a set of editable input components that allow users to modify the parameters of the various reports in the dashboard. This allows users to see different sets of data for the same set of reports.
3	Report Region	A report region shows a graphical view of the application's business data.

The dashboard page, prompts and reports are implemented as ADF bounded task Flows and page fragments. The prompts and regions are configured in an XML file called a Dashboard Prompt Configuration XML.   The dashboard prompt XML arranges visually the prompts and regions on the dashboard page.

The representation of the dashboards is depicted in the diagram below:

Figure 4–5   Representation of a Dashboard
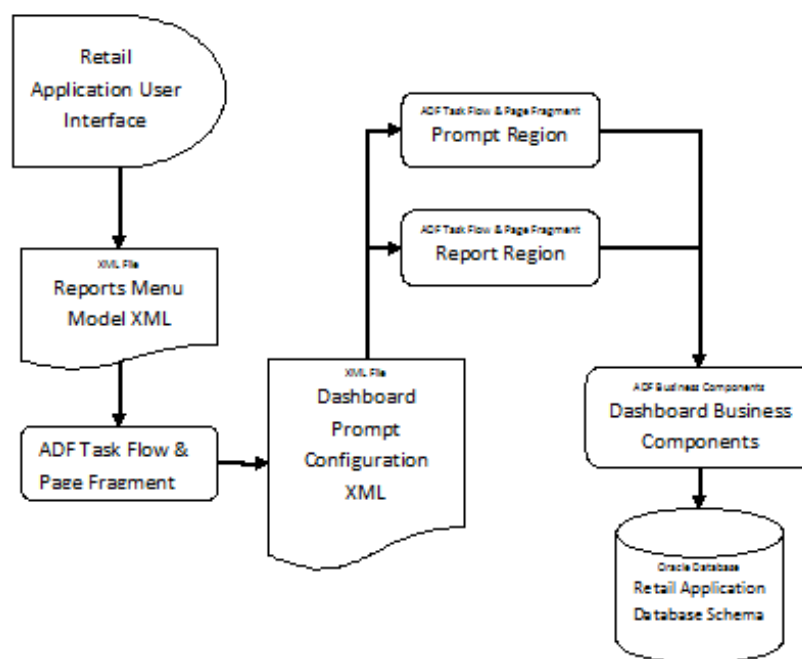
The dashboard task flow is launchable from the Retail Application User Interface's Reports Menu.   In order to make the report launchable, the dashboard task flow is registered in the application's reports menu model XML.

 A set of ADF business components enable data from the Retail Application schema to be rendered in the report regions.

The figure below shows a rendering of a dashboard (called Allocations Dashboard) in a Retail Application:

Figure 4–6   Example Allocations Dashboard
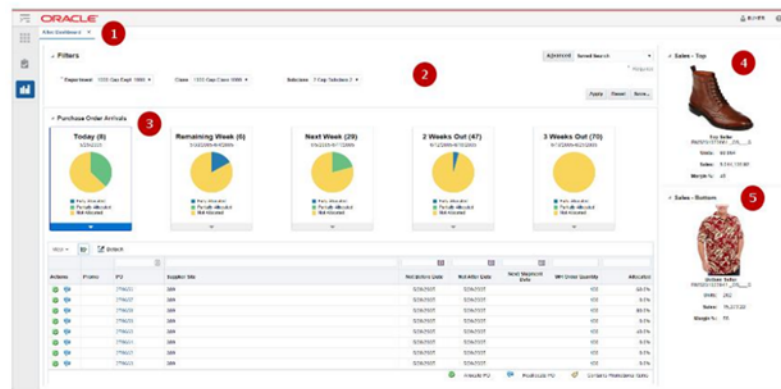
The different parts of this example dashboard are described below:

Table 4–6   Allocation Dashboard Region Descriptions

Number	Region	Description
1	Dashboard Page	The Allocations Dashboard example contains 1 prompt and 3 reports. It is implemented as an ADF bounded task flow called AllocDashboardFlow.
2	Prompt Region	This is the prompt region that users modify to change the view of the various reports in this dashboard. It is implemented as an ADF bounded task flow called DashboardFilterFlow.
3	Order Status Report	A report region showing a row of information tiles and a detail table. It is implemented as an ADF bounded task flow called OrderStatusFlow.
4	Top Seller Report	A report region showing the top selling item. It is implemented as an ADF bounded task flow called TopSellerFlow.
5	Bottom Seller Report	A report region showing the bottom sell-ing item. It is implemented as an ADF bounded task flow called BottomSellerFlow.

The reports menu model XML file (as shown below) contains an item called allocDashboard registering the task flow, AllocDashboardFlow, into the reports menu. Note that the dashboard task flow includes a parameter called dashboardConfigXML which references the location of the Dashboard Prompt Configuration XML file.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NavigationDefinition id="Folder_1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="urn:www.oracle.com:oracle.retail.apps.framework.navigation"
    xsi:schemaLocation="urn:www.oracle.com:oracle.retail.apps.framework.navigation
classpath:oracle/retail/apps/framework/uishell/navigation/model/schema/NavigationM
odel.xsd">
    <Items>
```

```
                <Item id="allocDashboard" title="Alloc Dashboard"
                    shortDesc="Alloc Dashboard"
                    type="taskflow">
                <url>/WEB-INF/AllocDashboardFlow.xml#AllocDashboardFlow</url>
                <Parameters>
                        <Parameter
id="dashboardConfigXML">/oracle/AllocDashboardPrompt.xml</Parameter>
                <Parameters>
            </Item>
        </Items>
</NavigationDefinition>
```

The Dashboard Prompt Configuration XML file for this example is called the
AllocDashboardPrompt.xml and it contains the following entries:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Dashboard layout="column"
xmlns="urn:www.oracle.com:oracle.retail.apps.framework.dashboard"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="urn:www.oracle.com:oracle.retail.apps.framework.dashboard
classpath:oracle/retail/apps/framework/dashboard/model/schema/dashboardSchema.xsd"
>
    <Vectors>
        <Vector>
            <Items>
                <Item id="filter" type="prompt">

<url>/WEB-INF/DashboardFilterFlow.xml#DashboardFilterFlow</url>
                </Item>
                <Item id="orderStatus" type="report">

<url>/WEB-INF/oracle/retail/apps/framework/dvtcontextawarereport/publicui/flow/DVT
ContextAwareReportFlow.xml#DVTContextAwareReportFlow</url>
                    <Parameters>
                        <Parameter
id="taskflowURL">/WEB-INF/OrderStatusFlow.xml#OrderStatusFlow</Parameter>
                        <Parameter id="parameterName1">departmentIds</Parameter>
                        <Parameter id="payloadKeyName1">DepartmentId</Parameter>
                        <Parameter id="parameterName2">classIds</Parameter>
                        <Parameter id="payloadKeyName2">ClassId</Parameter>
                        <Parameter id="parameterName3">subclassIds</Parameter>
                        <Parameter id="payloadKeyName3">SubclassId</Parameter>
                    </Parameters>
                </Item>
            </Items>
        </Vector>
        <Vector width="300px">
            <Items>
                <Item id="topSeller" type="report">

<url>/WEB-INF/oracle/retail/apps/framework/dvtcontextawarereport/publicui/flow/DVT
ContextAwareReportFlow.xml#DVTContextAwareReportFlow</url>
                    <Parameters>
                        <Parameter
id="taskflowURL">/WEB-INF/oracle/retail/apps/flow/TopSellerFlow.xml#TopSellerFlow<
/Parameter>
                        <Parameter id="parameterName1">departmentIds</Parameter>
                        <Parameter id="payloadKeyName1">DepartmentId</Parameter>
                        <Parameter id="parameterName2">classIds</Parameter>
                        <Parameter id="payloadKeyName2">ClassId</Parameter>
                        <Parameter id="parameterName3">subclassIds</Parameter>
```

```
                                <Parameter id="payloadKeyName3">SubclassId</Parameter>
                            </Parameters>
                        </Item>
                        <Item id="bottomSeller" type="report">

<url>/WEB-INF/oracle/retail/apps/framework/dvtcontextawarereport/publicui/flow/DVT
ContextAwareReportFlow.xml#DVTContextAwareReportFlow</url>
                            <Parameters>
                                <Parameter
id="taskflowURL">/WEB-INF/oracle/retail/apps/flow/BottomSellerFlow.xml#BottomSelle
rFlow</Parameter>
                                <Parameter id="parameterName1">departmentIds</Parameter>
                                <Parameter id="payloadKeyName1">DepartmentId</Parameter>
                                <Parameter id="parameterName2">classIds</Parameter>
                                <Parameter id="payloadKeyName2">ClassId</Parameter>
                                <Parameter id="parameterName3">subclassIds</Parameter>
                                <Parameter id="payloadKeyName3">SubclassId</Parameter>
                            </Parameters>
                        </Item>
                    </Items>
                </Vector>
            </Vectors>
</Dashboard>
```

The file registers the prompts and regions into a hierarchy of vectors and items. The structure of the dashboard prompt configuration XML file is explained in the following section.

