Contents

Documentation Accessibility

Follow us on Social Media

1  Getting Started

About the Application 1-1
Application Features 1-1
About the Home Page 1-2
Launching the Application 1-5
Considerations When Using the Application Interface 1-5
Managing Application Access 1-5
   Enabling Access to Application Artifacts On Mobile Devices 1-6
   About Cell Formatting 1-6
   Copying and Pasting Data Between Microsoft Excel and Application Web Forms 1-6
   About Instant Save 1-7
   About Autosave 1-7

2  Creating an Application

About Creating an Application 2-1
Selecting the Application Type 2-1
Creating a Reporting Application 2-2
Creating a Sample Application 2-3
Creating a Standard Application 2-4
Choosing Setup Options 2-5
   Setting Up the Calendar 2-5
   Setting Up Currencies 2-6
      About Simplified Multicurrency 2-6
      About Standard Multicurrency 2-9
   Customizing the Cube Names 2-9
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating User Variables</td>
<td>4-12</td>
</tr>
<tr>
<td>Setting Limits for User Variables on Forms</td>
<td>4-12</td>
</tr>
<tr>
<td>Deleting User Variables</td>
<td>4-13</td>
</tr>
<tr>
<td>Viewing Activity Reports</td>
<td>4-13</td>
</tr>
<tr>
<td>Importing and Exporting Data and Metadata</td>
<td>4-14</td>
</tr>
<tr>
<td>Importing Metadata</td>
<td>4-14</td>
</tr>
<tr>
<td>Creating the Import File</td>
<td>4-14</td>
</tr>
<tr>
<td>Loading the Import File</td>
<td>4-18</td>
</tr>
<tr>
<td>Exporting Metadata</td>
<td>4-19</td>
</tr>
<tr>
<td>Importing and Exporting Data</td>
<td>4-20</td>
</tr>
<tr>
<td>Importing Data</td>
<td>4-20</td>
</tr>
<tr>
<td>Driver Member Casting Errors</td>
<td>4-22</td>
</tr>
<tr>
<td>Exporting Data</td>
<td>4-24</td>
</tr>
<tr>
<td>Viewing Data Import and Export Status</td>
<td>4-25</td>
</tr>
<tr>
<td>Importing Exchange Rates for a Simplified Multicurrency Application</td>
<td>4-25</td>
</tr>
<tr>
<td>Creating and Refreshing Application Databases</td>
<td>4-26</td>
</tr>
<tr>
<td>Creating Application Databases</td>
<td>4-26</td>
</tr>
<tr>
<td>Before Refreshing the Database</td>
<td>4-26</td>
</tr>
<tr>
<td>Refreshing Application Databases</td>
<td>4-26</td>
</tr>
<tr>
<td>Adding an Aggregate Storage Outline to an Application</td>
<td>4-27</td>
</tr>
<tr>
<td>About Aggregate Storage</td>
<td>4-27</td>
</tr>
<tr>
<td>Aggregate Storage Outline Cube Characteristics</td>
<td>4-27</td>
</tr>
<tr>
<td>Process for Adding an Aggregate Storage Database to an Application</td>
<td>4-27</td>
</tr>
<tr>
<td>Converting to a Standard Application</td>
<td>4-28</td>
</tr>
<tr>
<td>Converting a Standard or Reporting Application to an Enterprise Application</td>
<td>4-28</td>
</tr>
<tr>
<td>Conversion Considerations</td>
<td>4-30</td>
</tr>
<tr>
<td>Removing an Application</td>
<td>4-30</td>
</tr>
<tr>
<td>Setting the Daily Maintenance Process Time</td>
<td>4-30</td>
</tr>
<tr>
<td>Uploading and Downloading Files Using the Application Inbox and Outbox</td>
<td>4-32</td>
</tr>
</tbody>
</table>

5 **Using the Member Selector**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working with Members</td>
<td>5-1</td>
</tr>
<tr>
<td>Making Selections</td>
<td>5-1</td>
</tr>
<tr>
<td>Member Relationships</td>
<td>5-4</td>
</tr>
<tr>
<td>Using Wildcards in Searches</td>
<td>5-4</td>
</tr>
<tr>
<td>Selecting Attribute Values as Members</td>
<td>5-5</td>
</tr>
<tr>
<td>Selecting Members for Forms</td>
<td>5-8</td>
</tr>
<tr>
<td>Selecting Substitution Variables as Members</td>
<td>5-9</td>
</tr>
<tr>
<td>Selecting User Variables as Members</td>
<td>5-10</td>
</tr>
</tbody>
</table>
# Designing Dashboards

The Power of Dashboards
Concepts in Designing Dashboards
Designing Dashboards
Creating Dashboards Containing Master Forms and Details
About Your Dashboard’s Layout
About the Gauge Chart Type
About the Tile Chart Type
Customizing Dashboard Colors
Setting Line Width in Line and Combination Charts
About Global and Local POVs
Dashboard POVs and Valid Intersections

# Designing Infolets

About Infolets
Anatomy of an Infolet
Determining Infolet Content
  - Designing Forms for Infolets
  - Designing Charts for Infolets
Using the Infolets Designer
Creating Infolets
Working with Infolets
Customizing the Application Interface to Access Infolets

# Managing Sandboxes

About Sandboxes
Enabling Sandboxes
Enabling Version Members
How Sandboxes Work with HSP_View and Version Members
  - The HSP_View Dimension
  - Sample Formula for the ConsolidatedData Member
  - Sandbox Version Members
  - Sandbox Implications on Member Formulas
What Happens When a Sandbox is Published?
Managing Sandboxes
# Defining Valid Intersections

## Understanding Valid Intersections
- Valid Intersection Groups
- Valid Intersection Rules
- Anchor and Nonanchor Dimensions
- Valid Intersection Examples
- Redundancy or Overlap in Valid Intersection Rules
- Shared Members and Valid Intersection Rules
- Substitution Variables and Valid Intersection Rules
- Evaluation Order

## Creating Valid Intersections

## Managing Valid Intersections
- Viewing Valid Intersections
- Changing the Valid Intersection Group Evaluation Order
- Disabling and Enabling Valid Intersection Groups
- Editing Details for a Valid Intersection Group
- Duplicating Valid Intersection Groups
- Deleting a Valid Intersection Group

## Suppressing Invalid Data in Forms

## Clearing Invalid Data
- About Invalid Data
- Working With Invalid Intersection Reports
- Clearing Data at Invalid Intersections

## Working with Valid Intersections
- Working with Valid Intersections in Application Forms
- Working with Valid Intersections in Calculation Manager Rule Runtime Prompts

# Defining Data Maps

## Understanding Data Maps
- About Applications
- About Reporting Cubes
- Unsupported Features with Data Maps
- Data Maps and Substitution Variables

## Creating Data Maps

## Refreshing Data Maps

## Managing Data Maps

## Changing Mapping Information
# Managing Application and System Settings

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>What Application and System Settings Can I Specify?</td>
<td>17-1</td>
</tr>
<tr>
<td>Defining User Variables</td>
<td>17-4</td>
</tr>
<tr>
<td>Customizing Your Display</td>
<td>17-4</td>
</tr>
<tr>
<td>Announcing Upcoming Events</td>
<td>17-5</td>
</tr>
<tr>
<td>Specifying Artifact Labels</td>
<td>17-5</td>
</tr>
<tr>
<td>Which Artifact Labels Can be Localized?</td>
<td>17-6</td>
</tr>
<tr>
<td>Working With the Artifact Labels Grid</td>
<td>17-7</td>
</tr>
<tr>
<td>Adding Languages and Defining Localized Artifact Labels</td>
<td>17-7</td>
</tr>
<tr>
<td>Exporting and Importing Localized Artifact Labels for Editing</td>
<td>17-8</td>
</tr>
</tbody>
</table>

# Accessing More Administrative Tasks

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About the Navigator Menu</td>
<td>18-1</td>
</tr>
<tr>
<td>Administering Data Load Settings</td>
<td>18-2</td>
</tr>
<tr>
<td>Importing Using Data Management</td>
<td>18-3</td>
</tr>
<tr>
<td>Importing Data Using Data Management</td>
<td>18-3</td>
</tr>
<tr>
<td>Import Scenario Case 1: Importing Data when Member Names Match</td>
<td>18-3</td>
</tr>
<tr>
<td>Import Scenario Case 2: Importing Data when Member Names Do Not Match</td>
<td>18-3</td>
</tr>
<tr>
<td>Scheduling Jobs in Data Management</td>
<td>18-7</td>
</tr>
<tr>
<td>Drilling Through to Source Data</td>
<td>18-8</td>
</tr>
<tr>
<td>For More Information</td>
<td>18-8</td>
</tr>
<tr>
<td>Administering Action Menus</td>
<td>18-11</td>
</tr>
<tr>
<td>Creating and Updating Action Menus</td>
<td>18-11</td>
</tr>
<tr>
<td>Working with Action Menu Items</td>
<td>18-11</td>
</tr>
<tr>
<td>Defining Action Menu Items</td>
<td>18-12</td>
</tr>
<tr>
<td>Administering Alias Tables</td>
<td>18-15</td>
</tr>
<tr>
<td>About Aliases</td>
<td>18-15</td>
</tr>
<tr>
<td>About Alias Tables</td>
<td>18-16</td>
</tr>
<tr>
<td>Working with Alias Tables</td>
<td>18-16</td>
</tr>
<tr>
<td>Specifying a Default Alias Table and Setting Member and Alias Display Options</td>
<td>18-17</td>
</tr>
<tr>
<td>Administering Dimensions</td>
<td>18-18</td>
</tr>
<tr>
<td>About Dimensions</td>
<td>18-18</td>
</tr>
<tr>
<td>Working with Dimension Hierarchies</td>
<td>18-19</td>
</tr>
<tr>
<td>Filtering the Dimension View by Cube</td>
<td>18-19</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Working with UDAs</td>
<td>18-53</td>
</tr>
<tr>
<td>Working with Member Formulas</td>
<td>18-54</td>
</tr>
<tr>
<td>Viewing Details of Formula Validation</td>
<td>18-55</td>
</tr>
<tr>
<td>Working with Formula Expressions</td>
<td>18-55</td>
</tr>
<tr>
<td>Administering Forms</td>
<td>18-59</td>
</tr>
<tr>
<td>About Forms</td>
<td>18-59</td>
</tr>
<tr>
<td>Form Components</td>
<td>18-60</td>
</tr>
<tr>
<td>Form Design Considerations</td>
<td>18-61</td>
</tr>
<tr>
<td>Understanding Implied Sharing in Forms</td>
<td>18-63</td>
</tr>
<tr>
<td>Creating Simple Forms</td>
<td>18-63</td>
</tr>
<tr>
<td>Defining the Layout</td>
<td>18-64</td>
</tr>
<tr>
<td>About Precision Settings</td>
<td>18-74</td>
</tr>
<tr>
<td>Setting Form Precision and Other Options</td>
<td>18-74</td>
</tr>
<tr>
<td>Creating Asymmetric Rows and Columns</td>
<td>18-75</td>
</tr>
<tr>
<td>Adding Formula Rows and Columns</td>
<td>18-75</td>
</tr>
<tr>
<td>Defining Simple Form Page and Point of View</td>
<td>18-76</td>
</tr>
<tr>
<td>Designing Specific Types of Forms</td>
<td>18-76</td>
</tr>
<tr>
<td>Designing Forms for Multiple Currencies</td>
<td>18-77</td>
</tr>
<tr>
<td>Designing Forms with Formula Rows and Columns</td>
<td>18-77</td>
</tr>
<tr>
<td>Designing Forms with Data Validation</td>
<td>18-77</td>
</tr>
<tr>
<td>Designing Forms with Global Assumptions</td>
<td>18-77</td>
</tr>
<tr>
<td>Designing Forms for Rolling Forecasts</td>
<td>18-78</td>
</tr>
<tr>
<td>Working with Forms and Form Components</td>
<td>18-81</td>
</tr>
<tr>
<td>Selecting and Opening Forms and Folders</td>
<td>18-81</td>
</tr>
<tr>
<td>Previewing Forms</td>
<td>18-82</td>
</tr>
<tr>
<td>Printing Form Definitions</td>
<td>18-82</td>
</tr>
<tr>
<td>Searching for Forms</td>
<td>18-83</td>
</tr>
<tr>
<td>Editing Forms</td>
<td>18-83</td>
</tr>
<tr>
<td>Moving, Deleting, and Renaming Forms</td>
<td>18-94</td>
</tr>
<tr>
<td>How Cell Formatting in Smart View Persists in Planning</td>
<td>18-94</td>
</tr>
<tr>
<td>Managing Forms and Folders</td>
<td>18-95</td>
</tr>
<tr>
<td>Creating Folders</td>
<td>18-95</td>
</tr>
<tr>
<td>Working with Folders</td>
<td>18-96</td>
</tr>
<tr>
<td>Administering Rules</td>
<td>18-97</td>
</tr>
<tr>
<td>About Rules</td>
<td>18-97</td>
</tr>
<tr>
<td>Adding and Removing Business Rules in Forms</td>
<td>18-97</td>
</tr>
<tr>
<td>Setting Business Rule Properties</td>
<td>18-99</td>
</tr>
<tr>
<td>Viewing Rules Usage</td>
<td>18-101</td>
</tr>
<tr>
<td>About Runtime Prompts</td>
<td>18-104</td>
</tr>
<tr>
<td>Understanding Runtime Prompts</td>
<td>18-104</td>
</tr>
<tr>
<td>About Runtime Prompts and Approvals Security</td>
<td>18-109</td>
</tr>
</tbody>
</table>
Using Smart View to Manage Applications

About Managing Applications in Smart View E-1
Planning Admin Extension and Office AutoCorrect E-1
Controlling the Display of Application Management Options in Smart View E-2
Downloading the Application Templates E-3
  Downloading the Template in Smart View E-3
  Downloading the Application Template Zip File from the Business Process E-4
Creating an Application E-5
Working with Artifacts in the Application Template E-6
  About Working with Artifacts in the Application Template E-6
  Application Definition E-6
  Dimension Definition E-10
  Attribute Dimension Definition E-11
  Data Definition E-13
  Substitution Variable Definition E-14
  Security Definition E-15
  Advanced Settings Definition E-17
Updating an Application in Smart View E-20
Deleting an Application E-21

Using Smart View to Import and Edit Application Metadata

About Smart View F-1
Planning Admin Extension and Office AutoCorrect F-2
Importing Planning Dimensions in Smart View F-2
  Importing Dimensions in Smart View F-2
Using Smart View Grids F-3
  Overview of the Smart View Grid F-3
  The Smart View Grid and Ribbon Display F-3
  Guidelines for Using the Smart View Grid F-4
  Default Metadata Dimension Member Properties F-5
Editing Members in Smart View F-7
Adding Application Members in Smart View F-8
Adding Members in Smart View F-8
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guidelines for Adding Members in Smart View</td>
<td>F-9</td>
</tr>
<tr>
<td>Moving Members in Smart View</td>
<td>F-10</td>
</tr>
<tr>
<td>Guidelines for Moving Members in Smart View</td>
<td>F-10</td>
</tr>
<tr>
<td>Working with Attribute Dimensions</td>
<td>F-10</td>
</tr>
<tr>
<td>Adding Attribute Dimension Members</td>
<td>F-11</td>
</tr>
<tr>
<td>Associating Attribute Dimension Members with Dimension Members</td>
<td>F-12</td>
</tr>
<tr>
<td>Designating Shared Members in Smart View</td>
<td>F-14</td>
</tr>
<tr>
<td>Refreshing and Creating Databases in Smart View</td>
<td>F-14</td>
</tr>
</tbody>
</table>

G  Best Practices for Designing Your Application

- Best Practices for Getting Started        G-1
- Best Practices for Planning Your Application G-3
- Best Practices for Applying Your Design   G-5

H  Using Direct Page Links for Forms and Dashboards in EPM Cloud

- URL Parameters                             H-1
- URL Pattern                                H-1
- User Authentication for Secured Access     H-2

I  Frequently Asked Questions
Documentation Accessibility

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

Access to Oracle Support

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

Follow EPM Information Development on these social media sites:

LinkedIn - http://www.linkedin.com/groups?gid=3127051&goback=.gmp_3127051
Twitter - http://twitter.com/hyperionepm
Facebook - http://www.facebook.com/pages/Hyperion-EPM-Info/102682103112642
See all videos on YouTube - https://www.youtube.com/oracleepminthecloud
To provide feedback on this documentation, send email to epmdoc_ww@oracle.com.
1

Getting Started

Planning is a budgeting and forecasting solution that integrates financial and operational planning processes and improves forecast accuracy.

Related Topics

- About the Application
  An application is a related set of dimensions and dimension members used to meet a set of planning needs.

- Application Features
  Key features in the Planning application enable you to reduce budgeting and planning cycles and improve forecast reliability.

- About the Home Page
  After you create the application, the Home page is the launch point for accessing your planning tasks in the simplified interface.

- Launching the Application
- Considerations When Using the Application Interface
- Managing Application Access

About the Application

An application is a related set of dimensions and dimension members used to meet a set of planning needs.

Each application has its own accounts, entities, scenarios, and other data elements.

See this guide for application administration tasks, including:

- Creating, deleting, and managing an application
- Creating forms, task lists, and menus
- Managing currency conversions and exchange rates
- Identifying the review and approval process, requirements, and participants, and managing the budgeting process

Application Features

Key features in the Planning application enable you to reduce budgeting and planning cycles and improve forecast reliability.

The application:

- Facilitates collaboration, communication, and control across multidivisional global enterprises
- Provides a framework for perpetual planning, to manage volatility and frequent planning cycles
• Decreases the total cost of ownership through a shorter roll out and implementa-
tion phase, and easier maintenance for an application
• Enhances decision-making with reporting, analysis, and planning
• Promotes modeling with complex business rules and allocations
• Integrates with other systems to load data
Watch this overview video for an introduction to the key features in the application.

Links to topics about the key application features described in the video:

• Designing Dashboards
• Managing Sandboxes
• Editing Dimension Properties in the Simplified Dimension Editor
• Focusing Your Analysis with Ad Hoc Grids
• Defining Valid Intersections
• Moving Data from One Cube to Another Cube Using Smart Push
• Using Your Own Excel Formulas in Planning

About the Home Page

After you create the application, the Home page is the launch point for accessing your
planning tasks in the simplified interface.

The simplified interface provides an intuitive user experience and an overview for
quick access to commonly used functions. For example, planners can access their
tasks, work with data, approve budgets, view reports, and control settings. Administra-
tors can manage and customize the application, create forms, dashboards, and info-
lets, import and export data and metadata, schedule jobs, define valid intersections,
make announcements, and create cross-subscription connections.

Watch this overview video to learn how to customize the interface to streamline your
workflow.

Overview: Customizing Workflow in Enterprise Performance Management Cloud
Watch these overview videos to learn how to perform key tasks in the Simplified Interface.

 Overview: Performing Key Tasks in Planning and Budgeting Cloud’s Simplified Interface Part 1

 Overview: Performing Key Tasks in Planning and Budgeting Cloud’s Simplified Interface Part 2

The Navigator icon in the left corner of the global header opens the Navigator menu, which serves as a sitemap of the application and displays links to all of the application pages to which you have access.

To refresh the Home page or to return to it after working elsewhere in the application, click the Home icon.

If your application uses infolets to show high-level, essential information, you can explore them by clicking the white dots that appear beneath the global header.
The left side of the Home page displays a greeting and the announcement area. The announcement area displays any system announcements entered by the administrator and helps you track your activity.

- **Activity**—Summarizes system announcements (the most recent announcement, sorted by effective date, appears at the top) and lists your open tasks
- **Recent**—Displays a list of links to recently visited forms and dashboards (up to 15 items). Clicking a link will launch the item in a secondary window. Clicking the star next to a link will tag it as a favorite.
- **Favorites**—Displays a list of links to forms or dashboards that were tagged as favorites, and prevents them from being overwritten. Click a link in Favorites to launch the item in a secondary window. To add items to Favorites, click Recent to view your recent user activity, and then click the star to the right of the item.
Launching the Application

To launch the application, enter the following URL in a new browser window:

https://Oracle PBCS service name/HyperionPlanning;

For example:

https://testnew1-testnew1.pbcs.us1.oraclecloud.com/HyperionPlanning

For information about enabling access to application artifacts on mobile devices and for considerations when using the application, see Managing Application Access.

Considerations When Using the Application Interface

Note the following considerations when using the application interface:

- The interface supports only Landscape mode for iPad and Android tablets.
- The user experience on tablets is better on iPad than on Android devices.
- Oracle recommends using the Safari web browser with iPads and Google Chrome with Android devices.
- If you use Internet Explorer (IE) 11, Oracle recommends that you use IE 11 in standard mode.
- A PDF viewer is required to view reports on the tablet. Oracle recommends the Adobe PDF viewer, but it can be opened in Kindle or Polaris Office on Android devices or a similar application on the iPad.
- For information on copying and pasting data between Microsoft Excel and application web forms, see Copying and Pasting Data Between Microsoft Excel and Application Web Forms.

Managing Application Access

Related Topics

- Enabling Access to Application Artifacts On Mobile Devices
- About Cell Formatting
- Copying and Pasting Data Between Microsoft Excel and Application Web Forms
- About Instant Save
- About Autosave
Enabling Access to Application Artifacts On Mobile Devices

To work with artifacts such as forms, task lists, and business rules on mobile devices, administrators must enable access to those artifacts for users.