Dashboard Prompt Configuration

The Dashboard Prompt Configuration XML file can be modified to change the dashboard layout or add or remove reports from the Dashboard. The root element of the model file is the "Dashboard" element. Other than schema and namespace information, the "Dashboard" element has a "layout" attribute that must be provided, and must contain a value of either "row" or "column". This attribute arranges the regions of the dashboard in rows or columns.

Inside the "Dashboard" element is a "Vectors" element, which contains a list of "Vector" elements. If the dashboard layout="row", then each "Vector" corresponds to a single row in the dashboard. Conversely, if layout="column", then each "Vector" corresponds to a column. To add a new row or column, simply add another "Vector" element inside the "Vectors" element.

Up to 12 rows and 6 columns can be displayed in a dashboard, not including separate header and footer regions. Any items beyond 12 rows or 6 columns will not appear on the dashboard.

Rows in the dashboard display from top to bottom. If layout="row", the first "Vector" will appear toward the top of the page. Columns appear from left to right, so if layout="column", the first "Vector" will appear toward the left of the page.

Each "Vector" element contains an "Items" element, which contains a list of "Item" elements. The xml structure for "Item" elements is very similar to the tasks or reports sidebar menu models.

The following attributes are supported on dashboard items:

Table 4–7   Supported Attributes

Name	Description	Required
id	A unique id for this region	Yes
type	The type of content contained in this region. Valid values: prompt, report	Yes
width	The width of this region. See below for more details.	No
height	The height of a region. See below for more details.	No
url	The task flow url to open in this region	Yes
Parameters	Parameters to be sent to the URL	No

In addition to the items arranged in rows or columns, you can add a header and footer region to the dashboard. Both horizontally span the full width of the dashboard, with the header appearing at the top and the footer appearing at the bottom. These can be added in the model by specifying a "HeaderItem" or "FooterItem" child of the root "Dashboard" element. These regions support all the same properties as other items in the dashboard.

Vectors and items expose width and height attributes to enable applications to resize elements to best fit on the dashboard. Certain rules and best practices apply when using these attributes.

First, if the dashboard is in row layout, then all items in a row must have the same height. The height attribute from the vector is used to specify the height of every item in that row. Since rows are assumed to span the full width, the vector width is ignored in this layout.

In the same way, when the dashboard is arranged in column layout, all items in a column must have the same width. The vector width is used to specify the width of every item in that column. Columns should expand vertically to accommodate all content, allowing a scrollbar if necessary. Therefore, vector height is ignored in this layout.

In addition to the width and height attributes, a vector also provides a dimensionsFrom attribute. The valid values of the dimensionsFrom attribute can be auto, child or parent. The dimensionsFrom attribute is used to specify whether the Items in the vector inherit their dimensions from the parent decorative box or not. When the dimensionsFrom attribute is parent, the Items in the vector will be stretched to fill any available space in the vector. When the dimensionsFrom attribute is child, the Items in the vector won't stretch but will display scroll bars if they cannot be accommodated in the available space in the decorative box. The default value of the dimensionsFrom attribute is 'auto' which gives preference to the child dimensions.

The following table indicates which attributes are used, based on which dashboard layout has been selected.

Table 4–8   Supported Attributes

	Row layout	Column layout
Vector width	ignored	valid
Vector height	valid	ignored
Item height	ignored	valid
Item width	valid	ignored

In column In column layout, either every vector width should be provided, or every width should be omitted. If all widths are omitted, columns default to divide the total width evenly. When the widths are omitted, the dimensionsFrom attribute of the vector can be set to parent if stretching of the Items is desired. If widths are provided, they should be specified as a percentage, and the sum of all widths should equal 100%.

In row layout, heights may be specified or omitted as desired. If a row's height is omitted, it will default to get the dimension from the content inside the regions. When the row's height is omitted, the dimensionsFrom attribute of the vector can be set to parent if stretching of the Items is desired. Heights should be specified using an exact unit such as px or em. Percentages should not be used.

If these rules and best practices are ignored, results may be inconsistent or undesirable. Items may not size as intended, and horizontal scrollbars or nested vertical scrollbars may appear.

In addition to resizing entire rows or columns, individual items inside the row or column can also be resized. As mentioned above, items in a row should all have the same height, so in row layout, item height is ignored and only item width is valid. And items in a column should have the same width, so item width is ignored and only item height is valid. See the table above for details.

The same rules and best practices that apply to vector height and width also apply to setting item height and width.

Refreshing Reports on Prompt Changes

When the user changes a value on the dashboard prompt, that value translates as a parameter change on the different reports on the dashboard.

This is accomplished through a feature in ADF called Contextual Events. To put simply, contextual events are signals that a page in a task flow can generate for which pages in other task flows can listen and react to. For the included dashboards, prompt regions generate a contextual event whenever the user changes a prompt field value. When a report "hears" an event, it will process the contextual event information (called a payload) to extract the changed prompt value, use that value as a parameter to the report query and refresh the report region.

In the example depicted in the previous section, the Dashboard Prompt Configuration XML file shows all three reports registered under a wrapper task flow called the DVTContextAwareReportFlow.

```
            <Item id="bottomSeller" type="report">

<url>/WEB-INF/oracle/retail/apps/framework/dvtcontextawarereport/publicui/flow/DVT
ContextAwareReportFlow.xml#DVTContextAwareReportFlow</url>
                <Parameters>
                    <Parameter
id="taskflowURL">/WEB-INF/oracle/retail/apps/flow/BottomSellerFlow.xml#BottomSelle
rFlow</Parameter>
                    <Parameter id="parameterName1">departmentIds</Parameter>
                    <Parameter id="payloadKeyName1">DepartmentId</Parameter>
                    <Parameter id="parameterName2">classIds</Parameter>
                    <Parameter id="payloadKeyName2">ClassId</Parameter>
                    <Parameter id="parameterName3">subclassIds</Parameter>
                    <Parameter id="payloadKeyName3">SubclassId</Parameter>
                </Parameters>
            </Item>
```

The DVTContextAwareReportFlow wrapper task flow enables a report to listen to contextual events generated by prompt regions.

Note the pairs of parameterNameN and payloadKeyNameN registered in the <Parameters> section for the report task flow BottomSellerFlow. A pair of parameter name and payload key name tells the framework what prompt value is mapped to a specific report parameter.   From the example above, parameterName1 is specified as "departmentIds" and payloadKeyName1 as "departmentIds". This means the following:

- There is a task flow parameter for the BottomSellerFlow called "departmentIds"

- When the user changes the value of the department ID in the prompt region, the system will generate a contextual event with the new department value stored as reference "DepartmentId".

- The report is context aware (as indicated by the wrapper task flow - DVTContextAwareReportFlow).   When the report detects the contextual event, the framework will get the new department ID value from the payload by using the key "DepartmentId", use it to update the BottomSellerFlow task flow's "departmentId" parameter, and refresh the report region.

Adding or Replacing a Report

This section discusses the steps needed to add or replace a report on an included dashboard.

1. In the Settings Menu, click on the Dashboard folder and then click on the name of the Dashboard.

2. Note the new report's task flow URL and parameters.

3. Modify the Dashboard Prompt Configuration XML file to add an item or replace an existing item for the new report.

4. If the report has to refresh when a prompt on the dashboard is changed by the user, make sure that report task flow is wrapped in a DVTContextAwareTaskFlow for its item entry in the dashboard prompt configuration XML file.

5. Test the Dashboard.

Removing a Report

This section discusses the steps needed to remove a report from an included dashboard.

1. In the Settings Menu, click on the Dashboard folder and then click on the name of the Dashboard.

2. Modify the dashboard prompt configuration XML file and remove the report. You may need to readjust the position of the other reports.

3. Test the Dashboard.

Changing a Dashboard Layout

This section discusses the steps needed to change the layout of the reports in included dashboards.

1. In the Settings Menu, click on the Dashboard folder and then click on the name of the Dashboard.

2. Modify the dashboard prompt configuration XML file and rearrange the position of the item entries. Save the dashboard configuration file.

3. Reopen and test the Dashboard.

Contextual Area Customization

Contextual Reports are reports that appear in a task flow's contextual area section. Contextual Area is a collapsible section to the right of the local area that provides a space to present information that can assist users in completing their tasks.

Since, the information presented in the contextual area is dependent on the task or workflow the user is on, contextual areas are associated to task flows and there can be only one contextual area per task flow.

Within a contextual area - multiple contextual reports can be configured.

Each contextual report can change its contents depending on the action being performed in the user's current workflow.

For example, in the screenshot below, as the user selects the line of business from the table, the line chart will show the annual trade for the selected line of business.

Figure 4–7   Contextual Area Example
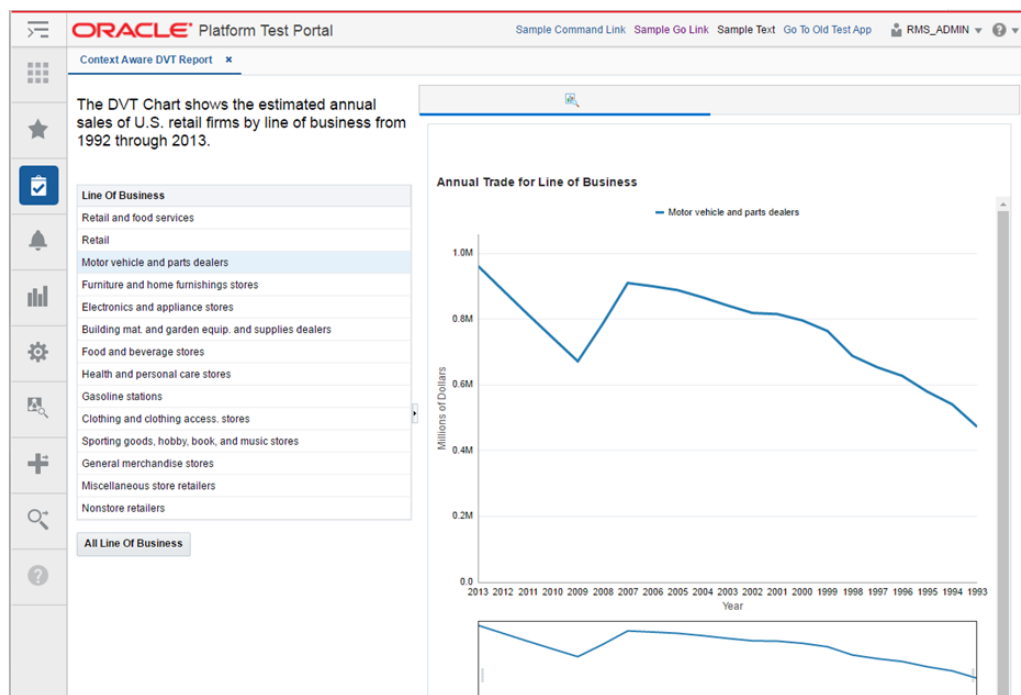
Each task flow publishes contextual business events on key activities happening in the screen.   Contextual reports can listen to those events and change its content depending on the payload information associated with the event.

Contextual reports can either be ADF based DVT reports or can be maintained in a separate BI reporting tool. The ADF based DVT reports are ADF taskflows, which are built using the ADF Data Visualization components. The reports maintained in an external reporting tool are URLs accessible in a web browser via a URL. Oracle Retail recommends using Oracle Business Intelligence Enterprise Edition (Oracle BI EE) for an external reporting tool.

The contextual reports are added into a Retail Application by modifying the task flow's contextual area model XML file.

The Contextual Area link displays the list of contextual reports in the application. When a contextual report name is clicked, it opens the contextual report xml file in an application tab. The contextual reports can be added, modified or removed from the contextual area by changing the contextual reports xml file. The contextual reports layout can also be changed using the contextual reports xml file.

Figure 4–8   Contextual Area Menu
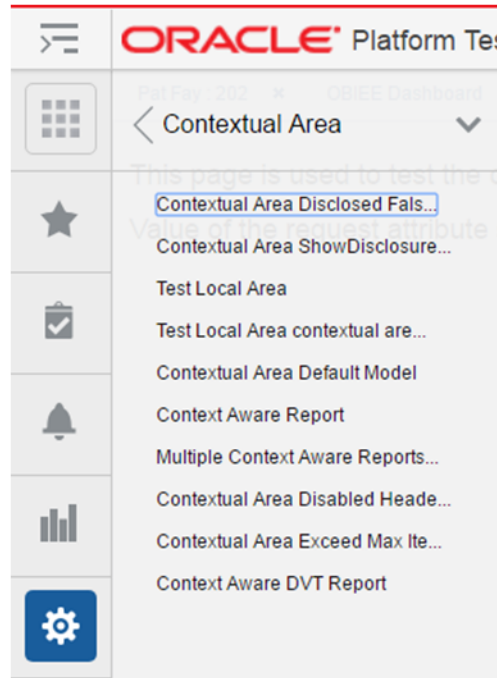
Adding a URL based Contextual Report

Adding a contextual report to a task flow entails the modification of the task flow's Contextual Area Model XML file. Multiple reports can be added to the model.   Each report is rendered in a collapsible panel box.

Before adding a URL based contextual report, the retailer must have:

- Obtained information about the Retail Application's list of contextual business events that can be broadcast from various workflows. Please refer to the application operations guide for the list of business events.

- Created one or more contextual BI reports in the BI reporting tool (e.g. Oracle BI EE).

- The web URL for each report must be available in order to proceed with the steps in this section.

- Any parameters to configure the content of the report must be known and should be accessible as parameters to the dashboard's URL.

Once the above pre-requisites have been satisfied, proceed with the following steps:

1. In the Settings Menu, click on the Contextual Area folder and then click on the Contextual Area taskflow name.

2. This will open the taskflow's contextual area model xml file.

3. Add an <Item> element within the topmost <Items> element that references the task flow called ViewContextAwareReportFlow.   The ViewContextAwareReportFlow is a framework for rendering URL based reports that will be aware of contextual business events emanating from the Retail Application task flows.

4. Make sure that the <Item> id is unique. Make sure the <Item> type is "taskflow". Provide a meaningful title.

5. Populate the parameters to the ViewContextAwareReportFlow by adding the following <Parameter>/<Parameters> elements.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NavigationDefinition … >
    <Items>

        <Item id="showCustomerMetric"
              type="taskflow"
              title="Customer Metric">
            <url>
/WEB-INF/oracle/retail/apps/framework/contextawarereport/publicui/flow/ViewCont
extAwareReportFlow.xml#ViewContextAwareReportFlow
            </url>

            <Parameters>
                <Parameter id="reportDescription">Item Metric</Parameter>
                <Parameter
id="actionType">AllocMaintItemSelectedEvent</Parameter>
                <Parameter id="primaryUrl">
<![CDATA[/faces/oracle/retail/apps/framework/contextawarereport/publicui/page/V
iewContextAwareReportTestPage.jspx?paramItemId=<selectedItemId>&paramItemType=<
selectedItemType:token01>&paramLanguage=<language>]]>
                </Parameter>
                <Parameter id="token01">regular</Parameter>
            </Parameters>

        </Item>

    </Items>
</NavigationDefinition>
```

2. This will open the taskflow's contextual area model xml file.

3. Add an <Item> element within the topmost <Items> element that references the task flow called DVTContextAwareReportFlow.   The DVTContextAwareReportFlow is a framework for rendering ADF DVT based reports that will be aware of contextual business events emanating from the Retail Application task flows.

4. Make sure that the <Item> id is unique. Make sure the <Item> type is "taskflow". Provide a meaningful title.

5. Populate the parameters to the DVTContextAwareReportFlow by adding the following <Parameter>/<Parameters> elements.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<NavigationDefinition … >
    <Items>

        <Item id="showCustomerMetric"
              type="taskflow"
              title="Customer Metric">
            <url>
/WEB-INF/oracle/retail/apps/framework/dvtcontextawarereport/publicui/flow/DVTCo
ntextAwareReportFlow.xml#DVTContextAwareReportFlow
            </url>
            <Parameters>
                <Parameter
id="taskflowURL">/WEB-INF/oracle/retail/apps/framework/uishell/navigation/conte
xtualarea/flow/ItemMetricFlow.xml#ItemMetricFlow</Parameter>
                <Parameter id="parameterName1">itemId</Parameter>
                <Parameter id="payloadKeyName1">ItemId</Parameter>
                <Parameter
id="parameterValue1">#{payload.valueMap['ItemId']}</Parameter>
            <Parameter id="refreshOnDisclosure">#{true}</Parameter>
            </Parameters>
        </Item>

    </Items>
</NavigationDefinition>
```

Figure 4–9   Manage Application Properties Settings Menu
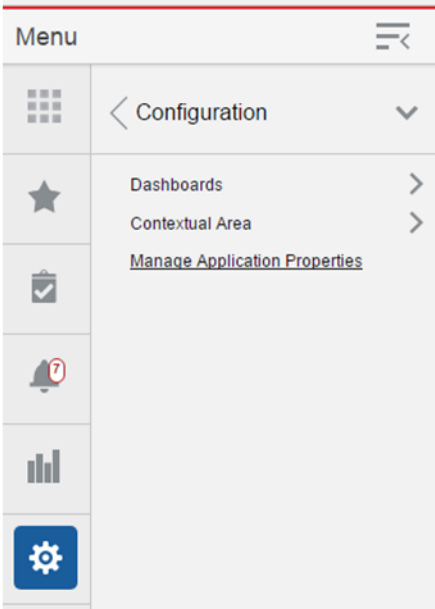
The Manage Application Properties screen allows administrators to search for configuration properties by deployment (a required criteria). Properties can also be searched by keywords against key names and descriptions.