To enable access to application artifacts on mobile devices:

1. From the Home page, click Navigator, and then under Setup, click Access Simplified Interface.
2. In Access Simplified Interface, click to launch the forms, tasks, and rules selector.
3. In the Selector, select the forms, tasks, and rules you want to enable for mobile device access, and then click OK.
4. Back in Access Simplified Interface, select the Forms, Tasks, and Rules tabs to view the artifacts that are enabled for mobile device access.

About Cell Formatting

You can apply either the formatting saved in Oracle Smart View for Office or the formatting set up in Planning for the form.

To select which formatting to apply:

1. In the form, click Actions, and then Apply.
2. Click:
   - Cell Styles: To use Planning’s formatting
   - Custom Styles: To use the formatting saved in Smart View

Copying and Pasting Data Between Microsoft Excel and Application Web Forms

To copy and paste data from Microsoft Excel:

1. In Excel, highlight the data in one cell or in a range of cells, and press Ctrl+C to copy the data onto the clipboard.
2. Highlight and select the target cell or cells in the application Web form, and then press Ctrl+V.
3. When the Clipboard helper is displayed, press Ctrl+V again. The data is pasted to the Clipboard helper.
4. Click Paste to paste the data into the application Web form.

The following features are supported in Internet Explorer browser only:

- Copying data from application web forms and paste it into Microsoft Excel
- Copying data from one application web form to another
- Copying and pasting nonnumeric data, for example, Smart List, Date, and Text data types
About Instant Save

When planners work in simple data forms and they click **Save**, their new or changed data is saved instantly—without a confirmation message—if the structure of the data form hasn't changed. For example, instant save works automatically unless any of these options is selected for the data form (because they change the structure of the data form):

- **Suppress missing data**
- **Suppress missing blocks**
- A business rule is selected to launch on **Save** (other than the default Calculate Form and Calculate Currencies business rules).

About Autosave

If the **Grid Property** option **Enable Autosave** is selected for a simple form, when planners move out of a cell, their changes are automatically saved, with no prompt or message. Cell values are aggregated to their parents, and the affected cells display with a green background. Also, with this option selected, planners can use Ctrl+Z to undo changes.

For autosave to work, the following suppress options must be turned off because they change the structure of the form:

- **Suppress missing data**
- **Suppress missing blocks**

For more information on this option and its dependent option, **Run Form Rules on Autosave**, see Setting Form Grid Properties.
Creating an Application

Create an application based on your planning needs. Reporting applications use basic application features. Standard applications use advanced custom features.

Related Topics

- About Creating an Application
- Selecting the Application Type
- Creating a Reporting Application
- Creating a Sample Application
- Creating a Standard Application
- Choosing Setup Options
- Taking the Product Tour

About Creating an Application

Before you create the application, we've provided some guidelines and best practices to help you determine your requirements and design your application so that it meets your business needs:

- See Best Practices for Designing Your Application.
- Click Take a quick tour. For more information, see Taking the Product Tour.

Watch this tutorial video to learn how to create an application.

Tutorial Video

To create an application:

1. Log in and select Start under Finance, and then select Planning & Budgeting.
2. Select the type of application you want to create. See Selecting the Application Type.

Related topics:

- Creating a Reporting Application
- Creating a Sample Application
- Creating a Standard Application
- Choosing Setup Options

Selecting the Application Type

Select an application type:
• **Standard**—Builds advanced applications for any business process. Select from two options:
  – **Sample**—Quickly and automatically creates a demo application using the sample Vision application that’s provided. See Creating a Sample Application.
  – **New**—Builds an advanced custom application. See Creating a Standard Application.

• **Enterprise**—Builds custom applications or uses predefined business processes to create applications for Financials, Workforce, Capital, and Projects. You can also build a Strategic Modeling solution. See Administering Planning Modules.

  **Note:**
  Only customers who purchase an Enterprise PBCS license or a PBCS Plus One Business Process option license can select the **Enterprise** application type. See Frequently Asked Questions for an explanation of the PBCS Plus One Business Process option license.

• **Reporting**—Builds a basic application that you can expand over time. See Creating a Reporting Application.

## Creating a Reporting Application

A Reporting application is a basic application with one cube and the core components that you need to get up and running quickly. If you later decide that you need more complex business logic, you can convert a Reporting application into a Standard or Enterprise application.

A Reporting application:
- Consists of one cube
- Allows you to add more dimensions
- Allows MDX member formulas only
- Doesn’t support business rules, map reporting, copy data, copy version, exchange rates, or currency conversion
- Can be converted into a Standard or Enterprise application

  **Note:**
  – To convert a Reporting application into a Standard application, see Converting to a Standard Application.
  – To convert a Reporting application into an Enterprise application, see Converting a Standard or Reporting Application to an Enterprise Application.

Default dimensions and characteristics for a Reporting application:
- **Account**—Only the root member is provided.
• Entity—Only the root member is provided.
• Period—Members are created based on the answers provided during setup. Users can add an alternate hierarchy later. The Beginning Balance and Year Total periods are provided.
• Years—Number of years, history, and future years are broken up based on the values chosen during application creation. An All Year parent is provided. More years can be added later. The default number of years is 10.
• Scenario—Five scenarios are provided: Plan, Actual, Forecast, Variance, and Variance Comments.
• Version—Three versions are provided: Base, What If, and Final. Hierarchical versions are allowed, for example, Base (child member)+What If (child member)=Final (parent member).

Note:
Generic members aren't added to the dimensions.

To create a Reporting application:

1. Click Reporting.
2. Name your application and enter a description.

Note:
Ensure that you adhere to the application naming restrictions outlined in Naming Restrictions.

3. The application provides setup options that you can select to help you create the initial framework for your planning process. If you need help deciding which options to choose, see Choosing Setup Options.
4. Review the summary screen, and, if the selections you made are correct, click Create.

After you create the application, populate it by performing these steps:
• Import metadata using the import feature. See Importing Metadata.
• Populate data by importing data from your source system. See Importing and Exporting Data.

Creating a Sample Application

Planning provides a Sample application, called Vision, that lets you quickly create an application with artifacts and data.
Creating a Standard Application

A Standard application is an advanced application with two cubes and the option to add more cubes. Complex business logic, such as business rules and allocations, is supported. You can't convert a Standard application into a Reporting application.

A Standard application:

- Allows you to add additional dimensions
- Enables sandboxes for the block storage cubes that are created when you create the application. (You can also enable sandboxes for custom cubes when you create the cube.)
- Supports custom calendars (up to weeks of the month)
- Supports business rules
- Allows member formulas
- Can't be converted into a Reporting application
- Can be converted into an Enterprise application

To create a Standard application:

1. Click **Standard**.
2. Click **New**.
3. Name your application and enter a description.

Note:

Ensure that you adhere to the application naming restrictions outlined in **Naming Restrictions**.
4. The application provides setup options that you can select to help you create the initial framework for your planning process. For descriptions of the setup options, see Choosing Setup Options.

5. Review the summary screen, and, if the selections you made are correct, click Create.

After you create a Standard application, populate it by performing these steps:

- Import application metadata using the import feature. See Importing Metadata.
- Populate application data by importing data from your source system. See Importing and Exporting Data.

Choosing Setup Options

Related Topics

- Setting Up the Calendar
  Set up the calendar for the application. You can set the base time period (frequency), start and end years, fiscal period, and enable rolling forecasts.

- Setting Up Currencies
  Specify the main currency for the application, and establish whether the application supports multiple currencies.

- Customizing the Cube Names
  Specify custom names for the input (block storage) and reporting (aggregate storage) cubes used in the application.

- Setting Up Custom Dimensions and Metadata
  Categorize data values into dimensions. You can create up to 13 user-defined, custom dimensions.

Setting Up the Calendar

Set up the calendar for the application. You can set the base time period (frequency), start and end years, fiscal period, and enable rolling forecasts.

The calendar establishes the application’s frequency, start and end years, first month of fiscal year (if monthly) or the start date of the first fiscal period (if weekly or quarterly), and lets you enable a rolling forecast and period duration.

For the fiscal year, set the fiscal year first month and specify whether the fiscal year starts from the same calendar year or the previous calendar year. You can later set up calculations based on the calendar year, for example, using formula expressions. When setting up formulas for an application, consider that formula expressions such as [TPDate] and [FirstDate] produce different results if the application starts in the same calendar year or previous calendar year.

To set up the calendar:

1. Select the Planning Frequency:
   - Monthly
   - Weekly
   - Quarterly
   - Custom
2. Select the **Start and End year** for the application.

3. Specify fiscal year information based on the **Planning Frequency**:
   - For **Monthly**, select the first month of the fiscal year
   - For **Weekly** and **Quarterly**, specify the start date of the first fiscal period

4. If you set the frequency to **Monthly**, select a **Weekly Distribution** option: **Even**, 445, 454, or 544.

   Weekly distribution sets the monthly distribution pattern, based on the number of fiscal weeks in a month. This selection determines how data in summary time periods spreads within the base time period. When users enter data into summary time periods, such as quarters, the value is distributed over base time periods in the summary time period.

   If you select a weekly distribution pattern other than **Even**, the application treats quarterly values as if they were divided into 13 weeks and distributes weeks according to the selected pattern. For example, if you select 5-4-4, the first month in a quarter has five weeks, and the last two months in the quarter have four weeks.

5. Decide whether to enable rolling forecasts and set the period duration. See About Rolling Forecasts.

### Setting Up Currencies

Specify the main currency for the application, and establish whether the application supports multiple currencies.

To set up the currency:

1. During application setup, select the main currency for the application.

2. Specify whether the application supports more than one currency, and if so, then select one of the following multiple currency options:
   - **Standard**—Creates the Currency dimension and a hidden dimension called HSP_Rates to store the exchange rates. See About Standard Multicurrency.
   - **Simplified**—Creates the Currency dimension and additional accounts are added to store the exchange rates. See About Simplified Multicurrency.

After the application is created, you can't change this option. Multiple currency support is available for level 0 members, regardless of base currency. The same currencies and exchange rates are used across cubes.

### About Simplified Multicurrency

In a Simplified multiple currency application, a single Currency dimension is added to the application and exchange rates are stored in the Accounts dimension.

Here is the workflow for working with a Simplified multiple currency application:

1. During application creation, select a main currency against which the entire application will refer to when a currency conversion is done.

2. After the application is created, in the Currency dimension, create a Currency member for each currency in the application. The Currency members you create are listed under **Input Currencies**. Data is entered or loaded in an input currency and data can be entered or loaded into multiple input currencies for any given combination; for example Entity and Account.
3. While creating a Currency member, you can specify whether the currency you're adding is also a reporting currency. Select the Reporting Currency option to make a currency a reporting currency.

   **Note:**
   Each application has one main currency, and is allowed to have many reporting currencies.

4. For simplified multiple currency applications with a custom aggregate storage cube, you must enable the Account, Period, and Currency dimensions for the custom aggregate storage cube. All dimensions in default aggregate storage cubes are enabled by default, but you must enable these dimensions manually for custom aggregate storage cubes. To enable Account, Period, and Currency dimensions for custom aggregate storage cubes, edit dimension properties for each dimension and select to enable them for the aggregate storage cube. See Editing Dimension Properties in the Simplified Dimension Editor.

5. Refresh the application. The Calculate Currency rule converts the input values from the current currency to the reporting currencies. Exchange rates are calculated from the input currency to the main currency of the application. Triangulation is used to calculate from an input currency to all reporting currencies.

6. Enter exchange rate data to the exchange rate form that was automatically created and seeded during application creation. The exchange rate form is called Exchange Rates to Main Currency; for example, Exchange Rates to USD. Enter the exchange rates by period for all input currencies against the main currency. You must enter the values across all intersections under which you wish to see the converted data.

   **Note:**
   You can also export the Exchange Rate Template to input and load currency exchange rates. See Importing Exchange Rates for a Simplified Multicurrency Application.

Exchange rates are stored in the Account dimension and they contain the ending and average rates used to convert foreign currencies into the main currency. These rates are in FX Rates-Average and FX Rates-Ending. These rates are locked and can't be deleted. Users with view permissions can view the rates for the currencies on the Exchange Rates to Main Currency form.

   **Note:**
   You can't move an Account member that is under an Exchange Rates account out from under Exchange Rates because the Exchange Rates account is locked. To move an Exchange Rate member out of the Exchange Rates account, you must delete the member and then re-add it to the new location.
7. There are two ways to view Reporting currency values:

• **Calculate Currencies** rule on a form—On the data input form, add the **Calculate Currencies** business rule. To add this rule, click the **Action** menu, and then select the **Business Rule** option. You can use the **Run After Save** or **Run Before Load** business rule properties. When data is entered for the input currency and the Calculate Currency rule is executed, the currency is converted and displays in the reporting currency form.

  **Note:**

  – Ensure that any forms created have the same intersection as that of the **Exchange Rates to Main Currency** form, with respect to Scenario, Version, and Year.

  – Ensure that both the input and converted forms have the same structure, the only difference between the forms being the currency.

  – Associate the **Calculate Currencies** rule to the forms where the input is being provided. You can set up forms to automatically run these rules when data in the form is saved. If not, users must manually launch the rule to convert data from one currency to the reporting currencies. After the rule is run, view the converted values by opening the form for which you want to see the converted values or by selecting that currency's member from the Currency dimension.

  – The **Calculate Currencies** rule is associated with forms in a block storage cube. Simplified currency conversion isn't supported for aggregate storage cubes.

  – You can restrict currency calculations in forms and batch currency rules based on the range defined for the scenario time period. In **Application Settings**, select **Yes** for **Enable currency calculation based scenario time period**.

• Create and launch a custom rule—To create a custom rule, from the Home page, click **Navigator** and then under **Create and Manage**, click **Currency Conversions**. For **Currency**, enter the reporting currency. Then enter the **Scenario**, **Version Type**, and **Version** details. Specify the **Years** for which you want to calculate the currency conversion rule. Click **Save**. To launch the rule, from the Home page, click **Rules**. To the right of the business rule, click **Launch**.

  **Note:**

  If you've enabled currency calculation based on the scenario time period, you cannot specify **Years** and you can only specify one **Scenario**.
About Standard Multicurrency

If you’re familiar with Planning, then you’re familiar with the Standard multiple currency approach. This is the approach that your current Planning applications use.

Selecting the standard multiple currency option during application creation creates two additional dimensions called Currency and HSP_Rates. You can add more currencies as members of the Currency dimension. The HSP_Rates dimension is a hidden dimension that stores exchange rates. This dimension includes these members and others that store currency rates:

- Hsp_InputValue—Stores data values
- Hsp_InputCurrency—Stores currency types for data values

For more information, see Hsp_Rates Dimension for Standard Multicurrency Applications.

For an account with a currency data type, you can choose from one of the following exchange rate types:

- Historical
- Average
- Ending

For more information, see Data Type and Exchange Rate Type.

Based on the exchange rate type, a user-defined attribute (UDA) is associated with the account that is used in the currency calculation. You can associate an entity to a currency. For example, when you associate the entity USA to the currency USD, the local currency in the form for the USA entity is USD.

You can create multiple exchange rate tables, each representing a different business scenario. Each scenario can be associated with only one exchange rate table.

You can run a calculate currency rule that generates the rule at runtime based on the members included in the form. This rule can be set to run when the form is saved or loaded, based on the requirement.

For more information about using the standard multiple currency approach, see Administering Dimensions.

Customizing the Cube Names

Specify custom names for the input (block storage) and reporting (aggregate storage) cubes used in the application.

You can customize the names for the cubes during application creation only. If no cube names are specified during application creation, then the input (block storage)
cube name defaults to “Plan1” and the reporting cube (aggregate storage) name defaults to the application name.

Caution:

After a cube is added, you can't rename it or delete it.

To customize the cube names:

1. In **Name of Input Cube**, enter a name for the data entry (block storage) cube.
2. In **Name of Reporting Cube**, enter a name for the reporting (aggregate storage) cube.

At this point in the application creation process, you can also enable sandboxes. For information about sandboxes, see **Managing Sandboxes**.

## Setting Up Custom Dimensions and Metadata

Categorize data values into dimensions. You can create up to 13 user-defined, custom dimensions.

The application includes two custom dimensions: Account and Entity. Use Account and user-defined dimensions to specify data to gather from planners. Use Entity to model the flow of planning information in the organization and establish the plan review path.

If you selected the Standard multicurrency option during application creation, the application includes the Hsp_Rates dimension for storing exchange rates. This dimension includes these members and others that store currency rates:

- **Hsp_InputValue**—stores data values
- **Hsp_InputCurrency**—stores currency types for data values

For detailed information about custom dimensions and Hsp_Rates, see **Editing Dimensions in the Simplified Dimension Editor**.

Note:

You have the option to either manually enter member names or you can set up Account, Entity, and custom dimension members by importing from metadata files. To import from metadata, you first export the metadata for each dimension to a flat file and then, on this window, import their metadata by browsing to that flat file. For instructions on creating the flat file, see **Creating the Import File**.

To set up custom dimensions:

1. To add a dimension or rename a prefilled dimension, to the right of **Dimension Type** in the **Dimension Name** column, enter the name of the dimension.
   
   Dimension names can have up to 80 characters and can include spaces.

2. For each dimension, enter its member names.
For example, for the Version dimension, you could enter Budget2014. You can add members later.

The Number of Records column displays the number of members either entered or in the flat file.

3. **Optional**: To import metadata for Account, Entity, or a custom dimension from a flat file, click **Browse**.

   Use this method for importing smaller dimensions (for example, 1,000 members or fewer); use the Import and Export method (described in Importing and Exporting Data and Metadata) for incremental updates or for loading larger dimensions.

   For instructions on creating flat files, see Creating the Import File. If the file has so many members that application performance would be impacted, an error displays and the number of records displayed is 0 (zero).

4. Click **Create**.

5. If you have more metadata to load, continue to Importing Metadata.

---

**Taking the Product Tour**

Clicking **Take a quick tour** links you to useful information about getting started with the application. It also provides best practices for determining your requirements and designing your application, tutorials for administering the application, and links you to the Help Center where the documentation library is located.

To take the product tour:

1. Click **Take a quick tour**.

2. Click the buttons at the bottom of the screen to page through the tour screens.
Setting Up Access Permissions

Assign access permissions to application artifacts such as dimensions, forms, dashboards, and task lists.

Related Topics
- About User and Role Management
- Application Artifacts That Can Be Assigned Permissions
- Types of Access Permissions
- Managing Permissions to Artifacts
- Reporting on Access Permissions
- Retrieving Changed Information from Access Control

About User and Role Management

The application employs several security layers to ensure security. Infrastructure components, which are implemented and managed by Oracle, create a secure environment for the application.

Business process-level security is ensured by using the following mechanisms that permit only authorized users to access the application:

- Single Sign-on (SSO)
- Role-based access to the application

Global roles are granted through the Oracle Cloud Identity Console. See Getting Started with Oracle Enterprise Performance Management Cloud for Administrators.

Access permissions are granted through the application interface and are described in this chapter.

Application Artifacts That Can Be Assigned Permissions

You can assign permissions to:

- Dimensions, including user-defined dimensions
**Note:**
You can assign permissions to members by selecting the dimension property **Apply Security**. If you omit or clear the **Apply Security** setting, all users can access the dimension's members. By default, the Account, Entity, Scenario, and Version dimensions are enabled for access permissions. Optionally, you can enable this option for Period, Years, and Custom dimensions.

- Launch privileges to rules
- Rule folders
- Forms
- Dashboards
- Infolets
- Form folders
- Dashboard folders
- Infolet folders
- Task lists

**Types of Access Permissions**

Access permissions include Read, Write, and None. You can also set who can launch which rules.

- **Launch**: Allow launch privileges

**Note:**
View user types have no Write permission to dimension members, so they can't launch rules having runtime prompts that include members, dimensions, member ranges, or cross-dimension runtime prompt types. They can, however, launch rules having runtime prompts of other types (for example, date type).

- **No Launch**: Disallow launch privileges

**Note:**
If a user inherits Launch permission to a rule by belonging to a group, and is also assigned No Launch permissions by belonging to another group, the more restrictive No Launch assignment takes precedence.

You can specify access permission for individual users and each group. When you assign a user to a group, that user acquires the group's access permissions. If an individ-
ual's access permissions conflict with those of a group the user belongs to, user access permissions take precedence.

Inheriting Permissions

Inheritance determines the user or group's access permissions. You can specify an attribute that causes the children or descendants of that member to inherit its permissions. Assigned permissions take precedence over inherited permissions. You can include or exclude the member from the permissions setting.

Table 3-1 Options for Inheriting Access Permissions

<table>
<thead>
<tr>
<th>Inheritance Option</th>
<th>Permission Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td>Only to the currently selected member</td>
</tr>
<tr>
<td>Children</td>
<td>To all children members in the level below the currently selected member</td>
</tr>
<tr>
<td>iChildren</td>
<td>To the currently selected member and all children members in the level below it</td>
</tr>
<tr>
<td>Descendant</td>
<td>To all descendant members below the currently selected member</td>
</tr>
<tr>
<td>iDescendant</td>
<td>To the currently selected member and all descendant members below it</td>
</tr>
</tbody>
</table>

How Permissions are Evaluated

When evaluating permissions, the application gives precedence in this order:

1. Role-level security. Users with the Service Administrator role have permissions to all application elements.
2. For Power User and Planner user types, permissions that are specifically assigned to users.
3. Permission assignments that are acquired by belonging to a group.