Figure 4–10   Manage Application Properties Query Panel

Properties matching the search criteria are displayed in a table from which a user can perform the following actions:

- View details of a property

- Edit a property

- Export properties to CSV

Figure 4–11   Manage Application Properties View Dialog
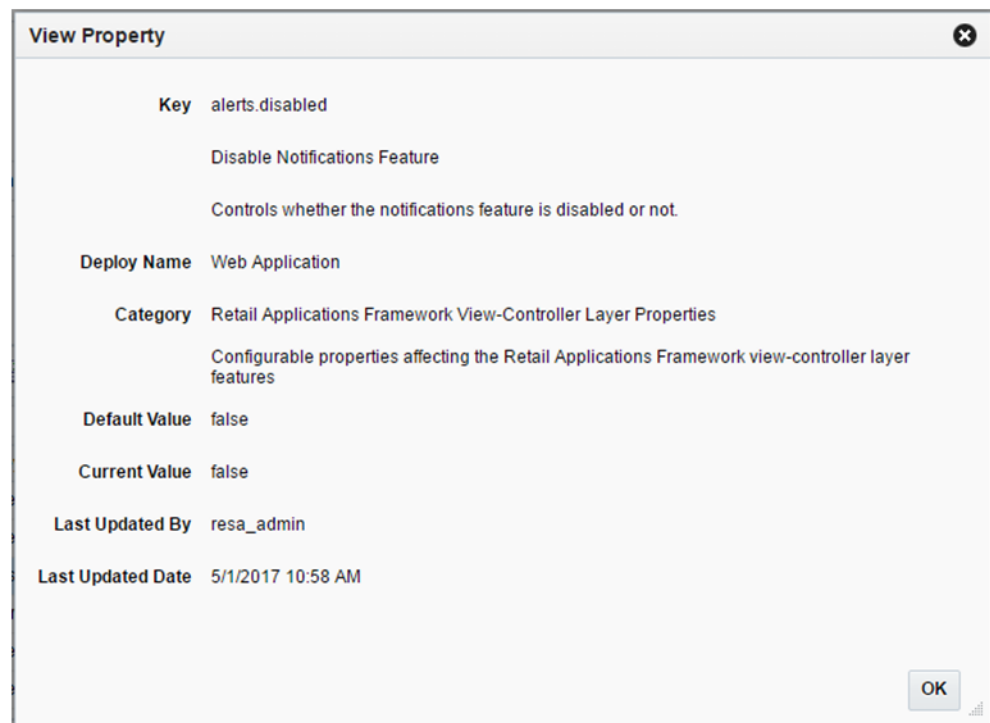
Figure 4–12   Manage Application Properties Edit Dialog
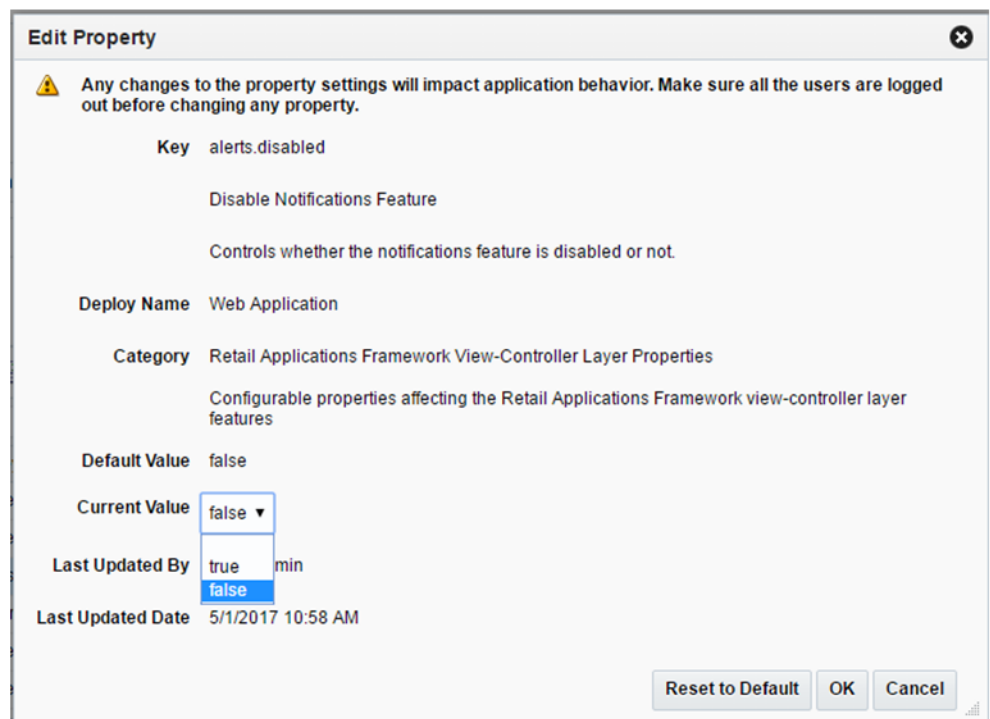
Important Considerations

This section discusses important points administrators must consider before updating configuration properties.

Recommended Time to Update Configurations

Any changes to the configuration properties will impact application behavior. It is recommended that administrators perform configuration updates during downtimes when no users are in the system performing critical business workflows.

Effectivity of Updates

- Updates to configuration against the main Retail web application are effective immediately. However, users will need to be advised to log-out and log back into the system to see the effects of the configuration change.

- Updates to configuration against other deployments such as REST Services, calculation engines, etc are not effective immediately. It can take up to 1 hour before the updates are realized.

- A server restart will guarantee the effectivity of any property updates regardless of the type of deployment.

Administration Overview

Oracle Retail Application Administration is an option in the Settings menu that can be used by an administrator to manage security policies, application navigator, notifications, resource text, Asynchronous Task Status and Manage Lookups. All the links in the administration menu except for the Manage Asynchronous Task Status and Manage Lookups links open the tasks in a new browser window using the Oracle Retail Application Administration Console (RAAC) application. The Manage Asynchronous Task Status and Manage Lookups functions exists within the application's realm and not the RAAC realm.

The following options are available in the Administration menu:

- Manage Security Policies
- Manage Application Navigator
- Manage Notifications
- Resource Bundles
- Manage Asynchronous Task Status
- Manage Lookups

Figure 4–13   Administration Menu
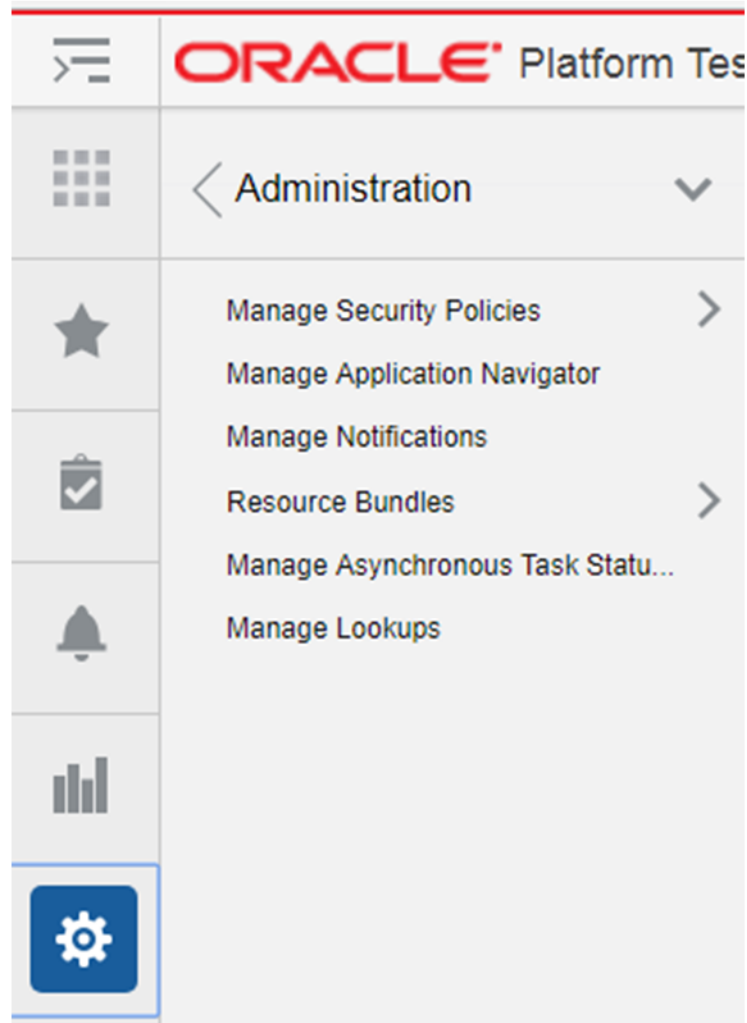
Manage Application Navigator

Application Navigator allows users to launch different applications from their current application. This lets users shuffle between multiple applications based on their privileges and avoid having to open a new tab and enter a new URL to launch an application. Users can instead click on the application, which will launch that application in a new tab or window based on the browser settings. If the application is a Single Sign-On (SSO) deployment then the user will be logged in by default and the default application page is displayed. If the deployment is non-SSO then the user is prompted to log in.

If a user has access to multiple applications (based on their defined role) they are able to see the list of available applications under the Application Navigator menu. If the user does not have access to any other applications, only the current application name will be displayed in the application navigator list.

Managing Application Links

Figure 4–14   Manage Application Navigator Screen

In order for users to see an application in the Application Navigator menu, they need to belong to the role associated with that application on the Manage Application Navigator screen. Only an application administrator is aware of the links to the different installed applications. RAAC provides the ability to add, modify and remove the links so they appear or not appear in the Application Navigator menu. Application administrator can determine which application role has access to which application based on customer's security model. For example, suppose Allocation users belonging to Allocation role X are to see the Invoice Match application link in the Allocation's Application Navigator menu. The administrator would launch RAAC from Allocation, navigate to Manage Application Navigator, add an entry for Invoice Match and enter X for the role.

System Records

A system row in RAF_INSTALLED_APPS table is one with the SYSTEM_ENTRY column set to 'Y'. These rows are locked and only administrators will be allowed to change the description, role, and URL.

Each application has one system record inserted at installation time in the RAF_ INSTALLED_APPS table. The system records cannot be added or deleted from RAAC. However, administrators are able to modify the role and URL associated with that system record's application using RAAC.

System records are not displayed in the Application Navigator list on the applications. Such records can be duplicated in RAAC (with SYSTEM_ENTRY='N') which then makes them available on the Application Navigator list.

Add

The Add action is enabled at all times and allows an administrator to add a new Application Navigator URL.

Add an Application Navigator URL

1. From the Actions menu, select Add. A new, empty Application Navigator entry is added to the Manage Application Navigator table.

2. Enter a name, the application URL, and the role to which the URL applies. The remaining columns are populated automatically.

3. System records cannot be inserted from the Manage Application Navigator screen. This will be created once per application by the installer at the application install time.

Modify

While all the columns on a row can be modified, only the ROLE and URL columns of a System Record can be modified by the administrators.

Duplicate

The Duplicate action adds a copy of each selected row below its originating row. The object name or unique identifier of the duplicate row is "Copy of Object Name" and appears in edit mode. Validation is performed so that a user is not presented with multiple applications with the same application name.

When a system record is duplicated, a copy of the record is created with the System_ Entry column set to 'N' and thus made available on the Application Navigator list.

Delete

The Delete action is enabled when an entry is selected.

When the user selects an application navigator entry and clicks Delete, the user is prompted with a warning message. Click Yes and the selected entry is removed.

A system entry record cannot be deleted from RAAC.

Figure 4–15   Delete Application Navigator Confirmation Dialog

Export to Excel

The Export to Excel action is enabled whenever you have the Manage Application Navigator window open. This action allows you to export the Manage Application Navigator list to a Microsoft Excel spreadsheet.

Manage Notifications

From the RAAC left hand navigation menu, the user can access the Manage Notifications link. The Manage Notifications tab displays the Manage Notifications table. This table is laid out using a master table and a detail table. The master table consists of Notification Types and the detail table consists of Notification Groups and Roles associated with the notification types.

Figure 4–16   Manage Notifications Screen
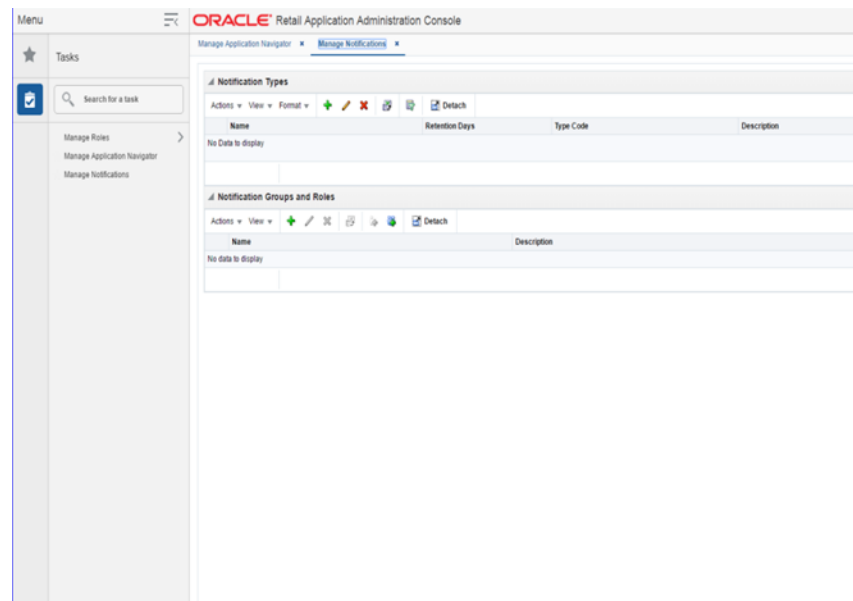
Overview

Retail Applications provide the ability to notify authenticated users in the application when business events occur. An example of a business event is when a background asynchronous task such as an approval of an allocation or a validation of a price change has completed.   Typically, the user is expected to take action on these notifications.

The notification icon on the Retail Application's navigation pane displays the count of unread notifications for the user. When a notification is generated, this count is incremented immediately and thereafter refreshed at regular configurable intervals.

The sliding sidebar menu expands when the user clicks on the notification icon. The expanded sidebar menu will show the latest notifications. Clicking on "See All" will show all the notifications in a new tab.

Clicking on the link for each launchable notification opens the specific UI flow that will allow the user to address any pending action for the notification.

Figure 4–17   Sliding Sidebar showing the Notification Count

Figure 4–18   Notifications Sidebar Menu
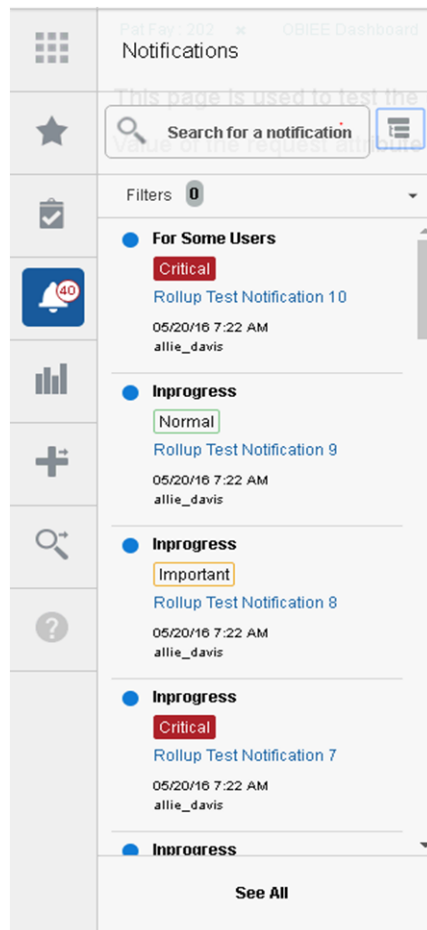
Notification ReST Services

ReST Endpoints have been exposed by the Notifications framework to ease the integration concerns with disparate applications.

Notification Types

Notification Types drive how users are notified about the result of their action. This result can be communicated to one user or a group of users based on the job roles.

Add

The Add action is enabled at all times and allows you to add a new notification type.

Add a Notification Type

1. From the Actions menu, select Add. The Create Notification Type dialog is displayed.

Figure 4–19   Create Notification Type Dialog

2. Each notification type requires the following information:

    ■   Name – The name of the notification type

    ■   Type Code – A meaningful name to identify the notification type

        The type code provides applications a way to uniquely identify a notification type.

    ■   Description – The description of the notification type

    ■   Retention Days – The number of days a notification of this type should be retained

        The reason for capturing Retention Days information at Notification Type level itself, is that it gives users the flexibility to delete certain Types of notifications quickly without cluttering notification views. To retain records infinitely, this value should be set to -1.

        > **Note:**   Administrators are discouraged from using '-1' for Retention Days as associated notifications will never be purged automatically. Only a system administrator can delete those manually through a sql script or direct row deletes through a sql tool.

        > **Note:**   Retention Days cannot be set to zero.

    ■   Email Address - Email addresses that must be notified in case a notification of this type is created. Multiple email addresses may be provided here, separated by blank space. This is an optional field, and can be skipped.

        > **Note:**   To be able to send emails for notifications, the Retail Alert Manager application must be deployed.

3. Click OK to create the new notification type.

Edit

The Edit action is enabled when a notification type is selected. The Edit action is used to add or modify the description and retention days of an existing notification type.

Edit a Notification Type

1. Select the notification type you want to edit.

2. From the Actions menu, select Edit. The Edit Duty Role dialog is displayed.

Figure 4–20   Edit Notification Type Dialog

3. Modify the type code, description, retention days or associated email addresses of the notification type and click OK.

Delete

The Delete action is enabled when you select a notification type that can be deleted from the database.

Deleting a notification type removes the notification type and all of the notification type's associated roles and groups. Any past notifications are also removed in order to maintain data integrity.