4. Parent-level assignments (for example, to parent members or folders).

Note:

If one member belongs to two groups with different permissions assigned to group members, the least restrictive permission takes precedence. For example, if one group assigns the member Read permission and another group assigns the same member Write permission, Write takes precedence. However if one of the groups assigns no permission (None) to its members, None takes precedence over Read and Write.

Managing Permissions to Artifacts

Related Topics

- About Assigning Permissions to Artifacts, Rules, and Folders
- Adding, Changing, and Removing Permissions to Artifacts and Folders
• Adding, Changing, and Removing Permissions to Rules and Rules Folders

About Assigning Permissions to Artifacts, Rules, and Folders

Administrators can assign permissions to artifacts (forms, dashboards, and infolets), rules, and folders.

For information on assigning permissions to dimension members, see Types of Access Permissions and Assigning Access to Dimension Members.

Principles:

• **Forms, Dashboards, and Infolets:**
  – Planners and Power Users can view or enter data only into forms to which they have permissions (and can work only with members to which they have permissions).
  – Service Administrators and Power Users can design forms, dashboards, and infolets.
  – Power Users can access forms, dashboards, and infolets they created or to which a Service Administrator assigned them permissions.
  – Service Administrators have Write permissions to all dimension members and to all forms, dashboards, and infolets.

• **Rules:**
  – Planners and Power Users can see and launch only rules to which they are assigned Launch permission.
  – Rulesets inherit launch permissions from the rules included in the ruleset.

• **Folders:**
  – Planners who are assigned permissions to a folder can access the items in that folder, unless they are assigned more specific permissions. Likewise, planners have Launch permission to the Calculation Manager rules in folders to which they are assigned permissions, unless they are assigned more specific permissions.
  – When you assign permissions to a folder, all folders under it inherit that permission.
  – If you assign specific permissions (for example, None or Write) to a folder, that permission takes precedence over its parent folder’s permissions. For example, if a user has Write permission to Folder1 that contains Folder2 to which the user has None permission, the user can open Folder1, but doesn’t see Folder2.
  – If you assign specific permissions (for example, Launch) to a Calculation Manager folder, that permission takes precedence over its parent folder’s permissions. For example, if a user has Launch permission to RulesFolder1 that contains RulesFolder2 to which the user has No Launch permission, the user can open RulesFolder1, but doesn't see RulesFolder2.
  – If a user has None permission to a folder called Folder1 that contains a form called Form1 to which the user has Write permission, the user can see Folder1 and Form1.
If a user has No Launch permission to a Calculation Manager folder called RulesFolder1 that contains a rule called Rule1 to which the user has Launch permission, the user can see RulesFolder1 and Rule1.

For procedures, see Adding, Changing, and Removing Permissions to Artifacts and Folders and Adding, Changing, and Removing Permissions to Rules and Rules Folders.

Adding, Changing, and Removing Permissions to Artifacts and Folders

The permissions assigned to artifacts (forms, dashboards, and infolets) and folders refer to the ability to modify the artifact and folder definitions and not the underlying data.

To assign permissions to artifacts and folders:

1. Click Data or Dashboards or Infolets.
2. Click the Actions icon next to the artifact or folder, and then click Assign Permission.

   Note:

   You can assign permissions to only one artifact or folder at a time.

3. Click Permissions.
4. Perform a task:
   - To add permissions, click Add User/Group, and then select from the list of available users and groups.
   - To edit the type of permissions, select an option:
     - Click Read to allow the selected users and groups to view the artifact or folder in the listing, but not allow them to create, edit, or delete the artifact or folder information.
     - Click Write to allow the selected users and groups to view the artifact or folder in the listing, and to create, edit, or delete the artifact or folder information.
     - Click None if you don't want the selected users or groups to view the artifact or folder in the listing.
   - To remove permissions, click × next to the user or group.

Adding, Changing, and Removing Permissions to Rules and Rules Folders

To assign permissions to rules and rules folders:

1. Click Rules.
2. Select the rule or rule folder, click , and then click Assign Permission.
Note:
You can assign permissions to only one rule or folder at a time.

3. Click Permissions.
4. Perform a task:
   - To add permissions, click Add User/Group, and then select from the list of available users and groups.
   - To edit the type of permissions, select an option:
     - Click Launch to allow the selected users and groups to launch the selected rules.
     - Click No Launch to prevent the selected users and groups from launching the selected rules.
   - To remove permissions, click next to the user or group.

Reporting on Access Permissions

You can view current access permissions and print reports.

To report on current access permissions for users and groups in the application:

1. From the Home page, click Navigator, and then under Monitor and Explore, click System Reports.
2. Click Access Control.
3. On Select User or Group, select from the available options.
4. From the left Available panel, select and move users or groups on which to report to the Selected panel.
5. From the left Available Objects panel, select and move objects on which to report to the Selected Objects panel.
6. Select reporting options:
   - For Show Matching Access of Type, select the access to view: Read, Write, or None.
   - For Group the Results By, select how to view the report: Users or Objects.
   - From the Report Type sections, select Assigned Access or Effective Access.
### Table 3-2 Access Report Types

<table>
<thead>
<tr>
<th>Report Type</th>
<th>Description</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assigned Access</td>
<td>Summarizes access permissions that administrators assign</td>
<td>Specify whether access permissions are assigned by member selection relation or group membership:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Show matching access of relation</strong>: Member, Children, Children (inclusive), Descendants, or Descendants (inclusive).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Show Inherited From Group</strong>: Shows access permissions inherited by users in a group.</td>
</tr>
<tr>
<td>Effective Access</td>
<td>Summarizes access permission assignments as the application evaluates them (for example, by member selection relation, such as children, or group membership). This is useful if there are conflicts in access permissions.</td>
<td>Describe the origin of the effective access permissions by selecting <strong>Show effective access origin</strong>. For example, a user named JSomebody may be assigned Write access permissions to Entity1 and belong to a group named Sales that is assigned Read access permissions to Entity1. This setting would show that JSomebody has Write access permissions to Entity1 because individual assigned access permissions supersede access permissions inherited by group membership.</td>
</tr>
</tbody>
</table>

**Note:**
Effective Access reports can't be generated for groups.

7. Click **Create Report**.

### Working with Access Permissions Reports

The report on access permissions displays in Adobe Acrobat. You can use the Adobe Acrobat toolbar to work with the report.

### Retrieving Changed Information from Access Control

Changes to group names or group membership from Access Control are not reflected in the application. To apply changes, you must reimport security artifacts.

To synchronize information after modifying group properties:

1. Access the environment as a Service Administrator.
2. Click **Tools**, and then **Migration**.
3. Export Security artifacts:
   a. Click **Planning**, and then select **Security**.
   b. Click **Export**.
   c. Enter a file name and click **OK**.
   d. Review the Migration Status Report to ensure that the export is successful.
   e. Click **Close**.
4. Import the snapshot that you created:
a. On the Migration page, click Snapshots.

b. Under Actions, click *** next to the snapshot that you created in the preceding step, and then click Import.

c. Click OK.

d. Review the Migration Status Report to ensure that the import is successful.

e. Click Close.
Managing Applications

View and manage applications, cubes, and dimensions using the application console.

Related Topics
- Application Overview
  Start here to learn about managing applications, dimensions, and dimension members.
- Refreshing the Application
  Refresh the application whenever you make changes to the application structure.
- Managing Cubes
  Use the cube editor to let you view or add cubes for Standard applications.
- Managing Dimensions
  Start here to learn how to manage dimensions and dimension members.
- Setting User Preferences
- Administering Variables
- Viewing Activity Reports
  Monitor user activities with activity reports.
- Importing and Exporting Data and Metadata
  Move data and metadata in and out of the application using Planning.
- Importing Exchange Rates for a Simplified Multicurrency Application
  Import exchange rates for a simplified multiple currency application using a template.
- Creating and Refreshing Application Databases
  Things you need to know before you refresh the application database.
- Adding an Aggregate Storage Outline to an Application
  Understand what aggregate storage is and how to use it in an application.
- Converting to a Standard Application
  To increase the number of cubes and use the logic required to support more extensive calculations, convert a Reporting application to a Standard application.
- Converting a Standard or Reporting Application to an Enterprise Application
  Learn about converting an application in this topic.
- Conversion Considerations
- Removing an Application
  Proceed with caution when removing an application.
- Setting the Daily Maintenance Process Time
  Understand how to determine and set the most convenient time to start the hour-long daily maintenance process.
- Uploading and Downloading Files Using the Application Inbox and Outbox
  Use the Inbox/Outbox Explorer to upload files to the server or download them to your computer. You can also use it to schedule import and export jobs.
Application Overview

Start here to learn about managing applications, dimensions, and dimension members.

An application is a related set of dimensions and dimension members used to meet a set of planning needs. Each application has its own accounts, entities, scenarios, and other data elements.

To create an application, see Creating an Application.

After an application is created, you can view and manage it using the application console. To view the application console, click Application, and then click Overview.

The application console shows the application properties (including the application type) and application statistics, such as the number of tasks, plans, rules, and approval hierarchies in your application. It also lists the cubes (for Standard applications) and dimensions (for Reporting applications) used by the application.

- To view the application overview, click Overview.
- To manage cubes, click Cubes. See Managing Cubes.
- To manage dimensions, click Dimensions. See Managing Dimensions.
- To view activity reports, click Activity. See Viewing Activity Reports.

Refreshing the Application

Refresh the application whenever you make changes to the application structure.

Changes made to the application are not reflected to users performing data entry and approvals tasks until the application is refreshed. For example, when you modify properties of a dimension member, add a member, or change access permissions, these changes are reflected to users after you refresh the application.

To refresh the application:

1. Click Application, and then click Overview.
2. Click Refresh.

Note:

The application is automatically refreshed during an application upgrade. If any refresh errors occur during an application upgrade, you can view the errors in the Jobs console. These errors are captured in the job called Content Upgrade. See Viewing Pending Jobs and Recent Activity.
Managing Cubes

Use the cube editor to let you view or add cubes for Standard applications.

You can add up to seven total cubes to a Standard application. After a cube is added, it will behave like any other cube.

Watch this tutorial video to learn how to manage cubes.

Tutorial Video

• Viewing and Managing Cubes
• Adding Cubes
• Clearing Cubes
• Improving Cube Performance

Viewing and Managing Cubes

To view and manage cubes:

1. Click Application, and then click Overview.
2. Click Cubes.
3. Perform one task:
   • To view the details of a cube, click the name of the cube that you want to view.
   • To add a cube, click Create, then complete the cube details, and then click Done.

Note:

When you create a new block storage cube, you can select the option Enable Sandboxes. Selecting this option also provides the Enable Sandboxes option for Version members. See Managing Sandboxes.

For cube details, see Adding Cubes.

Note:

You can rename a cube only during application creation. After a cube is added, you can't rename or delete it.
Adding Cubes

You can add up to seven total cubes—three block storage cubes and three corresponding aggregate storage cubes plus one consolidating aggregate storage cube, each with a unique aggregate storage application name.

After a cube is added, it behaves like any other application cube. If the cube maps to an aggregate storage database, then aggregate storage limitations apply.

Note:

For an aggregate storage cube, you must specify an application name to contain the database since an aggregate storage database must reside in its own application.

Clearing Cubes

Related Topics

- About Clearing Cubes
- Creating Clear Cube Jobs
- Starting and Deleting Clear Cube Jobs
- Scheduling Clear Cube Jobs

About Clearing Cubes

The application enables you to clear specific data within input and reporting cubes. You can start the Clear Cube job right away or schedule it to run later.

Note:

The Clear Cube job deletes the data you specify within input and reporting cubes, but it doesn't delete the application definition in the application's relational tables.

Creating Clear Cube Jobs

To create a Clear Cube job:

1. Click Application, and then click Overview.
2. Click Actions, and then Clear Cube.
3. On the Clear Cube page, click Create, complete the job name, select the cube, then select a clear option:

   - Clear All—For both block storage and aggregate storage cubes, delete all data within a cube that is associated with the entities you select:
• For block storage cubes only:
  – Upper-level blocks—Delete data in upper-level blocks
  – Non-input blocks—Delete data in non-input blocks
  – Dynamic blocks—Delete data in dynamic blocks
  – Empty blocks—Delete blocks with #MISSING values

• For aggregate storage cubes only:
  – All Aggregations—Delete all aggregate views
  – Partial Data (Basic Mode)—Clear the data in the specified region. You can select the point of view using the member selector.

  * Supporting Details
  * Comments
  * Attachments
  * Essbase Data Logical—In which the input cells in the specified region are written to a new data slice with negative, compensating values that result in a value of zero for the cells you want to clear. The process for logically clearing data completes in a length of time that is proportional to the size of the data being cleared. Because compensating cells are created, this option increases the size of the database.

  * Essbase Data Physical—In which the input cells in the specified region are physically removed from the aggregate storage database. The process for physically clearing data completes in a length of time that is proportional to the size of the input data, not the size of the data being cleared. Therefore, you might typically use this method only when you need to remove large slices of data.

  – Partial Data (Advanced Mode)—Clear Essbase data only by using an MDX expression in the text box provided.

    * Essbase Data Logical—In which the input cells in the specified region are written to a new data slice with negative, compensating values that result in a value of zero for the cells you want to clear. The process for logically clearing data completes in a length of time that is proportional to the size of the data being cleared. Because compensating cells are created, this option increases the size of the database.

    * Essbase Data Physical—in which the input cells in the specified region are physically removed from the aggregate storage database. The process for physically clearing data completes in a length of time that is proportional to the size of the input data, not the size of the data being cleared. Therefore, you might typically use this method only when you need to remove large slices of data.

4. Click Save and Close.
5. To view the job on the Clear Cube listing page, click Refresh.
To start the Clear Cube job right away or to delete a job, see Starting and Deleting Clear Cube Jobs.

To schedule a Clear Cube job to run immediately, later, or at intervals, go to the Jobs page. See Scheduling Jobs.

Starting and Deleting Clear Cube Jobs

To start a Clear Cube job right away or to delete a job:
1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then **Clear Cube**.
3. On the **Clear Cube** listing page, click next to the Clear Cube job that you want to start or delete, and then select one of the following options:
   - Click **Submit** to start the job right away.
   - Click **Delete** to delete the job.

Scheduling Clear Cube Jobs

You can schedule a Clear Cube job to run later or at intervals. To schedule a Clear Cube job, go to the Jobs page. See Scheduling Jobs.

Improving Cube Performance

The application enables you to schedule jobs that improve cube performance. These jobs include:

- **Restructure Cube**—Performs a full restructure of a block storage cube to eliminate or reduce fragmentation. This will also remove empty blocks. Running this job won't push any changes from Planning to Essbase.

- **Compact Outline**—Compacts the outline file of an aggregate storage cube. Compaction helps keep the outline file at an optimal size. Compacting the outline doesn't clear the data. Running this job won't push any changes from Planning to Essbase.

- **Merge Data Slices**—Merges incremental data slices of an aggregate storage cube. Fewer slices improve a cube's performance. You can merge all incremental data slices into the main database slice or merge all incremental data slices into a single data slice without changing the main database slice. You can optionally remove cells that have a value of zero.

- **Optimize Aggregation**—Generates optimized views based on collected query tracking information in an aggregate storage cube. This job has two actions:
  - **Enable Query Tracking**—Records a meaningful set of queries, and then uses the recorded query data to select the most appropriate set of aggregate views to materialize for that database. You must enable query tracking and allow it sufficient time to collect data before executing the aggregation process based on query data.

    Once enabled, query tracking continues until query tracking is disabled, the cube is reloaded or refreshed, the outline is compacted, the application is shut down, or additional aggregate views are materialized for the database. Query tracking does not resume automatically after these actions occur.
The following actions clear the stored query tracking information from the aggregate storage cube, but they do not disable query tracking:

- Merging incremental data slices
- Clearing data
- Clearing existing aggregations
- Clearing partial data
- Loading data into the aggregate storage cube

**Note:**

Users can stay logged in during these actions.

To create a job that improves cube performance, see **Scheduling Jobs**.

---

**Manage Dimensions**

Start here to learn how to manage dimensions and dimension members.

Dimensions categorize data values. These dimensions are provided with the application: Account, Entity, Scenario, Version, Period, Currency (for a multicurrency application), and Years.

Dimensions on the **Dimensions** page are listed in order of precedence. The order of dimensions is critical for the structure and performance of an application. The order of
your dimensions determines how your data calculations will perform. **Evaluation Order** enables you to specify which data type prevails when a data intersection has conflicting data types. For example, if Account members are set to the Currency data type, and Product members are set to the Smart List data type, you can set whether the Currency or Smart List data type prevails at an intersection.

![Note:](image)

For a Smart List to be viewable in forms, the dimension to which the Smart List type member belongs must be first in the evaluation order. See **Working with Smart Lists**.

To manage dimensions:

1. Click **Application**, click **Overview**, and then click **Dimensions**.
2. Perform a task:
   - To filter the dimensions by cube, see **Filtering the Dimension View by Cube**.
   - To view and edit the properties of dimensions and their members, click the name of the dimension. See **Editing Dimensions in the Simplified Dimension Editor**.
   - To reorder the dimensions, click the up or down arrow in the **Order** column next to the dimension that you want to move.
   - To set a different evaluation order, click the **Evaluation Order** field next to the dimension that you want to update.
   - To rename a dimension, click the name of the dimension that you want to rename, and then enter a new name in the **Dimension** field.
   - To create a dimension, click **Create**. See **Creating Dimensions**.
   - To import metadata, click **Import**. See **Importing Metadata**.
   - To export metadata, click **Export**. See **Exporting Metadata**.

**Filtering the Dimension View by Cube**

For a Standard application, you can filter the dimension view by cube. When you select a cube, only dimensions used in that cube are displayed.

To filter the dimension view by cube:

1. Click **Application**, click **Overview**, and then click **Dimensions**.
2. Click the down arrow to the right of **Cube** to filter the list of dimensions by cube.

By default, the application displays all cubes for a standard application. If the dimension list is filtered, the application displays only the dimensions used in the selected cube.

Filtering by cube (rather than viewing all cubes) also enables you to view these details about a dimension:

- Order of precedence
- Dimension density
• Evaluation order
  For information about editing dimensions, see Editing Dimensions in the Simplified Dimension Editor.

Creating Dimensions

Creating a dimension adds it to the list of dimensions in the dimension view.

To create a dimension:
1. Click Application, click Overview, and then click Dimensions.
2. Click Create.
3. On the Create Dimension page, enter or select values for the properties that are displayed.
   For descriptions of the properties, see Editing Member Properties in the Simplified Dimension Editor.
   For information about editing dimensions, see Editing Dimensions in the Simplified Dimension Editor.

Setting User Preferences

Administrators specify defaults for the current application. However, users can override these application defaults by setting preferences to control many aspects of the application, such as their profile photo and how numbers display in the application.

To learn more about user preferences options, see Setting Your Preferences in Working with Planning.

Administering Variables

Related Topics
• Working with Substitution Variables
• Working with User Variables

Working with Substitution Variables

Related Topics
• About Substitution Variables
• Creating and Assigning Values to Substitution Variables
• Deleting Substitution Variables

About Substitution Variables

Substitution variables act as global placeholders for information that changes regularly. For example, you could set the current month member to the substitution variable CurMnth so that when the month changes, you need not update the month value manually in the form or the report script. You create and assign values to substitution varia-
variables within the application. These substitution variables are then available in the application when you select members for a form.

For more information about selecting substitution variables, see Selecting Substitution Variables as Members.

Note:
If you migrated an application that uses Global variables, you can view, but not edit, them in the application.

Creating and Assigning Values to Substitution Variables

To create a substitution variable and assign values to it:

1. From the Home page, click Tools, and then click Variables.
2. Select the Substitution Variables tab.
3. Click .
   A new variable is added to the bottom of the list.
4. Scroll to the bottom of the list, click the down arrow next to the Cube name for the newly added variable, and then select the cube.
5. For Name, enter the name of the substitution variable (up to 80 characters).
6. For Value, enter a value for the substitution variable (up to 255 characters).
   You can specify a range of values when defining substitution variables; for example, FY16:FY18.

Note:
You can define a substitution variable range for base members and parent members. For base members, you can use a single colon (:) or a double colon (::) as delimiters; for example, SunEnT 110::111. However for parent members, you can only use a single colon (:).

7. Click Save.

Deleting Substitution Variables

To delete substitution variables:

1. From the Home page, click Tools, and then click Variables.
2. Select the Substitution Variables tab.
3. Next to the substitution variable you want to delete, click .
4. To confirm deletion, click Yes.
Working with User Variables

Related Topics

- About User Variables
- Managing User Variables
- Creating User Variables
- Setting Limits for User Variables on Forms
- Deleting User Variables

About User Variables

User variables act as filters in forms, enabling planners to focus only on certain members, such as a department. Before you can associate a user variable with a form, you must create the user variable. When you create forms with user variables, planners must first select values in preferences for the variable before opening forms. Afterward, planners can change the variable on the form only if it's a dynamic user variable. Otherwise, they must continue to set the variable in preferences. For example, if you create a user variable called Division, planners must select a division before working in the form.

Tip:

Attributes are another way to filter forms. But instead of explicitly filtering by an attribute (like Red), you can create a user variable for the attribute dimension, and then use the user variable as the filter. Then you can enable the user variable as a dynamic user variable which allows users to change the value of the filter at runtime. This is a useful technique that allows for dynamic filtering.

The first time planners select a variable for a form, they do so in preferences. Afterward, they can update the variable in preferences or in the form. For information about selecting user variables as members, see Selecting User Variables as Members.