Delete a Notification Type

1. Select the notification type you want to delete.

2. From the Actions menu, select Delete. The Confirm dialog is displayed.

Figure 4–21   Delete Notification Type Confirmation Dialog

3. Click OK to delete the selected notification type.

Export to Excel

The Export to Excel action is enabled whenever you have the Manage Notifications window open. This action allows you to export the Notification Types list to a Microsoft Excel spreadsheet.

Notification Groups and Roles

The Notification Groups and Roles table consists of all the roles associated with a notification type master record. If there are no records associated with the notification type then a "no data to display" message appears in the Notification Groups and Roles table when the notification type is selected in the Notification Types table.
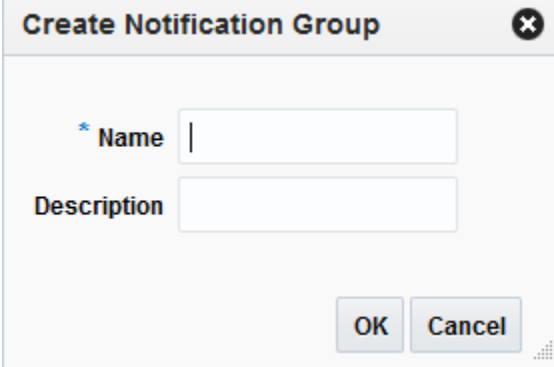
Create/Add Notification Group

The Create action is enabled at all times when a notification type is selected and allows you to create a new notification group associated with the selected notification type.

Create a Notification Group

1. From the Actions menu, select Create (or Add Notification Groups). The Create Notification Group dialog is displayed.

Figure 4–22   Create Notification Group Dialog

2. Enter a name and (optionally) a description of the group.

3. Click OK to create the new notification group.

Add Job Role

Once a notification group is created, you can add job roles that apply to the selected group.

Add a Job Role

1. From the Actions menu, select Add job Role. The Add Job Role dialog is displayed.

Figure 4–23   Create Job Role Dialog

2. Enter a name for the job role.

3. Click OK to create the new job role or OK and Add Another to create additional job roles.

   Upon clicking OK or OK and Add Another, the New name is validated. The validation checks for a valid name in the enterprise security solution of the retailer. If it finds a match, then it allows the user to proceed. If the name is not valid, an error is displayed.
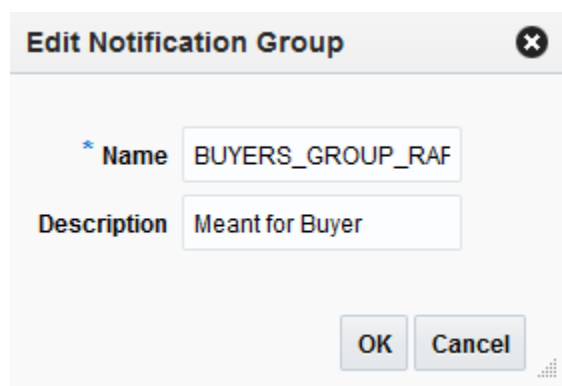
Edit

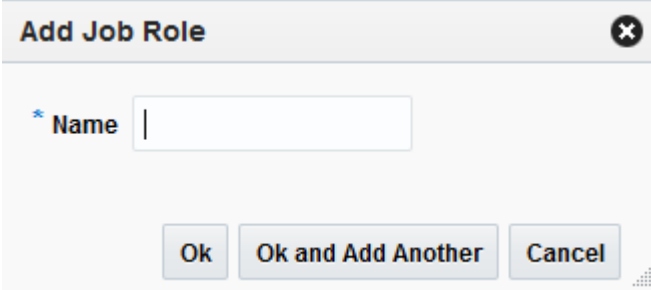The Edit action is enabled when a notification group or job role is selected. The Edit action is used to modify the name and description of an existing notification group or job role.

Edit a Notification Group

1. Select the notification group or job role you want to edit.

2. From the Actions menu, select Edit. The Edit Notification Group (or Edit Job Role) dialog is displayed.

Figure 4–24   Edit Notification Group Dialog

3. Modify the notification group or job role attributes and click OK.

   Upon clicking OK, the New name is validated. The validation checks for a valid name in the enterprise security solution of the retailer. If it finds a match, then it allows the user to proceed. If the name is not valid, an error is displayed.

Delete

The Delete action is enabled when you select a notification group or job role that can be deleted from the database.

Deleting a notification type removes the notification type and all of the notification type's associated roles and groups. Any past notifications are also removed in order to maintain data integrity.

Delete a Notification Type

1. Select the notification type you want to delete.

2. From the Actions menu, select Delete. The Confirm dialog is displayed.

Figure 4–25   Delete Notification Group Confirmation Dialog
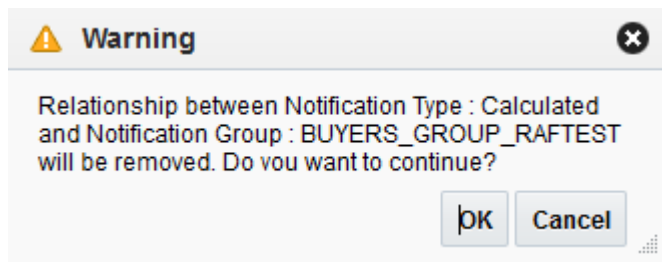
3. Click OK to delete the selected notification group or job role.

Export to Excel

The Export to Excel action is enabled whenever you have the Manage Notifications window open. This action allows you to export the Notification Groups and Roles list to a Microsoft Excel spreadsheet.

Resource Bundles

All Oracle Retail applications come packaged with resource bundles, files that contain text resources. These text resources appear throughout the Oracle Retail application as instructions, messages, labels, errors, virtually any text that appears in the application comes from a resource bundle.

Many Oracle Retail applications support the ability for retailers to customize the text that appears in the Oracle Retail application to better match that retailer's business language. The capability to change the text for a resource is provided through the Resource Bundles area of RAAC. Retailers can customize text resources for any Oracle Retail supported language.

Resources and Customizations Explained

All text resources in the Oracle Retail application have the following values:

- **Base Application Code** – This value identifies the Oracle Retail application that owns this text resource.

- **Language** – This value identifies the language of the text resource.

- **Bundle** – This value identifies the bundle file where this text resource resides.

- **Resource** – This value is the unique name that can be used to get the text for this resource.

- **Base Text** – This value is the text provided by the Oracle Retail application. This is the text that will appear in the application by default.

Every text resource in Oracle Retail applications is uniquely identified by the combination of Application Code, Language, Bundle, and Resource.

A resource customization is a change that a retailer made to the text of a resource. In addition to the values above, customizations also have the following values:

- **Custom Text** – This is the custom text provided by the retailer for the resource.

- **Text** – This is the text that will actually appear in the Oracle Retail application. If a retailer has provided custom text for the resource, then this value will display the retailer's custom text. Otherwise, it will display the base text provided by the Oracle Retail application.

Manage Resource Text

From the RAAC left hand navigation menu, the user can access the Manage Resource Text link. The Manage Resource Text screen provides the capability to customize the resources present in the resource bundles in an application. The user can search the resources in the application using the Bundle, Language, Resource, Contains Custom Text, or Text fields.

Figure 4–26 Manage Resource Text Screen
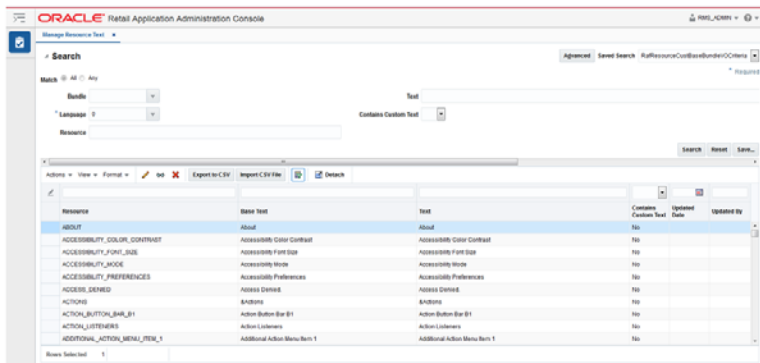
The results table in the Manage Resource Text screen will display the Resource, Base Text, Text, Contains Custom Text, Updated Date, and Updated By columns for the resources matching the entered search criteria. If there is a custom value present for a resource, it will be displayed in the Text column along with the date and the user who customized that entry. If there is no custom value present for a resource, the Text column will display the base text and the Updated Date and Updated By columns will be displayed empty. The user can view, add, edit, or remove custom text for a resource using this screen.

The user can also export the results table to an Excel CSV file, or import an Excel CSV file with customized resources using this screen. The export and import actions allow the user to provide custom text for many resources at one time, and upload all the custom values in a single action in RAAC.

Figure 4–27 View, Add, Edit or Delete Custom Text
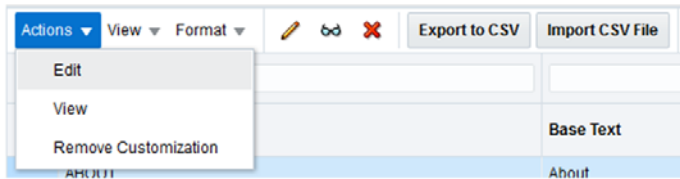
View

The user can view a resource, base text and custom text using the View menu item or the binoculars icon. The View option opens up a popup showing the resource, its base text and custom text. Only one record can be viewed at a time in a popup. If more than one record or no record is selected, a warning is displayed to the user.

Figure 4–28   View Resource Text
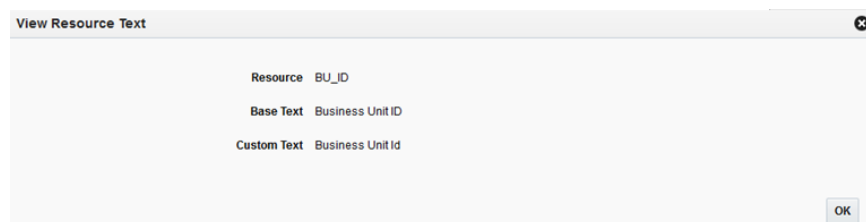
Add or Edit

The user can add or edit the custom text for a resource using the Edit menu item or the pencil icon. If no custom text is present for a resource, a new custom text will be added. If a custom text is already present for a resource, it will be modified. The Updated Date and Updated By columns will be changed to reflect the addition or modification. The Edit option opens up a popup showing the resource, its base text and editable custom t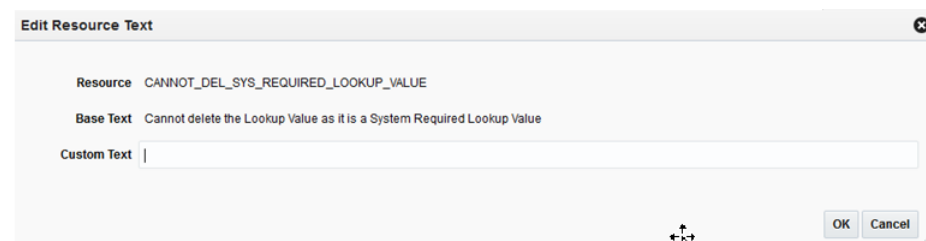ext. Only one record can be edited at a time in a popup. If more than one record or no record is selected, a warning is displayed to the user. The Edit popup can also be opened by double clicking a record on the results table.

Figure 4–29   Edit Resource Text

Remove Customization

The user can remove the custom text for one, many or all the selected resources using the Remove Customization menu item or the cross icon. The user can select a single or multiple records and remove custom values for the records. The Remove Customization option will give a confirmation warning message before the custom text is removed for the selected records. The base text is not impacted by the Remove Customization operation.

Figure 4–30   Delete Custom Text Confirmation Dialog

Export to CSV

The user can export the details of the resources from the results table to an Excel CSV file. When the user selects the Export to CSV action, the browser will perform its download behavior. Depending on the user's browser preferences, the browser may prompt the user asking whether the exported file should be opened or saved, what

program should be used to open the file, or where on the user's computer the file should be saved.

Import CSV File

See "Importing from a File" on page 4-38 for additional details.

Importing from a File

The user can change the values in the Text column of an exported CSV, and then import the CSV file. When the user does this, the values of the Text column are imported into the custom text of each newly customized resource. The user can customize multiple rows with a single action by importing a file which contains multiple customized rows. Importing a file that has not been modified will do nothing, since the text for each resource is already exactly the same in RAAC the Oracle Retail application.

The imported file must meet the following assumptions for the import to succeed:

- The imported file must be a well-formed CSV file. The user can use any compatible program to edit the exported CSV file, as long as the program is able to read the CSV file and save it again in CSV format.

- The user should only edit values in the Text column. If the user edits any of the values in the Base Application Code, Language, Bundled, or Resource columns, the import may fail, or invalid customizations may be created.

- The user can only add or edit customizations in the import. The user cannot remove customizations using import. Deleting the contents of the Text field or copying the Source Text into the Text field will cause the row to be ignored.

- The size of the imported file is limited to 2000 KB. Users will not be allowed to upload a file larger than this.

When the user selects the Import CSV file action, a popup opens prompting the user to select a file to import.

Figure 4–31   Import CSV File Dialog
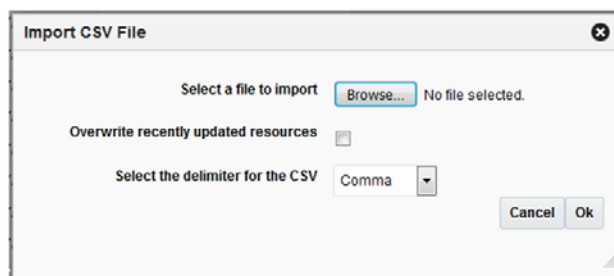
It is possible that, after a user exports resources to a CSV file, but before the user imports the file into RAAC, another user may go to RAAC and customize one or more of the exported resources. In this case, the values of resources in RAAC would be newer than the values in the exported file. In order to ensure that an import does not inadvertently overwrite these values, the import action will skip resources that have been recently updated in RAAC, and notify the user. The user can tell the import to instead use the values from the imported file by selecting the "Overwrite recently updated resources" checkbox on the import popup.

Imports Hub

From the RAAC left hand navigation menu, the user can access the Imports Hub link. The Imports Hub screen provides the capability to view and manage import actions that have been created, and take action to resolve any issues encountered during an import.

Figure 4–32   Imports Hub Screen

The Imports table will display information about each import that has been created. The table will display the Status, File Name, Errors, Created By, Creation Date, Last Updated By, and Last Updated Date. The user can create a new import, delete an import, mark an import as complete, restart an import, or view the errors for an import from this screen.

Figure 4–33   Import Actions Menu

Create

The user can import a CSV file from the Imports Hub page. See "Importing from a File" on page 4-38 for additional details.

Delete

The user can delete one or more imports. When the user deletes an import, that row is removed from the table completely, as well as any unresolved errors in the import. Any customizations that were already created by the import will still exist. The user cannot delete an import that is in "Pending" status. This is to make sure that an import is not deleted while it is being processed.

Note that there is no purge process for imports. All imports will remain in the database until manually deleted by the user.

Mark as Complete

The user can select an import that is in "Complete with Errors" status and change the status to "Complete" This indicates that the user has finished addressing errors with the import, and all outstanding errors should be dismissed.

Restart Import

After the user addresses any errors that were encountered when importing customizations, the user can restart the import to create customizations for the fixed resources.
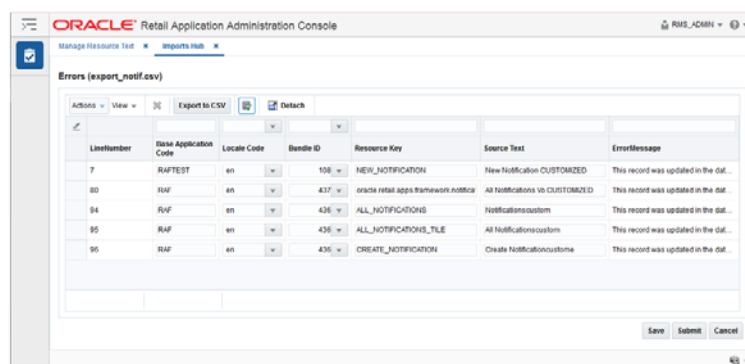
View

When an import on the Imports Hub screen is marked with status "Complete with Errors," this means that the import encountered errors with one or more of the customizations in the imported file. In the table, the Errors column indicates the number of customizations that had errors. This number becomes a link that the user can click to view details for the customizations with errors. This takes the user to a secondary screen within Imports Hub. See the section below titled "View Errors" for more details.

View Errors

The Import Errors screen allows the user to view the details of any errors encountered while importing customizations from a CSV file. The user can reach this screen via the Imports Hub screen. See "View" on page 4-36 within the "Imports Hub" section for additional details.

Figure 4–34   Import Errors Screen

The Import Errors screen allows the user to view all imported customizations that encountered errors, as well the error message indicating why that customization could not be created. The Errors table displays Line Number, Base Application Code, Language, Bundle, Resource, Text, and Error Message. The user can edit or delete an imported customization, or Export to CSV.

Edit

The Errors table provides inline editing for the Base Application Code, Language, Bundle, Resource, and Text columns. This allows the user to resolve cases where a customization does not identify an existing resource, or cases where two or more customizations identify the same resource.

Delete

The user can delete an imported customization. The imported record is removed from the errors table, and no customization is created.

Export to CSV

The user can export the contents of the Errors table back to a CSV file, fix any errors in the CSV, and re-import the file. In this case, the user would generally want to return to the Imports Hub screen and mark the first import as complete, since the errors were fixed by creating a new import.

Important Considerations While Customizing Resources

These are the important considerations that you should keep in mind while customizing resources:

- When customizing a resource text, consider the length of the custom text compared to the base text. Some of the labels and messages may not look good on the page, if the custom text is too long compared to the base text.

- After customizing the resource text, the changes are not immediately visible in the application. The customized resources are visible in the application only after the resource bundle cache expires. Please refer to the application documentation for the cache expiry time.