Managing User Variables

You can set user variables to limit the number of members displayed on a form, helping users focus on certain members. For example, if you create a user variable called Division for the Entity dimension, users can select a member for their own division. You can create any number of user variables for each dimension (including attribute dimensions), and select user variables for any axis in the form. See Defining Simple Form Page and Point of View.

The typical sequence of steps:

1. If necessary, create the appropriate parent-level members in the dimension outline.
2. Define user variables for each dimension you want users to be able to filter (including attribute dimensions).
Creating User Variables

To create user variables:

1. From the Home page, click Tools, and then click Variables.
2. Select the User Variables tab.
3. Click .
   A new variable is added to the bottom of the list.
4. Scroll to the bottom of the list and click the down arrow next to the Dimension name for the newly added variable. For Select Dimension, select the dimension for which to create a user variable (including attribute dimensions).
5. In the User Variable Name column, enter the name of the user variable.
6. Click to select dimension members.

**Note:**

The Use Context setting means the value can't be chosen by the user from user preferences. Instead, the value is passed from another form based on its Point of View context and the value will change dynamically based on the context of the form. However if the user variable is used elsewhere, such as in rows and columns, you must create a different user variable so the user can set its own value for it because it won't be set dynamically.

7. Click OK.

You can now associate the user variable with a form. See Selecting User Variables as Members. You can also set limits for user variables on forms. See Setting Limits for User Variables on Forms.

Afterward, planners can select members for the user variable. See Working with Planning.

Setting Limits for User Variables on Forms

You can limit which members users can see or select on individual forms by limiting their member selections for a user variable. The user variable limit applies the restriction to all Planners at the form level, regardless of their access permissions. In addition
to setting the limits by members, administrators can select limits by function (for example, "IChildren(Existing Employees)").

To set up user variable limits:

1. During form design, select **Enable Dynamic User Variables** to allow dynamic user variables in the form.
   
   See **Setting Form Precision and Other Options**.

2. Create a user variable.
   
   See **Creating User Variables**.

3. In the **User Variable Definition** field on the **Variables** page under the **User Variables** tab, click  and add or remove members to set limits.

   To select members by function, click  and select from the available options.

4. In the form designer, assign the user variable to the page axis.
   
   See **Defining Simple Form Page and Point of View**.

### Deleting User Variables

To delete user variables:

1. From the Home page, click **Tools**, and then click **Variables**.

2. Select the **User Variables** tab.

3. Next to the user variable you want to delete, click  .

4. To confirm deletion, click **Yes**.

### Viewing Activity Reports

Monitor user activities with activity reports.

Activity reports are automatically generated every day and enable administrators to understand application usage over a period of five, ten, thirty, or sixty days. They help streamline application design by identifying calculation scripts and user requests that impact application performance. Information provided in these reports include:

- User information such as number of users who signed into the application each day, each hour, last seven days, and last 30 days
- Percentage of requests that took more than 10 seconds to complete
- Ranking of requests by duration
- Ranking the performance of requests that took more than 10 seconds to complete
- Number of application design changes that occurred over the reporting period
- Number and duration of user interface requests per hour
- Number and duration of Essbase operations per hour
- CPU and memory usage by hour

To view activity reports:
1. Click **Application**, and then click **Overview**.
2. Click **Activity Reports**.
3. Click the down arrow next to **Reports from last** to filter the list of reports.
4. To view the report details, click the name of the report that you want to view.

## Importing and Exporting Data and Metadata

Move data and metadata in and out of the application using Planning.

### Related Topics

- **Importing Metadata**
  You can import metadata from a flat file in a comma-delimited, tab-delimited, or other format.

- **Exporting Metadata**
  You can export metadata to a flat file in a .csv (comma-delimited) or .txt (tab-delimited or other delimiter character) format.

- **Importing and Exporting Data**

## Importing Metadata

You can import metadata from a flat file in a comma-delimited, tab-delimited, or other format.

These artifacts are supported in the imports:

- Dimensions
- Smart Lists
- Exchange rate tables

To import metadata, perform these tasks:

- Create an import file for each artifact that you want to import. See [Creating the Import File](#).
- Load the import file or files (you can import multiple dimension files at the same time). See [Loading the Import File](#).

Watch this tutorial video to learn how to import metadata.

![Tutorial Video](#)

## Creating the Import File

### Related Topics

- **About the Import File**
- **Example Entity Dimension Import File**
- **Other Supported Delimiter Characters**
About the Import File

Before you begin loading, you must create an import file for each artifact you want to import (dimensions, smart lists, and exchange rate tables). The import file must contain a header record, and then below the header record, a list of metadata records that you want to import or update. The file format can be .csv (comma-delimited) or .txt (tab-delimited or other delimiter characters).

1. A header record, the first line in the file:
   - Lists the dimension and any member properties used by subsequent metadata records; the header record and subsequent records don't need to include all properties; properties that are not included are inherited from the corresponding parent's default property value
   - Is case sensitive
   - Can list properties in any order, as long as the subsequent metadata records are in the same order
   - Can use comma or tab delimiters. Other delimiter characters can be used if they are supported and the same delimiter character is used throughout the file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.

2. After the header record, a list of metadata records that you want to import or update. Each metadata record contains a delimited list (comma, tab, or other) of property values that matches the order designated in the header record. A metadata record can skip a property that is specified in the header record; in this case, the default property is assumed.

For an example import file, see Example Entity Dimension Import File.

Example Entity Dimension Import File

<table>
<thead>
<tr>
<th>Entity,Parent,Data Storage,Plan Type (Plan1),Plan Type (Plan2),Plan Type (Plan3),Data Type,Base Currency, Alias, default identity, Never share,TRUE,TRUE,TRUE,unspecified,USD,unspecified entity, AllEntities, Never share,TRUE,TRUE,TRUE,unspecified,USD,Total Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000,AllEntities, Never share,TRUE,TRUE,TRUE,unspecified,USD,Administration and Finance, 1011,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Office of the Secretary of Administration and Finance, 1012,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Bureau of State Office Buildings, 1101,1012, Never share,TRUE,TRUE,TRUE,unspecified,USD,Finance and Administration, 1102,1012, Never share,TRUE,TRUE,TRUE,unspecified,USD,State Office Buildings, 1103,1012, Never share,TRUE,TRUE,TRUE,unspecified,USD,Safety and Security, 1104,1012, Never share,TRUE,TRUE,TRUE,unspecified,USD,State House ADA,Communications Access, 1105,1012, Never share,TRUE,TRUE,TRUE,unspecified,USD,State House Events, 1013,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Civil Service Commission, 1014,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Department of Revenue, 1106,1014, Never share,TRUE,TRUE,TRUE,unspecified,USD,Child Support Enforcement, 1107,1014, Never share,TRUE,TRUE,TRUE,unspecified,USD,Division of Local Services, 1108,1014, Never share,TRUE,TRUE,TRUE,unspecified,USD,Appeal Tax Board, 1015,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Developmental Disabilities Council, 1016,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Division of Administrative Law Appeals, 1017,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,State Research, 1018,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Executive Office for Administration and Finance, 1019,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,George Fingold Library, 1020,1001, Never share,TRUE,TRUE,TRUE,unspecified,USD,Group Insurance Commission</td>
</tr>
</tbody>
</table>

In this example, the import file loads an Entity dimension with the required header record and three data records. This example is comma-delimited. The header record
specifies the member to be imported (Entity), the parent member (Parent) into which to import the member, and the Data Storage property to assign to the member.

Entity, Parent, Data Storage

e1, Entity,
e2, ,
e1, e2, Shared

Using this import file would result in this outline, assuming that no other members exist:

Entity
e1
e2

e1(Shared)

The first data record (e1, Entity) imports Entity member e1 as a child under the root member Entity. Unspecified values assume the default. For example, if data storage isn't specified, it assumes the default value, Never Share. The next data record (e2, ,) imports Entity member e2 under the dimension root member because no parent is specified, and sets data storage to Never Share. The last data record (e1, e2, Shared) imports a shared member of e1 under member e2, and sets data storage to Shared.

Other Supported Delimiter Characters

In addition to commas and tabs, the application supports these delimiter characters in import and export files:

- tilde (~)
- grave accent (’)
- exclamation point (!)
- number sign (#)
- question mark (?)
- dollar sign ($)
- percent sign (%)
- caret (^)
- ampersand (&)
• asterisk (*)
• parentheses ( )
• hyphen-minus (-)
• plus (+)
• colon (:)
• semicolon (;)
• angle brackets (< >)
• backslash (\)
• forward slash (/)
• vertical bar ( | )
• apostrophe (’)
• braces ({})
• underscore (_)
• brackets ([ ])
• at sign (@)
• period (.)

Only one character is supported for use as a delimiter. For example, one vertical bar ( | ) is supported, but two vertical bars ( || ) are not supported.

⚠️ Caution:

Not all of the characters listed can be used for all import and export scenarios. Note the following exceptions.

Import and Export Metadata Delimiter Exceptions

Don't use these delimiter characters in metadata import and export files.

Table 4-1  Import and Export Metadata Delimiter Exceptions

<table>
<thead>
<tr>
<th>Delimiter Character</th>
<th>Reason for Exception</th>
</tr>
</thead>
<tbody>
<tr>
<td>double quotation mark (”)</td>
<td>Creates an empty file</td>
</tr>
<tr>
<td>plus (+)</td>
<td>Causes an error if the metadata import file contains consolidation properties that use these characters</td>
</tr>
<tr>
<td>minus (-)</td>
<td></td>
</tr>
<tr>
<td>forward slash (/)</td>
<td></td>
</tr>
<tr>
<td>percent sign (%)</td>
<td></td>
</tr>
<tr>
<td>angle brackets (&lt; &gt;)</td>
<td>Causes an error if a property uses the value &lt;none&gt;</td>
</tr>
</tbody>
</table>
Note:
Any character that conflicts with a character in a member name will cause an error.

Import and Export Data Delimiter Exceptions

Table 4-2 Import and Export Data Delimiter Exceptions

<table>
<thead>
<tr>
<th>Delimiter Character</th>
<th>Reason for Exception</th>
</tr>
</thead>
<tbody>
<tr>
<td>apostrophe (')</td>
<td>Causes an error if used in a cube name</td>
</tr>
<tr>
<td>parentheses ( )</td>
<td>Causes an error if used in a data import file</td>
</tr>
<tr>
<td>double quotation mark (&quot;)</td>
<td>Creates an empty file</td>
</tr>
<tr>
<td>hyphen-minus (-)</td>
<td>Causes an error if used in a data import file</td>
</tr>
</tbody>
</table>

Loading the Import File

To load the import file:

1. Create an import file for each artifact (dimensions, smart lists, and exchange rate tables) that you want to import. See Creating the Import File.

2. Click Application, and then click Overview.

3. Click Dimensions, and then click Import.

4. On the Import Metadata page, click Create.

5. Select the location of the import file or files:
   - **Local**—Loads the import file or files from a location on your computer. For Import File, click Browse to select the import file on your computer for the artifact you’re importing.
   - **Inbox**—Loads the import file or files from the server. Enter the name of the file in Import File. See Uploading and Downloading Files Using the Application Inbox and Outbox.

6. For **File Type**, select Comma delimited, Tab delimited, or Other.
   - If Other is selected, enter the delimiter character that is used in the import file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.

7. Select **Clear Members** to delete members not explicitly specified in the load file before performing the import.
Note:

• Any member not specified is deleted from the application outline after importing the dimension unless it's an ancestor of a member that was specified, or it's a base member of a shared member that was specified.

• If Clear Members isn't selected, the import will only add or update existing members. Oracle recommends loading the metadata file without selecting Clear Members to make sure the file loads successfully. Then, select Clear Members and execute the import again.

8. **Optional:** If the location selected is Inbox, click **Save as Job** to save the import operation as a job, which you can schedule to run right away or at a later time. You can't select this option if Local is selected. See **Managing Jobs**.

Note:

• Saving an import operation as a job is useful to batch a load sequence; for example, import metadata, then import data, then run rules when the data load is complete.

• You can run up to five import jobs at one time.

9. **Optional:** If the location selected is Local, click **Validate** to test whether the import file format is correct.

10. Click **Import** to run the import operation.

If the cubes listed in the import file don't match the cube names in the application, the application will ignore unrecognized cube columns.

If the import fails, a Failed status will display in the Recent Activity column. Click to view the detailed status. If the load is successful, select All (instead of Errors) in the drop-down on the Status screen to see details about the import.

---

**Exporting Metadata**

You can export metadata to a flat file in a .csv (comma-delimited) or .txt (tab-delimited or other delimiter character) format.

These artifacts are supported in the exports:

• Dimensions
• Smart Lists
• Exchange rates

The application creates an export file for each artifact (.txt or .csv, depending on the file type), and all export files are consolidated into one zip file. You must extract the .csv or .txt files from the zip file if you want to use the files as import files (for example, when importing into another application).

Watch this tutorial video to learn how to export metadata.
To export metadata to a flat file:

1. Click **Application**, and then click **Overview**.
2. Click **Dimensions**, and then click **Export**.
3. On the **Export Metadata** page, click **Create**.
4. Select the target environment of the export file:
   - **Local**—Saves the export file to a location on your local computer.
   - **Outbox**—Saves the export file to the server where you can use it to schedule the export job. See **Uploading and Downloading Files Using the Application Inbox and Outbox**.

   **Note:**
   You can run up to five export jobs at one time.

5. Select the artifact or artifacts to export.
6. Select an option:
   - **Comma delimited**—Creates a comma-delimited .csv file for each artifact.
   - **Tab delimited**—Creates a tab-delimited .txt file for each artifact.
   - **Other**—Creates a .txt file for each artifact. Enter the delimiter character that you want to use in the export file. For a list of supported delimiter characters and exceptions, see **Other Supported Delimiter Characters**.

7. Click **Export**, and then specify where to save the export file.

### Importing and Exporting Data

After the dimensions are imported, you can populate the application data. You can't do a direct data load. You must import data from a file.

Watch this tutorial video to learn how to export and import data in the application.

### Importing Data

Watch this video about loading data into the application.
Overview Video

To import data:

1. Click Application, and then click Overview.
2. Click Actions, and then click Import Data.
3. Click Create.
4. Select the location of the data import file:
   - Local—Imports the data import file from a location on your computer.
   - Inbox—Imports the data import file from the server. See Uploading and Downloading Files Using the Application Inbox and Outbox.

   **Note:**
   
   Valid data import file types are .csv, .txt, or .zip.

5. Select the **Source Type** of the data import file:
   - Planning—Select this source type to load data in a Planning data file format.
   If you select this option, you must choose between **Comma delimited**, **Tab delimited**, or **Other**.

   **Note:**
   
   - If an import error occurs during a Planning data load, the import will continue to load valid data (partial load). Any errors will be noted in the import and export status. See Viewing Data Import and Export Status.
   - For a list of supported delimiter characters and exceptions, see **Other Supported Delimiter Characters**.
   - If data is exported to a file in Planning format, you can use Note-pad to change the database name in the file; for example, ASO-Cube, and then import the data file into the aggregate storage outline (provided all dimensions exist in the aggregate storage outline).

   - Essbase—Select this source type to load data in an Essbase data file format.
   If you select this option, you must choose a cube.

   **Note:**
   
   - If any import errors occur during an Essbase data load, the import will quit and no data will be loaded.

6. Enter or select the source file:
• If you selected Local, click Browse to navigate to the source file.
• If you selected Inbox, enter the name of the file in Source File.

7. **Optional**: Clear Include Metadata to improve data import performance and to prevent members from changing position in the outline during data import. Regardless of this option setting, the required metadata must already exist in the Planning application and in the Essbase outline before you can load the application data.

8. Click Save as Job to save the import operation as a job, which you can schedule to run later.

   **Note:**
   
   • Save as Job is available only when Outbox is selected.
   • Saving an import operation as a job is useful to batch a load sequence, for example, import metadata, then import data, then run rules when the data load is complete. See Managing Jobs.
   • You can run up to five import jobs at one time.

9. **Optional**: Click Validate to test whether the import file format is correct.

10. Click Import to run the import operation.

   **Note:**
   
   There is no option to clear data before import. Therefore, the import will not overwrite existing data values with #MISSING values.

   **Caution:**
   
   If the evaluation order in the application is changed after exporting data, the import may have unexpected results.

If a value in a data load input file doesn't match the evaluated driver member type; for example, if the evaluated type of a driver member is “date” and it’s corresponding input value isn't a valid date format, a driver member casting error occurs. See Driver Member Casting Errors.

**Driver Member Casting Errors**

A driver member casting error occurs when a value in a data load input file doesn't match the evaluated driver member type. Here are some examples of driver member casting errors:

• If a driver member is of type "Date" and it's corresponding input value isn't a valid date format
• If a nonexistent Smart List value is specified for a driver member of type "Smart List"
• If a nonnumeric value is supplied for a member of type "Currency," "Data," or "Percent."

A casting error can occur for each driver member specified in the input file and for every occurrence in a data record, so the number of casting errors for a given input record ranges from 0 to the number of drivers. Because successful driver member bindings can occur along with driver member binding errors, the application will not reject the entire input record, but the error will be noted in the import and export status. See Viewing Data Import and Export Status.

Consider the following data record input file where a bad date value is specified:

Entity, aDate, aSl, aText, aCurr, aPercent, Point-of-View, Data Load Cube Name

e1, exampleBadDateValue, sl1, text11, 888, 0.99, "FY15,Current,""BUVersion_1"", Local, ul, Jan", Plan1

Error messages displayed:
• After loading the data, the status window will read:

Planning Outline data store load process finished (Invalid data values were not loaded). 7 data records were read, 7 data records were processed, 7 were accepted for loading (verify actual load with Essbase log files), 0 were rejected.

• The error is noted as an error in the import and export status with the following entry details:

Category: Data record

Record Index: 2

Message: Unable to load specified data value in Record 2: (aDate: exampleBadDateValue)

• If multiple casting errors occur, each error is recorded in the same error message:

Unable to load specified data values in Record 2: (aSl: aaaaaasll), (aPercent: cc), (aDate: exampleBadDateValue), (aCurr: bb)

• If the error message count exceeds the maximum limit, it will issue the following message:

The warning [Unable to load specified data values] message count limit (100) exceeded: no more warnings of this type will be issued.
Exporting Data

To export data to a file:

1. Click **Application**, and then click **Overview**.
2. Click **Actions**, and then click **Export Data**.
3. Click **Create**.
4. On the **Export Data** page, select the target environment of the data export file:
   - **Local**—Saves the data export file to a location on your local computer.
   - **Outbox**—Saves the data export file to the server. See Uploading and Downloading Files Using the Application Inbox and Outbox.
5. Select the cube.
6. Select the file type:
   - **Comma delimited**—Creates a comma-delimited .csv file for each artifact.
   - **Tab delimited**—Creates a tab-delimited .txt file for each artifact.
   - **Other**—Creates a .txt file for each artifact. Enter the delimiter character that you want to use in the export file. For a list of supported delimiter characters and exceptions, see Other Supported Delimiter Characters.
7. For **Smart Lists**, specify **Export Labels** or **Export Names**.
8. For **Dynamic Members**, select whether to **Include** or **Exclude** dynamic calculation members during export.

**Note:**

Excluding dynamic members from exported data improves performance during data exports.

9. Select the slice of data to be exported.

**Note:**

If you're exporting data from an aggregate storage cube and you're exporting multiple members simultaneously, make sure that at least one of the members you select is a non-shared member. You can export shared members individually because it's equivalent to selecting a base member. However, selecting level 0 descendants of a parent member whose descendants are all shared members is equivalent to not selecting any members from a dimension and the export will not run. Therefore, you must ensure that one of the members selected is a non-shared member.

10. **Optional:** Click **Save as Job** to save the export operation as a job, which you can schedule to run immediately or at a later time.
11. Click Export, and then specify where to save the data export file.

Note:

To reduce the size of data export files, if a form has an entire row of #MISSING values, the row will be omitted from the data export file.

Viewing Data Import and Export Status

To view the status of a data import and export:

1. Click Application, and then click Jobs.
2. Click the name of the import or export job under Recent Activity.

Importing Exchange Rates for a Simplified Multicurrency Application

Import exchange rates for a simplified multiple currency application using a template.

To simplify managing exchange rates for a simplified multiple currency application, you can export the provided template, enter exchange rates in a spreadsheet, and import the rates to your application.

To import exchange rates using a template:

1. From the Home page, click Application, and then Overview.
2. From the Actions menu, select Export Exchange Rate Template, and then open ExchangeRateTemplate.csv or save it locally.
3. In the template file, enter average and ending currency exchange rates for each period and each currency, specify members for the POV, and then save the file.

Note:

The number 1 represents the exchange rate for the default currency in the application. Enter rates relative to the default currency.

4. From the Actions menu, select Import Exchange Rates, click Create, browse for the file, and then click Import.
Creating and Refreshing Application Databases

Things you need to know before you refresh the application database.

Related Topics
• Creating Application Databases
• Before Refreshing the Database
• Refreshing Application Databases

Creating Application Databases

The application database, which stores data for each cube in the application, is created when the application is created. See Creating an Application.

If you need to clear the data, you can do so using Calculation Manager. To refresh the data, see Refreshing Application Databases.

Before Refreshing the Database

You must refresh the database after changing the application structure, and modifying artifacts such as dimensions and members. Oracle recommends that you notify all users, asking them to save their work and close the application, before you refresh the database.