Manage Asynchronous Task Status

Asynchronous tasks are background processes launched by users of Oracle Retail Applications. An example of a background process is the calculation of allocations.

Each asynchronous task is logged in the database table called RAF_ASYNC_TASKS.

Administrators can view the latest status of asynchronous tasks through the Manage Asynchronous Task Status function from the Oracle Retail Application's Settings menu.

The user can access the Manage Asynchronous Task Status function from the Administration submenu under the Settings menu. This feature is not accessible from the RAAC left hand navigation menu since it is packaged with the application itself rather than the RAAC application. The Manage Resource Asynchronous Task Status screen displays the Asynchronous Task records. Users are able to search for a certain asynchronous task and view a task. Users are also able to purge asynchronous tasks present in the system.

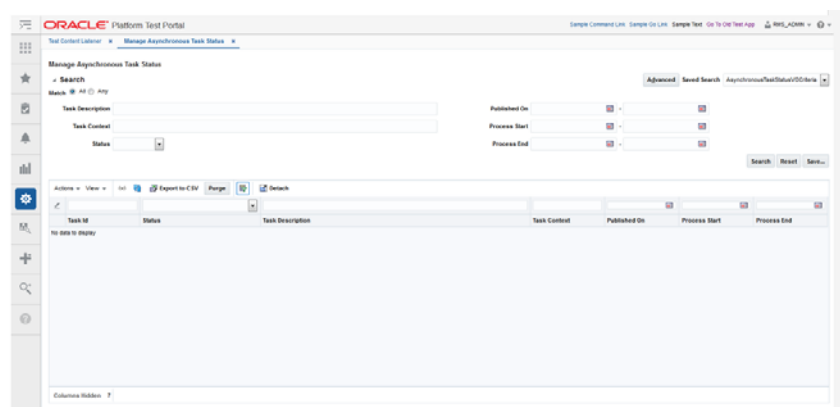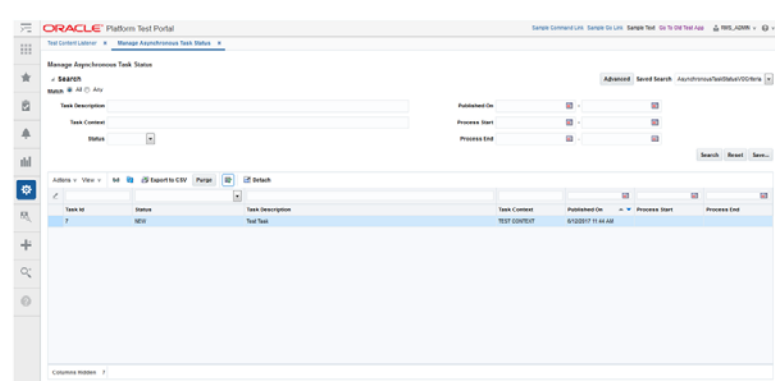*Figure 4–35   Manage Asynchronous Task Status Screen*

Figure 4–36   Manage Asynchronous Task Status Screen (After search button is clicked)

View

The user can view additional details for the selected asynchronous task by clicking on the View Actions menu item or the View icon of the table toolbar. The View option opens up a popup showing additional details for the selected asynchronous task. Only one record can be viewed at a time in a popup.
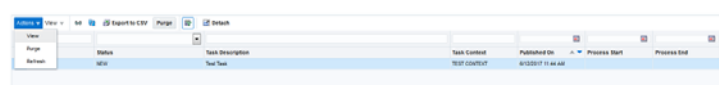
Figure 4–37   Action Menu

Figure 4–38   View Task Dialog

Purge Task

The user can purge asynchronous task records in the system. The number of days is defined in the async.task.purge.retention.days property which can be updated through the Manage Application Properties feature described above in this chapter. The Purge can be invoked through the table toolbar Purge button or Purge Actions menu item.

Figure 4–39   Purge Tasks Dialog
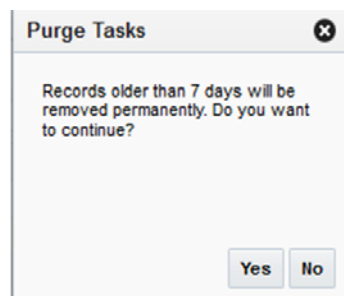
When the Purge action is complete, a popup will appear showing the number of purged asynchronous task records

Figure 4–40   Purge Tasks Success Dialog
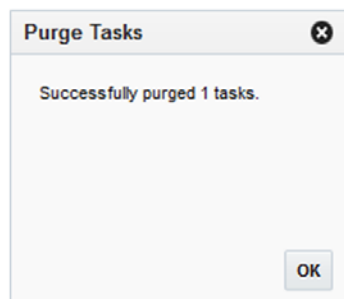
Export to Excel

The Export to Excel action is enabled whenever Asynchronous Task records are present in the table. This action allows the user to export the Asynchronous Task records to a Microsoft Excel spreadsheet.

Manage Lookups

Lookup records in Oracle Retail Applications represent items that users can select in dropdown and list-of-value inputs in application screens. Examples include:

- Location Types (warehouse or store)

- Clearance Status (worksheet, submitted, rejected, approved, etc)

- Markup Types (cost or retail)

These lookups are stored in the database tables RTC_LOOKUP_TYPES_B and RTC_ LOOKUP_VALUES_B.

Administrators can customize these lookups through the Oracle Retail Application's Settings menu.

From the Settings menu, the user can access the Manage Lookups screen under the Administration submenu. This feature is not accessible from the RAAC left hand navigation menu since it is packaged with the application itself rather than the RAAC application. The Manage Lookups screen displays the lookup types and their associated lookup values for the Retail Application. The Manage Lookups screen shows information about the lookup types and values such as the Lookup Code, Meaning, Description, System flag, and for the Lookup Value only - an Enabled flag. Users are also able to create, update, delete, cancel, and save changes made to the lookup types and values.

Figure 4–41   Manage Lookups Screen

Figure 4–42   Manage Lookups Screen (After clicking a different Lookup Type)

Add

The user can add a lookup type by clicking on the Add menu item or the Add icon of the Lookup Type table toolbar . The user can enter the lookup type's Code, Meaning, and Description. The system column is non-editable and will always be set to No when a user creates a new record. Lookup Types must be unique, so if a user enters an already existing Lookup Type Code, an error message will display.

Figure 4–43   Add Lookup Type Screen

To associate a lookup value with the lookup type, the user can click on the Add menu item or the Add icon of the Lookup Values table toolbar. The user can enter the lookup value's Code, Meaning, Description, and Enabled flag. The system flag is also non-editable.

Figure 4–44   Add Lookup Value Screen

Edit

The Lookup Types and Lookup Value tables are editable so the user can readily update the records. All columns are editable except for the system column. The Lookup Code, however, cannot be edited if it is a system record.

If a lookup type or lookup value is a System record, the user will not be able to update the lookup code for that record.

Lookup code must be unique, so if a user updates to an already existing Lookup Type or Lookup Value Code, an error message will display.

Figure 4–45   Edit Lookup Type (Existing Lookup Code)

Delete

The user can delete a lookup type or lookup value by clicking on the Delete menu item or the Delete icon of the Lookup table toolbar . To delete a lookup type or lookup value, the user should select a record then click either the Delete menu item or the Delete icon.

The Delete function is applicable only to Non-System records. When a user selects a system record, the Delete menu item and the Delete icon are disabled.

Figure 4–46   Delete Lookup Type

Figure 4–47   Delete Lookup Type (Disabled Delete Button on System Records)

Lookup Types

Actions ▾ View ▾ ✚ ✕ 🔄	Detach			
Lookup Type Code	Meaning	Description		System
ALLOC_STATUS	Allocation record status	The set of valid status an allocation record can be on at any given time.		Yes
CUSTOM_TYPE	Custom Type	This is a custom lookup type.		No

Figure 4–48   Delete Lookup Value

Lookup Values

Actions ▾ View ▾ ✚ ✕ 🔄	Detach				
Lookup Value Code	Meaning	Description	Enabled	System	
C	Custom	This is a custom lookup value.	Yes	No	

Figure 4–49   Delete Lookup Value (Disabled Delete Button on System Records)

Lookup Values

Actions ▾ View ▾ ✚ ✕ 🔄	Detach				
Lookup Value Code	Meaning	Description	Enabled	System	
A	Approved	The allocation record has been approved.	Yes	Yes	
C	Calculation	The allocation record is being processed in the system.	Yes	No	

Save / Cancel

Save and Cancel buttons are also available to commit and rollback the changes made to the records. These buttons are located at the bottom right section of the Manage Lookups screen.

The user can save changes made to both the lookup types and lookup values tables using the Save button. All changes made using the screen's Add, Edit, and Delete functions will take effect once the user clicks on the Save button.

The user may also use the cancel button to undo changes made to both the lookup types and lookup value tables. However, the Cancel button may only undo changes that have not yet been saved. All previously saved changes may not be undone using the Cancel button.

Clicking on the Cancel button also closes the Manage Lookups screen after the rollback is complete.