Before refreshing the database:
• Back up your outline file and export data from all databases.
• Decide if you want to enable all users or just the current administrator to use the application in administration mode during the refresh
• Decide if you want to log off all users
• Terminate any active application requests

Refreshing Application Databases

To refresh the database:

1. See Before Refreshing the Database.
2. Click Application, and then click Overview.
3. Click Actions, and then Refresh Database.
4. On the Refresh Database page, click Create, and then select from the following options:
   Before Refresh Database
   • Enable use of the application for—Lets All users or Administrators (or the current logged in administrator) access the application in administration mode during the refresh
   • Log off all users—Logs off all application users before starting the refresh
Adding an Aggregate Storage Outline to an Application

Understand what aggregate storage is and how to use it in an application.

Related Topics
• About Aggregate Storage
• Aggregate Storage Outline Cube Characteristics
• Process for Adding an Aggregate Storage Database to an Application

About Aggregate Storage

Aggregate storage is the database storage model that supports large-scale, sparsely distributed data that is categorized into many, potentially large dimensions. Selected data values are aggregated and stored, typically with improvements in aggregation time. Aggregate storage is an alternative to block storage (dense-sparse configuration).

Aggregate Storage Outline Cube Characteristics

• The application doesn't generate XREFs on aggregate storage databases. XREFs can only be generated on block storage databases.
• Because the application doesn't require all base dimensions on an aggregate storage database, approvals may not apply to the aggregate storage database if an approvals dimension is missing. If this is the case, normal security would apply.
• Dynamic time series members are not applicable for the Period dimension in an aggregate storage application.
• Creating and refreshing security filters are not applicable for aggregate storage databases.

Process for Adding an Aggregate Storage Database to an Application

To add an aggregate storage database to an application:

1. Create an aggregate storage cube. Perform one task:
   • Create an aggregate storage cube during application creation.
     See Creating an Application.
   • Add a new cube using the cube editor.
     See Managing Cubes.
2. Add dimensions to the aggregate storage cube.

Stop all active requests—Terminates any active requests in the Essbase application before starting the refresh

After Refresh Database

Enable use of the application for—Lets All users or Administrators use the application after the refresh

5. To refresh the database now, click Refresh Database. To run the refresh later, as a job, click Save as Job. See Managing Jobs.
See Editing Dimensions in the Simplified Dimension Editor.

- If a Currency, Years, Scenario, or Version dimension is valid for an aggregate storage cube, the dimension members are also valid for an aggregate storage cube.

- To perform currency calculations in a simplified multiple currency application for a custom aggregate storage cube, you must enable the Account, Period, and Currency dimensions for the aggregate storage cube. These dimensions in default aggregate storage cubes are enabled by default, but you must enable them manually for custom aggregate storage cubes. To enable Account, Period, and Currency dimensions for custom aggregate storage cubes, edit dimension properties for each dimension and select to enable them for the aggregate storage cube. See Editing Dimension Properties in the Simplified Dimension Editor.

3. Add dimension members. See Adding Members in the Simplified Dimension Editor.
4. Refresh the outline for the application. See Refreshing Application Databases.
5. Create a form using the dimensions associated with the aggregate storage database. See Administering Forms

Converting to a Standard Application

To increase the number of cubes and use the logic required to support more extensive calculations, convert a Reporting application to a Standard application.

During conversion, one cube, and the associated dimensions are added to your application. After conversion, you can add up to six additional cubes (three aggregate storage outline cubes and three block storage outline cubes).

You can enable sandboxes while converting a Reporting application to a Standard application. However, after you convert to a Standard application, you can't later enable it for sandboxes. See Enabling Sandboxes.

⚠️ Caution: Converting an application can't be undone. Be sure you back up your data before converting.

To convert to a Standard application:

1. Click Application, and then click Overview.
2. Click Actions, and then select Convert to Standard.

Converting a Standard or Reporting Application to an Enterprise Application

Learn about converting an application in this topic.
• If you purchased EPM Cloud after June 4, 2019, see New EPM Cloud Services in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators* to learn about migrating an application to the new services.

• If you purchased EPM Cloud before June 4, 2019, you can convert a Standard or Reporting application to an Enterprise application.

To convert a Standard or Reporting application:

1. Review the conversion considerations. See Conversion Considerations.
2. Back up the Standard or Reporting application.

3. Open the application, and then click Application, and then Overview.

4. Click Actions, then Convert to Enterprise, and then Continue to proceed. A message lets you know if the application is converted to Enterprise. If successful, log off.

   The application type is changed to "Enterprise", which includes options to enable and configure, and enables Groovy scripting. Metadata and data isn't changed during the conversion.

5. Log on and enable features that you need.

   These validations are performed when you enable features:
   - There can't be a member that already exists with the same name as any of the custom and base dimensions you specify in Map/Rename Dimensions. For example, if you create a custom dimension called Market in Financials, there can't be a member called Market in any other dimension.
   - There can't be a member with the name NO_<<customDimension>>, Total_<<Dimension>>, All_<<Dimension>>, where <<Dimension>> is the actual dimension name you specified. For example, if you rename Plan Element in Financials to Element for a custom dimension, there can't be No Element, Total Element. This is specific to modules that depend on any non-prefixed member.
   - The data storage for the Period and Account dimensions must be set to Never Share.
   - The Year dimension name must be Years.
   - 18 alias tables are predefined. The maximum supported is 32 (1 used internally). The conversion process attempts to merge the predefined alias tables with the alias tables in the existing application; if the count goes above 30, you get a validation error.
   - The conversion process calculates the number of existing dimensions in the application and the number of new dimensions that would be created if the module is initialized. If resulting sum of both is greater than 32 (the maximum allowed), you'll see a validation error.
   - For Projects, the name you specify for Program (a custom dimension option) shouldn't already exist as a non-attribute dimension.
Conversion Considerations

You can convert a Standard or Reporting application to an Enterprise application if:

- You are commercially authorized to implement an application of type "Enterprise".
- It does not use weekly, quarterly, or custom time periods.
- You rename the Year dimension to Years prior to converting.
- It uses simplified multicurrency (that is, it's not a standard currency application), if it's a multicurrency application.
- Artifact names don't use these prefixes:
  - oep_—Prefix for common artifacts
  - ofs_—Prefix for Financials artifacts
  - owp_—Prefix for Workforce artifacts
  - opf_—Prefix for Projects artifacts
  - ocx_—Prefix for Capital artifacts

Removing an Application

Proceed with caution when removing an application.

Because removing an application will delete all of its contents and any scheduled jobs, make a backup of your application.

To remove an application:

1. Click the **Application**, then click **Overview**.
2. Click **Actions**, and then **Remove Application**.

Setting the Daily Maintenance Process Time

Understand how to determine and set the most convenient time to start the hour-long daily maintenance process.

A business process instance requires one hour every day to perform routine maintenance. Service Administrators can select (and change) the most convenient time to start the hour-long daily maintenance process. In addition to routine maintenance, Oracle applies required patches to the business process instance during the maintenance process.

Because the business process instance isn't available to users during the maintenance process, the Service Administrator should identify a one-hour period when no one uses the business process. Any connected user will be logged off and will lose unsaved data.
The default maintenance start time is midnight Pacific Standard Time (PST). If you don't reset the default start time for your instance, Oracle resets it to a random time between 10:00 p.m. and 2:00 a.m. PST to reduce the business process maintenance duration.

**Note:**

If you really want the maintenance time to start at midnight PST, you should select it using this procedure. In such cases, Oracle will not change your selection.

To manage the automated maintenance window:

1. Access a business process instance.
2. Tap or click **Tools** and then **Daily Maintenance**.
3. In **Start Time**, select the time at which the daily maintenance process should start. Also, select your time zone from the drop-down list.
4. Click **Apply**.

For high volume data customers, the hour-long daily maintenance process may not be long enough to complete a full data export.

The **Export Planning Smart List textual data during daily maintenance for incremental data import** application setting, when enabled, exports the Essbase data plus the Planning member-driven Smart List intersections with their corresponding text labels during the daily maintenance process. This export can be lengthy because the system has to find all the possible intersections, uniquely identify their mappings, and then extract them.

The advantage to exporting the data for Planning member-driven Smart Lists in the daily maintenance process is that you can use the data later for an incremental (or selective) data load. However, the Planning member-driven Smart List data is not needed to restore your system.

To improve the maintenance process time for those customers who are experiencing lengthy data exports, you can disable the Planning member-driven Smart List data export during the maintenance process, thus only backing up the data needed to perform a full restoration.

Application setting: **Export Planning Smart List textual data during daily maintenance for incremental data import**

- **Yes**—Performs a complete export during the daily maintenance process, such that data, including the Planning Smart List data, can be incrementally imported to an application (this option may lengthen the maintenance process duration)
- **No** (default)—Creates an application backup during the maintenance process, such that data can be used as part of a full restoration

To view and change this setting, see **What Application and System Settings Can I Specify?**
Caution:

- If No is selected, the data portion of the backup does not have Planning Smart List data, so using the backup to perform an incremental (or selective) data load can lead to data loss.
- Importing only the Essbase portion of the data from a full maintenance process snapshot is problematic.

If you routinely use the maintenance process export for incremental data imports and are comfortable with the length of the current daily maintenance process, we recommend that you set the option to Yes. Otherwise, keep the default option (No) and perform a separate data export using the Migration user interface. Data exported during a migration can be used to perform an incremental data import, but keep in mind that since this export includes the Smart List data, the export duration can be lengthy depending on the data volume.

### Uploading and Downloading Files Using the Application Inbox and Outbox

Use the Inbox/Outbox Explorer to upload files to the server or download them to your computer. You can also use it to schedule import and export jobs.

To schedule an import or export job, see Scheduling Jobs.

To upload and download files using the Inbox/Outbox Explorer:

1. Click Application, and then click Overview.
2. Click Actions, and then Inbox/Outbox Explorer.
3. Filter by name, refresh the listing, and upload files from the server.
Using the Member Selector

Related Topics

- Working with Members
- Making Selections
- Member Relationships
- Using Wildcards in Searches
- Selecting Attribute Values as Members
- Selecting Members for Forms
- Selecting Substitution Variables as Members
- Selecting User Variables as Members

Working with Members

Use the Select Members dialog box to select the members to use with the following features:

- Business rule runtime prompts
- User variables
- Dynamic variables
- Data export
- Ad hoc forms
- Point of view and page axis where valid intersection rules are applied
- Valid intersections

If variables and attributes are defined, you can select variables and attributes. You can display and select members by member name or alias. The display options that you define for the Member Selection dialog box override those defined as an application default by an administrator, and those specified as an application preference.

Making Selections

Only members, substitution variables, and attributes to which you have access are displayed. The Selections pane only displays if you invoke the member selector for multiple member selection.

If valid intersection rules are used, only valid members are displayed in application forms and in runtime prompts for Calculation Manager rules. In ad hoc forms, both on the Web and in Oracle Smart View for Office, the application doesn't filter by valid intersections rules in point of view and page axis. Only cells in grids honor valid intersection rules in ad hoc forms.
To make selections:

1. Click .

2. **Optional:** Perform these tasks:
   - To enter search criteria (member name or alias only), press **Enter** (from the desktop) or click **Search** (on mobile).

   ![Note]
   The search isn't case-sensitive. You can search for a word, multiple words, or wildcard characters. See [Using Wildcards in Searches](#).

   - To change display options such as viewing variables and attributes, showing alias names, showing member counts, sorting alphabetically, refreshing the member list, or clearing selections, click **next to Search**, and then select from the list of display options.

   - To filter the members that are displayed in the member list, under **next to Search**, select **Add Filter**, and then select from the list of filter options.

   ![Note]
   Android users can click **to view and select filter options.**

3. Make selections by clicking a member in the member list.
   To understand how related members are selected, see [Member Relationships](#).
4. Optional: Perform these tasks:
   • To further refine which related members are selected in the Selections pane, click \( \Rightarrow \) to the right of the member.
   • If substitution variables or attributes are defined, below the member selection area, click \( \downarrow \) next to Members, and then select Substitution Variables or Attributes to select members for substitution variables or attributes. Members are displayed as children. Only members to which the user has read access are displayed in forms.
   • To move or remove members in the Selections pane, click \( \Rightarrow \) next to Selections.
   • To highlight the location of a selected member within the dimension hierarchy, double-click the member name in the Selections pane.

5. When you're done making selections, click OK.
Member Relationships

This table describes which members and related members are included during member selection.

Table 5-1   Member Relationships

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Members Included</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member</td>
<td>The selected member</td>
</tr>
<tr>
<td>Ancestors</td>
<td>All members above the selected member, excluding the selected member</td>
</tr>
<tr>
<td>Ancestors (inc)</td>
<td>The selected member and its ancestors</td>
</tr>
<tr>
<td>Children</td>
<td>All members in the level immediately below the selected member</td>
</tr>
<tr>
<td>Children (inc)</td>
<td>The selected member and its children</td>
</tr>
<tr>
<td>Descendants</td>
<td>All descendants of the selected member, excluding the selected member</td>
</tr>
<tr>
<td>Descendants (inc)</td>
<td>The selected member and its descendants</td>
</tr>
<tr>
<td>Siblings</td>
<td>All members from the same level in the hierarchy as the selected member, excluding the selected member</td>
</tr>
<tr>
<td>Siblings (inc)</td>
<td>The selected member and its siblings</td>
</tr>
<tr>
<td>Parents</td>
<td>The member in the level above the selected member</td>
</tr>
<tr>
<td>Parents (inc)</td>
<td>The selected member and its parent</td>
</tr>
<tr>
<td>Level 0 Descendants</td>
<td>All descendants of the selected member that have no children</td>
</tr>
<tr>
<td>Left Siblings</td>
<td>The members that appear before the selected member with the same parent</td>
</tr>
<tr>
<td>Left Siblings (inc)</td>
<td>The selected member and its left siblings</td>
</tr>
<tr>
<td>Right Siblings</td>
<td>The members that appear after the selected member with the same parent</td>
</tr>
<tr>
<td>Right Siblings (inc)</td>
<td>The selected member and its right siblings</td>
</tr>
<tr>
<td>Previous Sibling</td>
<td>The member that appears immediately before the selected member with the same parent</td>
</tr>
<tr>
<td>Next Sibling</td>
<td>The member that appears immediately after the selected member with the same parent</td>
</tr>
<tr>
<td>Previous Generation Member</td>
<td>The member that appears immediately before the selected member within the same generation</td>
</tr>
<tr>
<td>Next Generation Member</td>
<td>The member that appears immediately after the selected member within the same generation</td>
</tr>
</tbody>
</table>

Using Wildcards in Searches

You can use these wildcard characters to search for members.
Table 5-2  Wildcard Characters

<table>
<thead>
<tr>
<th>Wildcard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>Match any single character</td>
</tr>
</tbody>
</table>
| *        | Match zero or multiple characters. For example, enter "sale*" to find "Sales" and "Sale" because * includes zero or more characters after the word "sale."
|           | The default search uses the * wildcard. For example, entering "cash" searches for "*cash*" and returns "Restricted Cash", "Cash Equivalents", "Cash", and "Noncash Expenses" because the word "cash" appears within each matched item. |
| #        | Match any single number (0-9) |
| [list]   | Match any single character within a specified list of characters. You can list specific characters to use as wildcard. For example, enter [plan] to use all the letters within the brackets as a single wildcard character. You can use the "." character to specify a range, such as [A-Z] or [!0-9]. To use the "." character as part of the list, enter it at the beginning of the list. For example, [!@&] uses the characters within the brackets as wildcard characters. |
| [!list]  | Match any single character not found within a specified list of characters. The "." character can also be used to indicate a range, such as [!A-Z] or [!0-9]. |

Selecting Attribute Values as Members

If attribute members are defined, you can select attribute values on the Select Members dialog box. For attribute members, selecting a non-level 0 attribute selects all level 0 descendants and applies the operator to each. For attributes of type numeric, date, and Boolean (where false = 0 and true = 1), evaluation is based on the minimum and maximum values. For text attributes, evaluation is based on the position from top to bottom in the hierarchy. The top position has the lowest value, and the bottom position has the highest value.

Example: Numeric attribute

In this example, the selected operator is applied to each level 0 descendant, based on the numeric value. For example, selecting NotEqual and Small in the Member Selection dialog box includes all values not equal to 1 and not equal to 2, so the selection includes 3, 4, 5, and 6. Selecting Greater and Small includes all values greater than 1 or greater than 2, so the selection includes 2, 3, 4, 5, and 6.

Size
Small
1
2
Table 5-3  Example: Numeric Attribute Evaluation

<table>
<thead>
<tr>
<th>Selected Operator</th>
<th>Selected Attribute Value</th>
<th>Result</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal</td>
<td>Large</td>
<td>5, 6</td>
<td>The Equal operator is applied to all level 0 descendants of Large, which includes 5 and 6.</td>
</tr>
<tr>
<td>Less</td>
<td>Medium</td>
<td>1, 2, 3</td>
<td>The Less operator is applied to all level 0 descendants of Medium. This includes values &lt; 3 OR &lt; 4, which results in 1, 2, and 3.</td>
</tr>
<tr>
<td>Greater</td>
<td>Medium</td>
<td>4, 5, 6</td>
<td>The Greater operator is applied to all level 0 descendants of Medium. This includes values &gt; 3 OR &gt; 4, which results in 4, 5, and 6.</td>
</tr>
<tr>
<td>GreaterOrEqual</td>
<td>Medium</td>
<td>3, 4, 5, 6</td>
<td>The GreaterOrEqual operator is applied to all level 0 descendants of Medium. This includes values &gt;=3 OR &gt;= 4, which results in 3, 4, 5, and 6.</td>
</tr>
<tr>
<td>LessOrEqual</td>
<td>Medium</td>
<td>1, 2, 3, 4</td>
<td>The LessOrEqual operator is applied to all level 0 descendants of Medium. This includes values &lt;=3 OR &lt;=4, which results in 1, 2, 3, and 4.</td>
</tr>
<tr>
<td>NotEqual</td>
<td>Medium</td>
<td>1, 2, 5, 6</td>
<td>The NotEqual operator is applied to all level 0 descendants of Medium. This includes values not equal to 3 AND not equal to 4, which results in 1, 2, 5, and 6.</td>
</tr>
</tbody>
</table>

Example: Text attribute

For text attributes, the selected operator is applied to each level 0 descendant based on its position in the hierarchy, from top (lowest value) to bottom (highest value).

In this example, Envelope is at the top position and has the lowest value. Packet has the next higher value, followed by Box, Carton, Barrel and Crate. Crate is at the bottom position and has the highest value.
For this text attribute, selecting Less and Small includes values that are less than Envelope or less than Packet. Because Envelope is less than Packet, the resulting selection includes only Envelope. Likewise, selecting Greater and Large includes values that are greater than Barrel or greater than Crate, so the resulting selection includes only Crate.

Containers
Small
Envelope
Packet
Medium
Box
Carton
Large
Barrel
Crate

Table 5-4 Example: Text Attribute Evaluation

<table>
<thead>
<tr>
<th>Selected Operator</th>
<th>Selected Attribute Value</th>
<th>Result</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal</td>
<td>Medium</td>
<td>Box, Carton</td>
<td>The Equal operator is applied to all level 0 descendents of Medium, which includes Box and Carton.</td>
</tr>
<tr>
<td>NotEqual</td>
<td>Medium</td>
<td>Envelope, Packet, Barrel, Crate</td>
<td>The NotEqual operator is applied to all level 0 descendents of Medium. This includes values not equal to Box AND not equal to Carton, which results in Envelope, Packet, Barrel, and Crate.</td>
</tr>
<tr>
<td>Less</td>
<td>Medium</td>
<td>Box, Packet, Envelope</td>
<td>The Less operator is applied to all level 0 descendents of Medium. This includes everything at a lower position than Carton OR a lower position than Box, which results in Box, Packet, and Envelope.</td>
</tr>
<tr>
<td>LessOrEqual</td>
<td>Medium</td>
<td>Envelope, Packet, Box, Carton</td>
<td>The LessOrEqual operator is applied to all level 0 descendents of Medium. This includes everything at the same position as Carton OR at a lower position than Carton, which results in Envelope, Packet, Box, and Carton.</td>
</tr>
</tbody>
</table>
Selecting Members for Forms

When selecting members for forms:

- To filter members from certain users, restrict their access permissions to members, and then refresh the plan.

- The order of members in the **Selected Members** list determines the order on forms. To change the order, select a member and click the Up or Down Arrow above the selected members list.

**Note:**

If you select members individually and select their parent first, the parent displays in the form at the top of its member hierarchy. (Note that depending on the number of hierarchy levels, calculating totals for the parent of individually selected members could take several passes, slowing calculations). The parent of members selected by relationship, for example, by I(Descendants), displays at the bottom of the hierarchy.

- In the **Layout** tab of the **Form Management** dialog box, you can open the **Member Selection** dialog box by clicking the member selection icon, or by right-clicking a row or column and selecting **Select Members**.

- To select different sets of members across the same dimension, see **Creating Asymmetric Rows and Columns**.

- For forms with multiple dimensions in a row or column, you can set member selection options for a dimension by selecting that dimension from the **Dimensions** drop-down list that is displayed in the Member Selection dialog box for multiple dimensions in a row or column.

- If you click the member selection icon, an option is displayed for **Place Selection in Separate Rows** or **Place Selection in Separate Columns**. This adds the selection to the rows or columns after the last existing row or column on the form. For example, for a form that contains members Acct1, Acct2, and Acct3 in Column A, if you select these members with **Place Selection in Separate Columns** selected, Acct1 is selected for column A, Acct2 for column B, and Acct3 for column C. If you select the members without this option, all of the members are selected for column A.

This feature is available for single members only, not for members selected with functions, such as Children (inc). For example, if you select Q/IChildren for Column A and select **Place Selection in Separate Columns**, the form layout isn't changed.

- All settings except Count are retained after the Member Selection dialog box is closed, and members in the Member Selection dialog box are displayed based on user-defined settings. Members displayed on the Layout tab don't inherit the display settings defined in Member Selection dialog box. Instead, they are displayed using the member name.

- To define different sets of members for a dimension, see **Creating Asymmetric Rows and Columns**.
To set display, functionality, and printing options, see Setting Form Precision and Other Options.

Selecting Substitution Variables as Members

Substitution variables act as global placeholders for information that changes regularly. Substitution variables are especially useful for developing and reporting on rolling forecasts. When you select substitution variables as members on the form, their values are based on dynamically generated information. For example, you could set the current month member to the substitution variable CurMnth so that when the month changes, you need not update the month value manually in the form or the report script.

Note:

- When you open or calculate values on forms, the application replaces substitution variables with values assigned to them.
- You create and assign values to substitution variables within the application. These substitution variables are then available in the application when you select members for a form. For instructions on creating and assigning values to substitution variables using the application, see Working with Substitution Variables.
- Substitution variables must be appropriate for the context in forms. For example, you could select a substitution variable named CurrQtr with a value of Qtr2 as a member of the Time Period dimension. It's not valid to select a substitution variable named CurrYr for the Years dimension if its value is Feb. You can set substitution variables at the application or database level.

The same substitution variable can exist on multiple levels; the application uses the first one it finds as it searches in this order:

1. Database
2. Application

- You can select from substitution variables if they are enabled for runtime prompts in business rules, and their values match a member set in the runtime prompt for a business rule.
- The application checks the validity of substitution variables when they are used (for example, when the form is opened). It doesn't check when you design forms, so you should test substitution variables by saving and opening forms.
- If you migrated an application that uses Global variables, you can view, but not edit, them in Planning.

To specify substitution variables in forms:

1. Create the form (see Creating Simple Forms).
2. In Member Selection, select substitution variables the same way you select members, to move substitution variables to and from Selected Members.
When selected, a substitution variable is preceded by an ampersand (&). For example:

&CurrentScenario

3. Click OK.

Selecting User Variables as Members

User variables act as filters in forms, enabling planners to focus only on certain members, such as a department. Before you can associate a user variable with a form, you must create the user variable. See Managing User Variables.

When you create forms with user variables, planners must select values for the variable before opening forms. For example, if you create a user variable called Division, planners must select a division before working in the form. The first time you select a variable for a form, you do it in preferences. Afterward, you can update the variable in preferences or in the form.

To select user variables for forms:

1. Create the form (see Creating Simple Forms).

2. On Member Selection, select user variables the same way you select members, using the arrows to move substitution variables to and from Selected Members.

   User variables are displayed for the current dimension. For example, user variables for the Entity dimension might display as follows:

   Division = [User Variable]

   When selected, a user variable is preceded by an ampersand. For example:

   Idescendants(&Division)

3. Click OK.
Connecting Subscriptions in EPM Cloud

Related Topics
• About Connecting EPM Cloud Subscriptions
• Considerations When Migrating Cross-Subscription Connections
• Creating, Editing, and Deleting Connections to Other EPM Cloud Subscriptions
• Connecting to External Web Services
• Specifying Advanced Options for External Connections
• Common Connection Errors and Resolutions
• Navigating Across EPM Cloud Subscriptions
• Customizing Navigation Flows to Access Other EPM Cloud Subscriptions

About Connecting EPM Cloud Subscriptions

Overview
Service Administrators can connect multiple EPM Cloud subscriptions of the following types:
• Planning
• Oracle Financial Consolidation and Close Cloud
• Oracle Tax Reporting Cloud
• Oracle Profitability and Cost Management Cloud
• Oracle Account Reconciliation Cloud
• Narrative Reporting
• Oracle Strategic Workforce Planning Cloud

Once administrators set up the connections, users who have access across EPM Cloud subscriptions can navigate across them from a single access point with one login. Also artifacts such as forms, dashboards, and so on, from across subscriptions can be co-mingled within a cluster or within tabs on a card in navigation flows. Artifacts in the target subscription are accessible based on the user's role.

Watch this overview video to learn how to integrate EPM business process flows.

Watch this overview video to learn how to customize your EPM Cloud workflow.
Which EPM Cloud subscriptions can I connect?

The source subscription is the subscription from which you’re creating the connection. The target subscription is the subscription to which you’re connecting from the source subscription.

You can connect these source subscriptions (these subscriptions can also be target subscriptions):

• Planning
• Oracle Financial Consolidation and Close Cloud
• Oracle Tax Reporting Cloud

Source subscriptions can also connect to these target subscriptions (these subscriptions can't be source subscriptions):

• Oracle Account Reconciliation Cloud
• Oracle Profitability and Cost Management Cloud
• Narrative Reporting
• Strategic Workforce Planning

What are the ways I can connect to other EPM Cloud subscriptions?

• Toggle between the source subscription and the target subscription on the Naviga
gator menu. See Navigating Across EPM Cloud Subscriptions.

• Customize navigation flows in the source subscription to access clusters, cards,
and artifacts in other target subscriptions from the Home page. See Customizing Navigation Flows to Access Other EPM Cloud Subscriptions.

Considerations

• Only Service Administrators create cross-subscription connections.

Users click a navigation link to open the linked subscription. Access within the linked subscription is determined by the predefined role and access permissions, if any, assigned to the user.

• For cross-subscription navigation to be seamless, all subscription instances to which cross-subscription navigation flows are setup must belong to the same identity domain.

Note:

If the target and source subscription instances are not on the same identity domain, then you'll not be able to establish a connection between them.

• Service Administrators cannot configure cross-subscription connections using corporate SSO (identity provider) credentials.
If your subscriptions are configured for SSO, ensure that identity domain credentials are maintained for the Service Administrators who configure cross-subscription connections. See Enabling Sign In With Identity Domain Credentials.

- Migrating cross-subscription connections between test and production environments can cause issues in certain use case scenarios. For more information, see Considerations When Migrating Cross-Subscription Connections.

Considerations When Migrating Cross-Subscription Connections

Oracle recognizes that it's common practice for administrators to try out new features, such as connecting subscriptions, on test environments and then migrate to production environments. However, in doing so, it could cause some issues after migration. We'll describe some use case scenarios that you need to be aware of.

In the following scenarios, assume you have subscriptions for Oracle Financial Consolidation and Close Cloud and Planning.

Use Case Scenario 1: Test to Production

In this scenario, the administrator has defined a connection between the Planning test instance and the Oracle Financial Consolidation and Close Cloud test instance. The administrator then uses this connection to build a navigation flow in the Planning test environment that refers to a card in the Oracle Financial Consolidation and Close Cloud test environment. If the administrator chooses to export all artifacts in the test environment, the export will include connections and navigation flows, including the connection to the Oracle Financial Consolidation and Close Cloud test instance.

When the administrator imports this snapshot into the Planning production environment, they will have an undesirable connection to the Oracle Financial Consolidation and Close Cloud test instance from the Planning production environment.

When migrating connections from test environments to production environments, it's important to ensure that any connections that were defined in the test environment are changed to point to the corresponding production instances.

Use Case Scenario 2: Production to Production or Test to Test

This scenario doesn't have any caveats.

Use Case Scenario 3: Production to Test

In this scenario, the administrator might be trying to replicate an issue on a test instance. In this case it's important that the administrator remember to change any connections that point to a production instance to a test instance. Connections that point to a production instance from a test environment could inadvertently tamper with production.

Creating, Editing, and Deleting Connections to Other EPM Cloud Subscriptions

Before you can create connections to other EPM Cloud subscriptions, you must ensure you have access to the source and target subscriptions you're connecting. You
must also have URLs for the other subscriptions you're connecting and login details for each subscription such as user ID (Service Administrator) and password.

To create, edit, duplicate, and delete connections:

1. Login to the source subscription.
2. From the Home page, click **Tools**, and then click **Connections**.
3. Choose an action:
   - To add a connection:
     a. On the **Manage Connections** page, click **Create**.
     b. Select the target subscription you want to add.
     c. Enter the target subscription connection details:
        - **In URL**, enter the URL of the target subscription instance; for example, http(s)://your-target-host-url.com. This is the URL that you normally use to sign in to the target subscription instance.
        - **In Connection Name** enter name for this navigation link; for example, Consolidation Application.
        - Enter an optional description for the link.
        - Use **Service Administrator** and **Password** to specify the credentials of a Service Administrator.

        **Note:**
        These credentials are used only to define and validate the navigation link. When a user logs in, their own role and access will be applied to access the target subscription.

        - The **Domain** field is automatically populated based on the URL you enter. If there is no domain in the URL, then the **Domain** field is left blank.
     d. Click **Validate**.
     e. If the validation is successful, click **Save and Close**.
   - To edit connections:
     a. On the **Manage Connections** page, click the name of a connection.
     b. Edit connection details.
     c. Click **Validate**.
     d. If the validation is successful, click **Save and Close**.
   - To duplicate a connection:
     a. On the **Manage Connections** page, in the **Action** column next to the connection you want to duplicate, click *******.
     b. Click **Duplicate**.
     c. Enter a name for the duplicate connection, then click **OK**.
   - To delete a connection:
a. On the **Manage Connections** page, in the **Action** column next to the connection you want to delete, click ✕

b. Click **Delete**.

When target subscriptions are connected to an EPM Cloud source subscription, they are listed in the **My Connections** pane on the **Navigator** menu of the source subscription. The **My Connections** pane on the **Navigator** menu is where you can navigate across subscriptions. See [Navigating Across EPM Cloud Subscriptions](#).

## Connecting to External Web Services

Service Administrators can also connect to external Web services for the purpose of reading data from and writing to an external Web service.

This connection can be referenced or used in a Groovy script to create a communications link between the Groovy script and the external HTTP/HTTPS resource. For more details and examples of how this connection can be used in a Groovy script, see the Java API documentation for the **Connection** and the **HttpRequest** objects in the EPM Groovy object model.

### Note:

The **Other Web Service Provider** connection type is only available for use with those business processes that allow Groovy Rules to be created. See [Oracle Enterprise Performance Management Cloud, Groovy Rules Java API Reference](#).

Before you can create connections to external Web services, you must ensure you have access to the Web service you’re connecting. You must also have URLs for the Web service and any login details, if required.

To create a connection to an external Web service:

1. Login to the source environment.
2. From the Home page, click **Tools**, and then click **Connections**.
3. On the **Manage Connections** page, click **Create**.
4. Click **Other Web Service Provider**.
5. Enter a **Connection Name** and a **Description** for the connection.
6. Enter the **URL** for the target connection.
7. Enter optional advanced options for the URL link.

### Note:

The optional advanced options enable you to specify query or header parameters when defining an external connection. See [Specifying Advanced Options for External Connections](#).
8. Enter User and Password login credentials for the connection, if required.

9. Click Save and Close.

Specifying Advanced Options for External Connections

The optional advanced options enable you to specify query or header parameters when defining an external connection.

**Note:**

The ability to define query parameters for an external connection is only available for use with those business processes that allow Groovy Rules to be created. See the Oracle Enterprise Performance Management Cloud, Groovy Rules Java API Reference.

To specify advanced options for external connections:

1. Create an external connection or open an existing external connection.
   See Connecting to External Web Services.

2. Enter connection details, and then click Show Advanced Options.

3. Specify query details as follows:

   • —Add query
   • —Delete query
   • **Type**—Select Header or Parameter.
     - **Header** sets a default header that will be sent on every request made for this connection. **Parameter** sets a default query parameter that will be sent on every request made for this connection.
   • **Secure**—If selected, the value entered in the Value field will be encrypted. Clearing the Secure check box for a row will remove the value.
     
     An example header that one would secure is the Bearer Token for external Web services that supports Bearer Authentication, or the API Key query parameter for external Web services that supports API keys for authentication.
   • **Name**—Enter a name for the header or query parameter.
   • **Value**—Enter the value for the header or query parameter.

Common Connection Errors and Resolutions

**Known Errors and Resolutions**

• The target connection does not belong to the same domain as the source. Ensure that the target connection is in the same domain as the source.
• The product version of the target is different from that of the source. Before creating a connection, ensure that both the source and target product versions are the same.

• The user name must have a Service Administrator role.

• The target connection must have an application. Create an application on the target and try again.

• The user name or password is invalid. Ensure the user is a Service Administrator.
  – Ensure the user password used in the connection definition is not expired.
  – Ensure the Domain field is not edited by the user. The domain is automatically populated from the URL. If there is no domain in the URL, then the Domain field is left blank.

• The URL provided is not for a business process of type <selected provider type name>.
  For example, the user has selected a provider type for Planning, but has provided a URL for Oracle Financial Consolidation and Close Cloud.

Unknown Errors and Resolutions

• The target you were trying to reach could not be found. Check the URL to ensure that the path is correct.
  – Ensure that the URL is valid.
  – Ensure that the URL is not a Workspace URL (for example, http://<hostname>/workspace).

• Unable to establish a connection due to an unexpected error. For common connection errors and possible resolutions, see "Connecting Subscriptions in EPM Cloud" in your Administering guide at Cloud Documentation.
  – The target server is down.
  – The target server is in maintenance mode.
  – The target server is performing daily maintenance and will not be available until the maintenance process is complete.
  – The target server URL is not a trusted website and is denied access.

Navigating Across EPM Cloud Subscriptions

Once an administrator creates connections to other EPM Cloud subscriptions, the connections are listed in the My Connections pane on the Navigator menu.
You can toggle between the subscriptions from this location. You must have access to the other subscriptions in order to open them. Artifacts are accessible based on the user's role.

To open another Oracle Enterprise Performance Management Cloud subscription:

1. From the Home page, click Navigator.

2. If subscriptions are connected and you have access to those subscriptions, you'll see a list of connected subscriptions in the My Connections pane. Click a subscription to open it.

Note:
Click the icon to the right of the subscription name to open the subscription in a new window.

Customizing Navigation Flows to Access Other EPM Cloud Subscriptions

You can customize the application interface to access other EPM Cloud subscriptions from the Home page of a source subscription. For example, you can add artifacts to the Home page, such as forms or financial reports, from other EPM Cloud subscriptions. You can group these artifacts (called cards) into clusters by customizing navigation flows. Clusters and cards from target EPM Cloud subscriptions can be directly included in the navigation flows of source EPM Cloud subscriptions. You can also use the Navigation Flow Designer to customize cards to have tabular pages where each tab is an artifact from a different subscription.

These two use cases describe in detail how to customize navigation flows to access other EPM Cloud subscriptions:

- Grouping Cards from Other EPM Cloud Subscriptions into Clusters
- Configuring Cards with Tabs from Multiple EPM Cloud Subscriptions
To learn more about designing navigation flows, see “Designing Custom Navigation Flows” in your Administering guide.

Grouping Cards from Other EPM Cloud Subscriptions into Clusters

You can group cards from various EPM Cloud subscriptions into a cluster that is accessible from the Home page of a source subscription. For example, you can create a cluster within Oracle Financial Consolidation and Close Cloud consisting of cards with pre-built external reports from Narrative Reporting.

Cards from multiple subscriptions can also be included within the same cluster on a source environment. For example, a Oracle Tax Reporting Cloud user can launch a Journals icon from Oracle Financial Consolidation and Close Cloud without leaving Oracle Tax Reporting Cloud.
You create clusters and add cards to clusters by customizing navigation flows. For general information about navigation flows, see "Designing Custom Navigation Flows" in your Administering guide.

To create a cluster made up of cards from other EPM Cloud subscriptions:

1. Launch the Navigation Flow page and create a navigation flow or edit an existing navigation flow:

   ![Navigation Flow Page](image)

   **Note:**

   To create a navigation flow, you must first select an existing navigation flow and make a copy of it. Then edit the duplicate navigation flow details and save them.

   a. Click **Tools**, and then click **Navigation Flows**.

   b. To create a navigation flow, select the navigation flow you want to duplicate, then in the upper right-hand corner of the page, click ⚙, and then select **Create Copy**. Enter a name for the navigation flow, and then click **OK**.

   **Note:**

   New flows are marked **Inactive** until they are activated by the administrator. To activate or deactivate a navigation flow, in the **Active** column, click **Active** or **Inactive**. Only one navigation flow at a time can be active.

   c. To edit an existing navigation flow, click the name of the navigation flow you want to edit.
2. Create a cluster or add an existing cluster:
   a. If it isn't already open, from the **Navigation Flow** page, click the name of the navigation flow in which you want to add a cluster.
   b. To create a new cluster, click **Add Cluster**, enter or select the cluster details, and then choose an icon for the cluster.
   c. If there is an existing cluster you want to add from another subscription, click **Add Existing Card/Cluster**, select the target subscription under **My Connections**, and then choose the cluster you want to add to your navigation flow.

   Note the following:
   - Clusters can't be directly selected from Narrative Reporting and Oracle Profitability and Cost Management Cloud using the **Add Existing Card/Cluster** option.
   - Clusters that are added from another navigation flow or from another subscription will display the localized labels that were defined in the source navigation flow. To update the cluster labels in your navigation flow, on the Home page, click **Tools**, and then click **Artifact Labels**. See “Specifying Artifact Labels” in your Administering guide.

3. Select the cards to include in the cluster:
   a. From the **Navigation Flow** page, navigate to the card you want to add to the cluster. If the card is within another subscription, first select the subscription under **My Connections**, and then navigate to the card in that subscription.
   b. To the right of the card that you want to move, in the **Order** column, click ▶.
   c. Select the cluster, and then click **OK**.

   The cards will appear in the listing as children of the cluster. Use the up and down arrows next to the cards to reorder the cards within the cluster, if needed.

4. Click **Save and Close**.

   You must activate the navigation flow and reload it to view your design time changes. To reload a navigation flow, click the down arrow next to your user name. Then on the **Setting and Actions** menu, click **Reload Navigation Flow**.
If you can’t see your referenced artifacts after activating and reloading the navigation flow, see Why Is My Referenced Card, Tab, or Cluster Not Visible?

Configuring Cards with Tabs from Multiple EPM Cloud Subscriptions

You can also customize cards in navigation flows to have tabular pages where each tab is an artifact from a different subscription. For example, a Planning user can click a Revenue icon which launches a card with horizontal tabs showing reports from Narrative Reporting.

You create tabular cards by customizing navigation flows. For general information about navigation flows, see “Designing Custom Navigation Flows” in your Administration guide.

To configure a card made up of tabs and sub-tabs from other EPM Cloud subscriptions:

1. Launch the Navigation Flow page and create a navigation flow or edit an existing navigation flow:
To create a navigation flow, you must first select an existing navigation flow and make a copy of it. Then edit the duplicate navigation flow details and save them.

a. Click Tools, and then click Navigation Flows.

b. To create a navigation flow, select the navigation flow you want to duplicate, then in the upper right-hand corner of the page, click , and then select Create Copy. Enter a name for the navigation flow, and then click OK.

Note:
New flows are marked Inactive until they are activated by the administrator. To activate or deactivate a navigation flow, in the Active column, click Active or Inactive. Only one navigation flow at a time can be active.

c. To edit an existing navigation flow, click the name of the navigation flow you want to edit.

2. Add a tabular card with artifacts from various target subscriptions:

a. If there is an existing card you want to add from another subscription, from the Navigation Flow page, click Add Existing Card/Cluster, select the target subscription under My Connections, and then choose the card you want to add to your navigation flow.

Note the following:

- Cards can’t be directly selected from Narrative Reporting and Oracle Profitability and Cost Management Cloud using the Add Existing Card/Cluster option.
- Cards that are added from another navigation flow or from another subscription will display the localized labels that were defined in the source navigation flow. To update the card labels in your navigation flow, on the Home page, click Tools, and then click Artifact Labels. See “Specifying Artifact Labels” in your Administering guide.

b. To add a new tabular card to the navigation flow, from the Navigation Flow page, click Add Card, and then select details for the card:

- **Name**—Enter a label for the card.
- **Visible**—Select whether the card is visible to users on the Home page.
- **Cluster**—If clusters exist, select a cluster for the card or select None.
- **Icon**—Select the icon that will be displayed for the card you’re creating. Choose from the available icons provided in the icon library.
- **Content**—Select from the following options:
  - **Page Type**—Select a multiple page (tabular) format.
3. Add tabs and sub tabs to the tabular card:
   a. To add an existing tab, click **Add Existing Tab**.
   b. To add a new tab, click **Add New Tab**, and then edit tab details.
   c. Click **Add New Sub Tab** or **Add Existing Sub Tab**, and then edit sub tab details.
   d. For **Artifact**, click to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include forms, dashboards, and financial reports. To select an artifact from another subscription, select the target subscription under **My Connections**, and then choose the artifact you want to add to your tab.
   e. Repeat adding tabs and sub tabs until the card is complete.

4. Click **Save and Close**.

**Note:**

- For cards with multiple tabs or sub tabs, the last tab accessed by a user will be retained the next time the user accesses the card in the same session. If the user logs out and then logs back in, the default tab will be displayed.
- Tabs or sub tabs that are added from another navigation flow or from another subscription will display the localized labels that were defined in the source navigation flow. To update the tab labels in your navigation flow, on the Home page, click **Tools**, and then click **Artifact Labels**. See “Specifying Artifact Labels” in your Administering guide.

You must reload the navigation flow to view your design time changes. To reload a navigation flow, click the down arrow next to your user name. Then on the **Setting and Actions** menu, click **Reload Navigation Flow**.

**Settings and Actions**

- **Reload Navigation Flow**
- Downloads...
- Help...
- **Cloud Customer Connect**
- Provide Feedback...
- Oracle Support...
- About...
- Sign Out
If you can't see your referenced artifacts after reloading the navigation flow, see Why Is My Referenced Card, Tab, or Cluster Not Visible?
Designing Custom Navigation Flows

Customize the application interface using navigation flows. Navigation flows enable designers to control how roles or groups interact with the application.

Related Topics
- Understanding Navigation Flows
- Viewing and Working with Navigation Flows

Understanding Navigation Flows

Navigation flows give application designers control over how various roles, or groups, interact with the application. The Navigation Flow Designer enables you to customize the application interface. For example, you can change the names of the cards and clusters that display on the Home page and the order in which they are displayed. You can hide cards, create new cards, and group cards into clusters. You can also customize the vertical and horizontal tabs that display on a card.

Watch these videos to better understand navigation flows.

- Overview Video
- Tutorial Video

Watch this overview video to learn how to customize your EPM Cloud workflow.

Related Topics:
- What Can Be Customized in the Application Interface?
- Navigation Flow Customization Categories
- Navigation Flow Permissions
- Predefined Navigation Flows
- Viewing and Working with Navigation Flows

What Can Be Customized in the Application Interface?

- Labels for cards or tabs
- Icons that are used for cards or tabs
- Hide and unhide cards and tabs
- Display order of cards and tabs
• Add new cards
• Add existing cards
• Add new horizontal or vertical tabs
• Remove navigation flows, cards, and tabs
• Group cards into clusters
• Add existing clusters


Navigation Flow Customization Categories

Navigation flows are categorized as follows for customization:

1. Global—Navigation flows are seen by all users
2. Role—Navigation flows are seen only by users in a specific role; for example, Planner or Interactive User
3. Group—Navigation flows are seen only by users belonging to a specific group

Navigation flows can be defined at any of these levels. In cases where navigation flows exist at multiple levels, updates are applied in the order of highest (global) to lowest (groups).

For example, if you create a navigation flow that displays an icon on the Home page named "My Tasks", and then another administrator duplicates the navigation flow, makes the following changes to the card, and then associates the navigation flow with a group:

• At the global level, they rename "My Tasks" to "Company Tasks"
• At the group level, for a group named Sales, they rename "My Tasks" to "Sales Tasks"

Users who belong to the group called Sales will see the label "Sales Tasks" in the navigation flow instead of "My Tasks," and all other users will see the label "Company Tasks".

Navigation Flow Permissions

The application offers three levels of permissions for navigation flows:

• Role-based—Permissions are granted to users or groups assigned to a specific role; for example, a Planner will see different cards displayed on the Home page than an Administrator
• Artifact-based—Permissions are granted to users or groups who can see certain artifacts; for example, a Planner will see only the forms to which they have been assigned permission
• Global—Permissions are granted to all users

Predefined Navigation Flows

The application comes with one predefined navigation flow, called Default. The Default navigation flow is read only; therefore, you can’t make any modifications to it.
These are the operations you can and cannot perform on the Default navigation flow:

- **Name**—You can’t modify the name.
- **Delete**—You can’t delete the navigation flow.
- **Edit**—You can view the navigation flow details, but you can’t change anything.
- **Activate or Deactivate**—You can activate or deactivate the navigation flow.
- **Duplicate**—You can make a copy of the navigation flow.

### Viewing and Working with Navigation Flows

Administrators can view a list of navigation flows, including the predefined navigation flow, within the Navigation Flow page.

The Navigation Flow page lists each navigation flow by name, indicates the role or the group that has access to the navigation flow (if assigned), and provides a description of the navigation flow (if provided). The listing also indicates whether the navigation flow is active or not.

⚠️ **Caution:**

There are naming restrictions for navigation flows, cards, clusters, tabs, and infolets in navigation flows. You cannot use these special characters:

- ampersand ( & )
- less than sign ( < )
- greater than sign ( > )
- quotation mark ( ” )
- backslash ( \ )
- plus sign ( + )

To view the navigation flow:

1. Click **Tools**, and then click **Navigation Flows**.
2. To work with a navigation flow, perform an action:
   - To create and duplicate navigation flows, see *Creating and Duplicating Navigation Flows*.
   - To edit a navigation flow, see *Editing a Navigation Flow*.
   - To activate or deactivate a navigation flow, see *Activating and Deactivating Navigation Flows*.
   - To rename cards and tabs, see *Customizing Labels for Cards, Tabs, and Clusters*.
   - To customize the graphics used for cards and tabs, see *Customizing Icons for Cards and Vertical Tabs*.
   - To hide and unhide cards and tabs, see *Hiding and Unhiding Cards and Tabs*.
Creating and Duplicating Navigation Flows

To create a navigation flow, you must first select an existing navigation flow and make a copy of it. Then edit the duplicate navigation flow details and save them.

To create and duplicate a navigation flow:

2. Select the navigation flow you want to copy, then in the upper right-hand corner of the page, click , and then select Create Copy.
3. Enter a name for the navigation flow, and then click OK.

**Note:**

Ensure that you adhere to the navigation flow naming restrictions outlined in Viewing and Working with Navigation Flows.


**Note:**

New flows are marked Inactive until they are activated by the administrator. To activate a navigation flow, see Activating and Deactivating Navigation Flows.

To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

Editing a Navigation Flow

To edit a navigation flow:

2. Click the name of the navigation flow you want to edit.
3. Edit details of the navigation flow:

- **Name**—The name is editable if the navigation flow isn’t a predefined navigation flow.

**Note:**

Ensure that you adhere to the navigation flow naming restrictions outlined in Viewing and Working with Navigation Flows.

- **Assign To**—Click to assign the navigation flow to a group of users or to a role.
- **Visible**—Indicates whether a card in the navigation flow is visible on the Home page to the users in the group.
- **Order**—The cards within the navigation flow are listed in the order in which they are displayed on the Home page, if visible. Selecting an up or down arrow option repositions the cards in the listing and changes the display order of the cards on the Home page. Selecting the right arrow moves a card into a cluster.
- **Remove**—Removes a card or cluster from the navigation flow.
- **Add Card**—Adds a new card to the navigation flow. See Adding Cards.
- **Add Cluster**—Adds a new cluster to the navigation flow. See Grouping Cards into Clusters.
- **Add Existing Card/Cluster**—Adds an existing card or cluster to the navigation flow.

To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

### Activating and Deactivating Navigation Flows

You can create multiple navigation flows for each category (global, role, or group), but only one navigation flow can be active in each category. Whenever you make a navigation flow active, the other navigation flows in the same category will become inactive.

**Note:**

The application requires one active global navigation flow. To make a different global navigation flow active, select another global navigation flow and activate it.
For information about categories, see Navigation Flow Customization Categories.

These are the operations users can and cannot perform on an active navigation flow:

- Name—Users can’t modify the name.
- Delete—Users can’t delete the navigation flow.
- Edit:
  - Users can view the navigation flow definition, but they can’t change anything.
  - If the application is in administration mode, then users can save any modifications.
- Activate or Deactivate—Users can activate or deactivate a navigation flow.
- Duplicate—Users can make a copy of a navigation flow.

To activate or deactivate a navigation flow:

2. In the Active column, click Active or Inactive. An active flow will be marked inactive. Conversely, an inactive flow will be marked active.

Customizing Labels for Cards, Tabs, and Clusters

You can customize the labels for cards (the icons that display on the Home page), tabs, and clusters. Labels are limited to 25 characters or less. For vertical tabs, there is no character limitation since the label for vertical tabs displays as hover text.

To customize labels for cards, tabs, and clusters:

1. Open the Navigation Flow page and click the name of the navigation flow you want to edit. See Viewing and Working with Navigation Flows.
2. If customizing the label for a card or cluster:
   a. Click the name of the card or cluster you want to edit.
   b. Enter a new name and save it.

   **Note:**

   - You can edit the label here. However, if the label is defined in the Artifact Labels page on the Tools cluster, that definition will take precedence and will display during runtime. To change a label permanently, redefine it in the Artifact Labels page. See "Specifying Artifact Labels" in your Administering guide.
   - Ensure that you adhere to the naming restrictions outlined in Viewing and Working with Navigation Flows.

3. If customizing the label for a tab:
   a. Click the name of the card you want to edit.
   b. In the tab listing on the Manage Tab page, click the name of the tab you want to edit.
c. Enter a new name for the tab and save it.

To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

Customizing Icons for Cards and Vertical Tabs

You can change the icons used for cards and vertical tabs. You must pick from the available icons provided in the icon library.

To customize the icons for cards and vertical tabs:

1. Open the Navigation Flow page and click the name of the navigation flow you want to edit. See Viewing and Working with Navigation Flows.

2. If customizing the icon for a card:
   a. Click the name of the card you want to edit.
   b. Click the icon for the card, select a new icon from the library, and then save it.

3. If customizing the icon for a tab:
   a. Click the name of the icon you want to edit.
   b. In the tab listing on the Manage Tab page, click the name of the tab you want to edit.
   c. Click the icon for the tab, select a new icon from the library, and then save it.

To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

Hiding and Unhiding Cards and Tabs

You can't hide the following navigation elements:

- The Application cluster and the Settings icon in the Application cluster.
- The Tools cluster and these icons in the Tools cluster:
  - Access Control
  - Navigation Flows
  - Daily Maintenance
  - Migration

To hide and unhide cards and tabs:

1. Click the Navigation Flow icon and click the name of the navigation flow you want to edit. See Viewing and Working with Navigation Flows.

2. If hiding or unhiding a card:
   a. Click the name of the card you want to edit.
   b. Set Visible to Yes or No.

3. If hiding or unhiding a tab:
   a. Click the name of the card you want to edit.
   b. In the tab listing on the Manage Tab page, click the name of the tab you want to edit.
Changing the Display Order of Cards on the Home Page

You can change the display order of cards in the Navigation Flow Designer. Cards display on the Home page in the order they appear within the listing.

To change the display order of the cards on the Home page:

1. Open the **Navigation Flow** page. See [Viewing and Working with Navigation Flows](#).
2. In the listing, use the up and down arrows in the **Order** column to move cards up or down in the navigation flow order.

To reload a navigation flow to view design time changes, see [Reloading a Navigation Flow](#).

Adding Cards

The icons you see on the application Home page are called cards. Cards are specific to each functional area of the application. Each card navigates users to the corresponding area, where other information is displayed as one or more tabbed pages. You can create single page or multiple page (tabular) cards.

You can also group cards into clusters. See [Grouping Cards into Clusters](#).

To add cards to a navigation flow:

1. Open the **Navigation Flow** page and click the name of the navigation flow you want to edit. See [Viewing and Working with Navigation Flows](#).
2. To add an existing card to the navigation flow, click ‡, click **Add Existing Card/Cluster**, and then select a card. If there is an existing card you want to add from another subscription, click ‡, click **Add Existing Card/Cluster**, select the target subscription under **My Connections**, and then choose the card you want to add to your navigation flow.

Note the following:

- Cards can’t be directly selected from Narrative Reporting and Oracle Profitability and Cost Management Cloud using the **Add Existing Card/Cluster** option.
- Cards that are added from another navigation flow or from another subscription will display the localized labels that were defined in the source navigation flow. To update the card labels in your navigation flow, on the Home page, click **Tools**, and then click **Artifact Labels**. See “Specifying Artifact Labels” in your Administering guide.

3. To add a new card to the navigation flow, click ‡, click **Add Card**, and then select details for the new card:
   - **Name**—Enter a label for the card.
Note:
Ensure that you adhere to the naming restrictions outlined in Viewing and Working with Navigation Flows.

- **Visible**—Select whether the card is visible to users on the Home page.
- **Cluster**—If clusters exist, select a cluster for the card or select **None**.
- **Icon**—Select the graphic that will be displayed for the card you’re creating. Choose from the available graphics provided in the graphics library.
- **Content**—Select from the following options:
  - **Page Type**—Select a single page or a multiple page (tabular) format.
  - **Artifact**—Click to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include simple forms, dashboards, and financial reports. To select an artifact from another subscription, select the target subscription under **My Connections**, and then choose the artifact you want to add.

Note:
Composite forms are not supported in navigation flows. Oracle recommends that you create a dashboard instead.

- **Orientation**—For a card with a tabular page format, select **Vertical** or **Horizontal**.
  - For a page with a tabular format, add new or existing tabs and sub tabs. See Adding Tabs to a Tabular Page.

4. Click **Save and Close**.

To reload a navigation flow to view design time changes, see **Reloading a Navigation Flow**.

### Adding Tabs to a Tabular Page

Tabs can be horizontal or vertical. For example, the **Valid Intersections** card (under the **Application** cluster) is a tabular page with two horizontal tabs at the top of the page: **Setup** and **Reports**.

You can also create tabular pages with vertical tabs. Vertical tabs display a graphic and text appears when the cursor is hovered over the tab. Horizontal tabs display text labels only.

To add tabs to a tabular page:

1. Open the **Navigation Flow** page and click the name of the navigation flow you want to edit. See **Viewing and Working with Navigation Flows**.

2. Add a new card by clicking **Add Card** or edit an existing card by clicking the name of the card you want to edit.
3. On the Manage Card page, for Page Type, select Tabular Page. A tab listing displays at the bottom of the Manage Card page.

4. To edit an existing tab, click a tab name from the tab listing, and edit tab details.

5. To add a new or existing tab:
   a. To add an existing tab, click Add Existing Tab.
   b. To add a new tab, click Add New Tab, and then edit tab details.
   c. For Artifact, click ![Artifact](image) to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include simple forms, dashboards, and financial reports. To select an artifact from another subscription, select the target subscription under My Connections, and then choose the artifact you want to add.

6. To add new or existing sub tabs to a tab:
   a. Click the name of a tab in the tab listing.
   b. For Page Type, select Tabular Page.
   c. Click Add New Sub Tab or Add Existing Sub Tab, and then edit sub tab details.
   d. For Artifact, click ![Artifact](image) to select an artifact in the Artifact Library; for example, if the artifact is a form, then select the specific form from the artifact listing. Available artifacts include simple forms, dashboards, and financial reports. To select an artifact from another subscription, select the target subscription under My Connections, and then choose the artifact you want to add.

7. Click Save and Close.

To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

**Note:**

- For cards with multiple tabs or sub tabs, the last tab accessed by a user will be retained the next time the user accesses the card in the same session. If the user logs out and then logs back in, the default tab will be displayed.
- Tabs or sub tabs that are added from another navigation flow or from another subscription will display the localized labels that were defined in the source navigation flow. To update the tab labels in your navigation flow, on the Home page, click Tools, and then click Artifact Labels. See “Specifying Artifact Labels” in your Administering guide.

### Removing Navigation Flows, Cards, and Tabs

You can't remove the following navigation elements:

- The Application cluster and the Settings icon in the Application cluster.
- The Tools cluster and these icons in the Tools cluster:
To remove navigation flows, cards, and tabs:


2. If removing a navigation flow:
   a. Select the navigation flow you want to remove.
   b. In the upper right-hand corner of the page, click ⚙, and then select Delete.

   **Note:**
   You can’t delete the predefined navigation flow, called Default.

3. If removing a card:
   a. Click the name of the navigation flow you want to edit.
   b. In the Remove column for the card you want to remove, click ✗.

4. If removing a tab:
   a. Click the name of the navigation flow you want to edit.
   b. Click the name of the card you want to edit.
   c. In the tab listing at the bottom of the Manage Tab page, in the Remove column for the tab you want to remove, click ✗.

To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

### Grouping Cards into Clusters

A cluster is a grouping of cards. You must first create a cluster and then you can assign cards to it. You can also add existing clusters to navigation flows.

To group cards into clusters:

1. Create a new cluster or add an existing cluster:
   a. Open the Navigation Flow page and click the name of the navigation flow in which you want to add a cluster. See Viewing and Working with Navigation Flows.
   b. To create a new cluster, click +, click Add Cluster, enter or select the cluster details, and then choose a graphic for the cluster.
c. To add an existing cluster, click +, click Add Existing Card/Cluster. If there is an existing cluster you want to add from another subscription, click +, click Add Existing Card/Cluster, select the target subscription under My Connections, and then choose the cluster you want to add to your navigation flow.

**Note:**

- Clusters can't be directly selected from Narrative Reporting and Oracle Profitability and Cost Management Cloud using the Add Existing Card/Cluster option.
- Clusters that are added from another navigation flow or from another subscription will display the localized labels that were defined in the source navigation flow. To update the cluster labels in your navigation flow, on the Home page, click Tools, and then click Artifact Labels. See "Specifying Artifact Labels" in your Administering guide.

d. Click Save and Close.

The newly added cluster displays in the listing.

2. Select the cards to include in the cluster:

   a. To the right of each card that you want to move, in the Order column, click ▼.

   b. Select the cluster, and then click OK.

   The cards will appear in the listing as children of the cluster. Use the up and down arrows next to the cards to reorder the cards within the cluster, if needed.

   To reload a navigation flow to view design time changes, see Reloading a Navigation Flow.

Reloading a Navigation Flow

To display design changes while you're working with a navigation flow, you can reload the navigation flow.
To reload a navigation flow after making design changes:

1. From the Home page, click the down arrow next to the user name (upper right corner of the screen).

Switching Navigation Flows at Runtime

If you belong to multiple groups or if a navigation flow is assigned to a role, you might have access to more than one navigation flow.

To switch navigation flows at runtime:

1. From the Home page, click Navigator.
2. On the Navigator menu, click the down arrow next to the title of the navigation flow.
3. Select the navigation flow that you want to view.
Why Is My Referenced Card, Tab, or Cluster Not Visible?

When customizing navigation flows to connect EPM Cloud subscriptions, at times the connected artifacts may not be visible. Here are some common reasons why:

• The target subscription is down; for example, to perform routine daily maintenance.

• The password provided in the connection definition is expired.

• The user name provided in the connection no longer has an administrator role.

• The product version of one of the subscriptions has changed; for example, 17.09.xxx and 17.10.xxx don’t communicate because both subscriptions must be the same version.

Note the following:

– This version restriction only applies to the first four digits of the version number.

– This version restriction impacts customers who have multiple EPM Cloud subscriptions and who might have stopped taking monthly updates for a given subscription.

• The password was not setup again after performing a Migration. Since passwords are not migrated, you must ensure the password is setup again after the Migration backup is restored.

• The referenced navigation flow was deleted on the target subscription, or:
  – The referenced navigation flow artifact (card, tab, or cluster) was deleted.
  – The referenced artifact from the target flow was deleted or renamed.
  – The currently logged in user’s access to the given artifact was revoked.
Editing Dimensions in the Simplified Dimension Editor

Related Topics

- About Editing Dimensions in the Simplified Dimension Editor
  Work with application dimensions and members in the simplified grid interface.
- Dimension Overview
- Accessing the Simplified Dimension Editor
- Working with the Simplified Dimension Editor Grid
- Editing Dimension Properties in the Simplified Dimension Editor
- Editing Member Properties in the Simplified Dimension Editor
- Setting Up Scenarios and Specifying Versions

About Editing Dimensions in the Simplified Dimension Editor

Work with application dimensions and members in the simplified grid interface.

The Simplified dimension editor displays dimensions and members in a grid format. With the grid format, dimensions and members are editable on a single page. You can edit member properties directly on the grid and you can perform ad hoc operations such as zooming in, zooming out, keep selected, remove selected, and freeze.

Watch this overview video for an introduction to the Simplified dimension editor.

Overview Video

Related topics:

- Creating Dimensions
- Dimension Overview
- Accessing the Simplified Dimension Editor
- Working with the Simplified Dimension Editor Grid
- Editing Dimension Properties in the Simplified Dimension Editor
- Editing Member Properties in the Simplified Dimension Editor

Dimension Overview

Related Topics

- About Dimensions and Members
- About Sparse and Dense Dimensions
About Dimensions and Members

Dimensions categorize data values. Seven dimensions are included with the application: Account, Entity, Scenario, Version, Period, Years, and Currency. You can create up to 13 user-defined custom dimensions.

Members are components of dimensions.

About Sparse and Dense Dimensions

Sparse dimensions lack data values for the majority of member combinations. Dense dimensions have data values for the majority of member combinations. At least one dense dimension is required. Custom attributes can't be assigned to dense dimensions. The application designates the Account and Period dimensions as dense, and the remaining dimensions as sparse. To optimize performance for sparse dimensions, the application searches for and calculates only occupied data values in each dimension combination, reducing calculation time and disk usage. You can modify these settings.

About Dimension Hierarchies

Dimension hierarchies define structural and mathematical relationships, and consolidations between members in the application. Relationships are represented graphically in a collapsible hierarchy diagram. The levels below the cube name are dimensions, and the levels below each dimension are members.

The Period dimension can contain the member YearTotal, which contains members Q1, Q2, Q3, and Q4. Members Q1, Q2, Q3, and Q4 contain their own members for the corresponding months in the year. To consolidate data values in the Period dimension, roll up monthly data values to get quarterly data values, and quarterly data values to get yearly data values.

Members of the same level that belong to the same dimension or member are called siblings. For example, Q1, Q2, Q3, and Q4 are siblings because they are at the same level in the hierarchy, and are members of the same member, YearTotal.

The members of a dimension are called children of the dimension. Members that belong to a member are called children of that member. The member YearTotal is a child of Period, the members of Q1, Q2, Q3, and Q4 are children of YearTotal, and Jan, Feb, and Mar are children of Q1. Q1 is the parent of Jan, Feb, and Mar, YearTotal is the parent of Q1, Q2, Q3, and Q4, and Period is the parent of YearTotal.

About Custom Dimensions

The application includes two custom dimensions: Account and Entity. You can edit the names of these dimensions, and create up to 13 user-defined dimensions.
Caution:
You can't delete custom dimensions after you create them.

Use Account and user-defined dimensions to specify data to gather from planners. Use Entity to model the flow of planning information in the organization and establish the plan review path.

- Aggregation Options
- Storage Options

Aggregation Options
You can define calculations within dimension hierarchies using aggregation options. Aggregation options determine how child member values aggregate to parent members:

- + Addition
- - Subtraction
- * Multiplication
- / Division
- % Percent
- ~ Ignore
- Never (don't aggregate, regardless of hierarchy)

Storage Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Calc and Store</td>
<td>Calculates data values of members, and stores values.</td>
</tr>
<tr>
<td>Store</td>
<td>Stores data values of members.</td>
</tr>
<tr>
<td>Dynamic Calc</td>
<td>Calculates data values of members, and disregards the values.</td>
</tr>
<tr>
<td>Never Share</td>
<td>Prohibits members in the same dimension from sharing data values.</td>
</tr>
<tr>
<td>Shared</td>
<td>Allows members in the same dimension to share data values.</td>
</tr>
<tr>
<td>Label Only</td>
<td>Has no data associated with the member.</td>
</tr>
</tbody>
</table>

About Dynamic Calc
With dynamically calculated members, the application calculates data values of members, and disregards these values. The limit is 100 children under a Dynamic Calc parent. Changing a member's storage to Dynamic Calc may result in loss of data, de-
pending on how the data was originally derived. You may need to update outlines, calculations, or both to get the dynamically calculated value.

**Dynamic Calc Versus Dynamic Calc and Store**

In most cases, you can optimize calculations and lower disk usage by using Dynamic Calc instead of Dynamic Calc and Store when calculating members of sparse dimensions. Use Dynamic Calc and Store for members of sparse dimensions with complex formulas, or that users retrieve frequently.

For members of dense dimensions, use Dynamic Calc. Dynamic Calc and Store provides only a small decrease in retrieval time and regular calculation time, and doesn't significantly lower disk usage. For data values accessed concurrently by many users, use Dynamic Calc. Retrieval time may be significantly lower than for Dynamic Calc and Store.

**Note:**

- Don't use Dynamic Calc for base-level members for which users enter data.
- Don't use Dynamic Calc for a parent member if you enter data for that member in a target version. Parent members set to Dynamic Calc are read-only in target versions.
- Data values are not saved for Dynamic Calc members.

**About Store Data Storage**

Don't set parent members to Store if their children are set to Dynamic Calc. With this combination, new totals for parents are not calculated when users save and refresh forms.

**About Shared Data Storage**

Use Shared to allow alternate rollup structures in the application.

**About Never Share Data Storage**

The default data storage type is Never Share when you add user-defined custom dimensions. You can use Never Share for parent members with only one child member that aggregates to the parent, to apply access to the child member.

**About Label Only Data Storage**

Label-only members are virtual members; they are typically used for navigation and have no associated data.
Note:

- You can't assign level 0 members as label-only.
- Label-only members can display values.
- Making dimension members label-only minimizes database space by decreasing block size.
- You can't assign attributes to label-only members.
- In a multicurrency application, you can't apply label-only storage to members of these dimensions: Entity, Versions, Currencies, and user-defined custom dimensions. To store exchange rates, use Never Share.
- Data Storage for children of label-only parents is set to Never Share by default.

Caution:

Don't design forms in which label-only parents follow their first child member, as you can't save data in the first child member. Instead, create forms with label-only parents selected before their children, or don't select label-only parents for forms.

About Entities

Entities typically match your organization's structure, such as geographical regions, departments, or divisions. Create entity members for groups that submit plans for approval. Entity members help define budget review, or approvals (see Managing Approvals).

For example, you may have regional centers preparing budgets for country headquarters. The country headquarters may prepare plans for corporate headquarters. To match this structure, create members for the regions, countries and headquarters. Specify regions as children of country members, and country members as children of headquarters.

Forms support multiple currencies per entity, enabling data entry for multiple currencies and reporting against one currency. However, the application supports a base entity for each entity. You can set the currency for entered values, which are converted to other currencies having defined exchange rates.

About Accounts

Account dimension members specify the information needed from budget planners. Create an account structure that lets budget preparers input data for budget items. You can define calculations in the account structure.

- Account Types
- Saved Assumptions
• Data Type and Exchange Rate Type

Account Types

Account type defines accounts’ time balance (how values flow over time) and determines accounts’ sign behavior for variance reporting with member formulas.

Examples of Using Account Types

Table 8-2 Using Account Types

<table>
<thead>
<tr>
<th>Account Type</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expense</td>
<td>Cost of doing business</td>
</tr>
<tr>
<td>Revenue</td>
<td>Source of income</td>
</tr>
<tr>
<td>Asset</td>
<td>Company resource</td>
</tr>
<tr>
<td>Liability and Equity</td>
<td>Residual interest or obligation to creditors</td>
</tr>
<tr>
<td>Saved assumption</td>
<td>Centralized planning assumptions ensuring consistency across the application</td>
</tr>
</tbody>
</table>

Summary of Account Types

Table 8-3 Summary of Account Types

<table>
<thead>
<tr>
<th>Account Type</th>
<th>Time Balance</th>
<th>Variance Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revenue</td>
<td>Flow</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Expense</td>
<td>Flow</td>
<td>Expense</td>
</tr>
<tr>
<td>Asset</td>
<td>Balance</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Liability</td>
<td>Balance</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Equity</td>
<td>Balance</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Saved Assumption</td>
<td>User-defined</td>
<td>User-defined</td>
</tr>
</tbody>
</table>

Variance reporting and time balance settings are system-defined; only Saved Assumption is user-defined.

Time Balance Property

Time balance specifies how the application calculates the value of summary time periods.

Table 8-4 Time Balance Properties

<table>
<thead>
<tr>
<th>Time Balance Property</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow</td>
<td>Aggregate of all values for a summary time period as a period total.</td>
<td>Jan: 10 Feb: 15 Mar: 20 Q1: 45</td>
</tr>
<tr>
<td>First</td>
<td>Beginning value in a summary time period as the period total.</td>
<td>Jan: 10 Feb: 15 Mar: 20 Q1: 10</td>
</tr>
<tr>
<td>Balance</td>
<td>Ending value in a summary time period as the period total.</td>
<td>Jan: 10 Feb: 15 Mar: 20 Q1: 20</td>
</tr>
<tr>
<td>Time Balance Property</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Average</td>
<td>Average for all the child values in a summary time period as the period total.</td>
<td>Jan: 10 Feb: 15 Mar: 20 Q1: 15</td>
</tr>
<tr>
<td>Fill</td>
<td>The value set at the parent is filled into all its descendants. If a child value changes, the default aggregation logic applies up to its parent. Consolidation operators and member formulas overwrite Fill values when the members are recalculated.</td>
<td>Jan: 10; Feb: 10; Mar: 10; Q1: 30</td>
</tr>
<tr>
<td>Weighted Average - Actual_Actual</td>
<td>Weighted daily average, based on the actual number of days in a year; accounts for leap year, in which February has 29 days. In the example, the average for Q1 is calculated: (1) Multiply each month's value in Q1 by the number of days in the month, (2) Sum these values, (3) Divide the total by the number of days in Q1. Assuming it's a leap year, the result is calculated: (10 * 31 + 15 * 29 + 20 * 31) / 91 = 15</td>
<td>Jan: 10 Feb: 15 Mar: 20 Q1: 15</td>
</tr>
<tr>
<td>Weighted Average - Actual_365</td>
<td>Weighted daily average, based on 365 days in a year, assuming that February has 28 days; doesn't account for leap years. In the example, the average for Q1 is calculated: (1) Multiply each month's value in Q1 by the number of days in the month, (2) Sum these values, (3) Divide the total by the number of days in Q1. Assuming it's not a leap year, the result is calculated: (10 * 31 + 15 * 28 + 20 * 31) / 90 = 15</td>
<td>Jan: 10 Feb: 15 Mar: 20 Q1: 15</td>
</tr>
</tbody>
</table>

You can use the Weighted Average - Actual_Actual and Weighted Average - Actual_365 time balance properties only with a standard monthly calendar that rolls up to four quarters. For information on how the application calculates and spreads data with the different Time Balance settings, see *Working with Planning*.

**Account Types and Variance Reporting**

An account’s variance reporting property determines whether it’s treated as an expense when used in member formulas:

- **Expense**—The actual value is subtracted from the budgeted value to determine the variance
- **Non-Expense**—The budgeted value is subtracted from the actual value to determine the variance
Setting Account Calculations for Zeros and Missing Values

With time balance properties First, Balance, and Average, specify how database calculations treat zeros and missing values with the Skip options.

### Table 8-5 Effect of Skip Options When Time Balance is Set to First

<table>
<thead>
<tr>
<th>Skip Option</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Zeros and #MISSING values are considered when calculating parent values (the default). In the example, the value of the first child (Jan) is 0, and zeros are considered when calculating the parent value, so Q1 = 0.</td>
<td>Jan: 0 Feb: 20 Mar: 25 Q1: 0</td>
</tr>
<tr>
<td>Missing</td>
<td>Excludes #MISSING values when calculating parent values. In the example, the value of the first child (Jan) is #MISSING, and #MISSING values are not considered when calculating parent values, so Q1 = second child (Feb), or 20.</td>
<td>Jan: #MISSING Feb: 20 Mar: 25 Q1: 20</td>
</tr>
<tr>
<td>Zeros</td>
<td>Excludes zero values when calculating parent values. In the example, the value of the first child (Jan) is 0, and zero values are not considered when calculating parent values, so Q1 = the second child (Feb), or 20.</td>
<td>Jan: 0 Feb: 20 Mar: 25 Q1: 20</td>
</tr>
<tr>
<td>Missing and Zeros</td>
<td>Excludes #MISSING and zero values when calculating parent values. In the example, the value of the first child (Jan) is zero, and the value of the second child (Feb) is missing. Because missing and zero values are not considered when calculating parent values, Q1 = the third child (Mar), or 25.</td>
<td>Jan: 0 Feb: #MISSING Mar: 25 Q1: 25</td>
</tr>
</tbody>
</table>

Saved Assumptions

Use saved assumptions to centralize planning assumptions, identifying key business drivers and ensuring application consistency. You select time balance and variance reporting properties.

- Variance reporting determines the variance between budgeted and actual data, as an expense or non-expense.
- Time balance determines the ending value for summary time periods.

Examples of how time balance and variance reporting properties are used with saved assumption account members:

- Create a saved assumption of an expense type for variance reporting, assuming that the actual amount spent on headcount is less than the amount budgeted. To determine the variance, the application subtracts the actual amount from the budgeted amount.
• Determine the value for office floor space by using the time period’s last value.
• Make an assumption about the number of product units sold at the end of the time period. Determine the final value for the summary time period by aggregating the number of units sold across time periods.

Data Type and Exchange Rate Type

Data type and exchange rate type determine how values are stored in account members, and the exchange rates used to calculate values. Available data type for account members' values:
• Currency—Stores and displays in the default currency.
• Non-currency—Stores and displays as a numeric value.
• Percentage—Stores a numeric value and displays as a percent.
• Date—Displays as a date.
• Text—Displays as text.

In a multiple currency application, for accounts with the Currency data type, available Exchange Rate types (valid for any time period):
• Average—Average exchange rate
• Ending—Ending exchange rate
• Historical—Exchange rate in effect when, for example, earnings for a Retained Earnings account were earned or assets for a Fixed Assets account were purchased.

In a Simplified multiple currency application, for accounts with the Currency data type, available Exchange Rate types (valid for any time period):
• FX Rates - Average
• FX Rates - Ending

Accounts, Entities, Periods, and Cubes

By assigning cubes for Account, Entity, and Period members, you set to which cubes the members' children have access. For example, Total Sales Account may be valid for Revenue and P&L, but Fixed Assets Account may be valid for only Balance Sheet. Not assigning a cube to a member prevents that member's children from accessing that cube.

When moving members, if the new parent is valid for different cubes, members remain valid only for cubes they have in common with the new parent. If the new parent of an account member has another source cube, the member's source cube is set to the first new valid cube of that member.

Entities and Cubes

Typically, entity members prepare different plans. When defining entity members, specify cubes for which they are valid. Because forms are associated with cubes, you can control which entity members can enter data for each cube.
Accounts and Cubes

If accounts are valid for multiple cubes, specify the source cube to determine which cube's database stores the account value for them.

Accessing the Simplified Dimension Editor

To access the Simplified dimension editor:

1. From the Home page, click Application, and then click Overview.
2. Click Dimensions.
3. Click the down arrow to the right of Cube to filter the list of dimensions by cube.
4. Click the name of the dimension that you want to view.
5. Select from the following tabs:
   - Edit Dimension Properties—Click to view and edit dimension details. See Editing Dimension Properties in the Simplified Dimension Editor.
   - Edit Member Properties—Click to view and edit dimension members. See Editing Member Properties in the Simplified Dimension Editor.

Working with the Simplified Dimension Editor Grid

Related Topics

- Switching to Another Dimension
- Customizing the Column Layout
- Viewing Ancestors
- Showing Member Usage in an Application
- Focusing Your Editing
- Finding Members
- Moving Members
- Sorting Members
- Moving Members to Another Hierarchy
- Working with Member Formulas
- Copying Member Names from Microsoft Excel

Switching to Another Dimension

To switch to another dimension while viewing the Simplified dimension editor grid:

1. View Edit Member Properties.
   - See Accessing the Simplified Dimension Editor.
2. Click the down arrow next to the dimension name at the top of the page.
Customizing the Column Layout

Each column in the Simplified dimension editor grid represents a member property (Member Name, Parent Member, Default Data Storage, and so on). The columns that initially display on the grid can be different based on which dimension type you’re editing. You can customize the layout of columns in the Simplified dimension editor grid by hiding, unhiding, or resizing columns. You can also display the complete set of properties (all columns) by clearing the Default mode option; for example, by clearing Default mode, you can view the UDA column for the Account dimension (the UDA column is hidden in Default mode).

To customize the column layout in the Simplified dimension editor grid:

1. View Edit Member Properties.
   
   See Accessing the Simplified Dimension Editor.

2. On the dimension grid, right-click any column heading.
   
   A checklist of columns displays. Also displayed are column or grid resizing options.

3. Select or clear the check boxes for the columns you want to hide or unhide on the grid.

   Note:
   
   To view all property columns in the grid, clear the Default mode check box. Default mode is selected by default, and limits the properties that are displayed. Clearing this option displays a larger (complete) set of properties (as columns).

4. To change the size of the grid or the columns that are displayed on the grid, select or clear the following resizing options:
   
   - Force fit columns—Resizes the columns so that all columns are visible on the grid without scrolling.
   
   - Synchronous resize—Resets the size of the grid to original settings.

Viewing Ancestors

Ancestors are all the members above the selected member in the dimension hierarchy.

To view the ancestors for the selected member in the Simplified dimension editor grid:

1. View Edit Member Properties.
   
   See Accessing the Simplified Dimension Editor.

2. Select a member on the dimension editor grid.

3. Click Show Ancestors.
Showing Member Usage in an Application

Before performing such operations as deleting members, it's important to understand where in the application the members are used (in which forms, approval units, exchange rates, and so on) by using **Show Usage**.

To view where members are used in an application using the Simplified dimension editor:

1. View **Edit Member Properties**.  
   See **Accessing the Simplified Dimension Editor**.
2. Select a member on the dimension editor grid.
3. Click **Show Usage**.

Focusing Your Editing

Use zoom in, zoom out, keep selected, remove selected, and freeze to focus your editing while working in the Simplified dimension editor grid.

To perform these operations while viewing the dimension editor grid:

1. View **Edit Member Properties**.  
   See **Accessing the Simplified Dimension Editor**.
2. To focus your editing on specific members in the grid, select a member, and then click one of the following zoom operations:
   • **Zoom In Next level**—Displays all members one level below the selected member
   • **Zoom In All levels**—Displays all descendant members below the selected member
   • **Zoom In Bottom level**—Displays all descendant members of the selected member that have no children
   • **Zoom Out**—Displays the member one level above the selected member
3. To focus your editing on specific rows or columns in the grid, select a row or column, and then choose from the following operations:
   • **Keep Selected**—Displays only the selected row or column on the grid
   • **Remove Selected**—Removes the selected row or column from the grid
   • **Freeze** (Columns only)—Keeps the selected column and all columns to the left of the selected column stationary so the column or columns can't be scrolled. Only columns to the right of the frozen column can be scrolled. For example, you can freeze the first column that includes the member name so that you can scroll and edit that member's properties and still see the member name. To unfreeze columns, click **Freeze** once again.

Finding Members

To find dimension members in the Simplified dimension editor grid:

1. View **Edit Member Properties** for a dimension.
See Accessing the Simplified Dimension Editor.

2. For Search, select Name, Alias, or Both.
3. Enter the search text (member name, alias, or partial string) for which to search.
4. Click Search Up or Search Down.

Moving Members

To move members:
1. View Edit Member Properties.
   See Accessing the Simplified Dimension Editor.
2. Select the member to move.
3. To move the member up one position, click \( \text{\textup{\textasciitilde}} \). To move the member down one position, click \( \text{\textdownarrow} \).

Sorting Members

You can sort members in ascending or descending order, by children or descendants. Sorting members affects the outline.

To sort members using the Simplified dimension editor:
1. View Edit Member Properties.
   See Accessing the Simplified Dimension Editor.
2. On the dimension grid, select the members whose children or descendants you want to sort.
3. For Sort, select Children or Descendants.
4. Click \( \text{\textup{\textasciitilde}} \) or \( \text{\textdownarrow} \).

**Note:**

- If you do not see the Sort options at the top of the page, click \( \text{\textdownarrow} \).
- Sorting by children affects only members in the level immediately below the selected member. Sorting by descendants affects all descendants of the selected member.

Moving Members to Another Hierarchy

To move members to another hierarchy in the Simplified dimension editor:
1. View Edit Member Properties.
   See Accessing the Simplified Dimension Editor.
2. In the Parent Member column of the grid, type a new parent name for the member you want to move.
3. Click Save.

Working with Member Formulas

You can define or edit member formulas directly in the Simplified dimension editor grid, in the formula bar, or in the **Member Formula** dialog where you can validate member formulas.

You can define member formulas to combine operators, calculation functions, dimension and member names, and numeric constants to perform calculations on members. Member formulas can also include:

- Operator type, function, value, member name, UDA, and so on allowed in formulas.
- Predefined formula expressions, including Smart List values, that expand into a formula or value upon database refresh.

To define or edit member formulas in the Simplified dimension editor:

1. View **Edit Member Properties**. See Accessing the Simplified Dimension Editor.
2. In the **Default Formula** column of the grid, select the member for which you want to define or edit a formula. Define or edit the formula for the member using one of the following options:
   - Click the cell once more in the dimension editor grid to enter or edit the formula.
   - Click within the formula bar above the dimension editor grid, and then enter or edit the formula.
   - Click next to the formula bar, and then enter or edit the formula.

   **Tip:**

   To include member names in formulas, keep the focus on the formula cell in the grid. Press Ctrl while clicking the member name you want to include in the formula. The member name will display in the formula bar.

3. **Optional**: To check the validity of a member formula, click next to the formula bar, and then click **Validate**.
4. Click **Save**.

Copying Member Names from Microsoft Excel

To copy and paste member names from Microsoft Excel:

1. In Excel, highlight the member names in one cell or in a range of cells, and press Ctrl+C to copy the data onto the clipboard.
2. Highlight and select the target cell or cells in the Simplified dimension editor, and then press Ctrl+V.
3. When the Clipboard helper is displayed, press Ctrl+V again. The data is pasted to the Clipboard helper.

4. Click **Paste** to paste the data into the Simplified dimension editor.

### Editing Dimension Properties in the Simplified Dimension Editor

To access the properties of a dimension in the Simplified dimension editor:

1. From the Home page, click **Application**, and then click **Overview**.
2. Click the **Dimensions** tab, and then click the name of the dimension for which you want to view dimension properties.
3. Click the **Edit Dimension Properties** tab.

Dimension properties must conform to guidelines listed in **Naming Restrictions**.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dimension</strong></td>
<td>Enter a name that is unique across all dimensions.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td><strong>Optional</strong>: Enter a description.</td>
</tr>
<tr>
<td><strong>Alias Table and Alias</strong></td>
<td><strong>Optional</strong>: Select an alias table. Enter an alternate name for the dimension. See <strong>Administering Alias Tables</strong>.</td>
</tr>
<tr>
<td><strong>Cube</strong></td>
<td>Select the cubes for which the dimension is enabled. Clearing this option disables all members of the dimension for the deselected cube.</td>
</tr>
<tr>
<td><strong>Two Pass Calculation</strong></td>
<td>Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with Dynamic Calc or Dynamic Calc and Store properties.</td>
</tr>
<tr>
<td><strong>Apply Security</strong></td>
<td>Allow security to be set on the dimension members; must be selected before assigning access rights to dimension members. Otherwise, dimensions have no security and users can access members without restriction. See <strong>Assigning Access to Dimension Members</strong>.</td>
</tr>
</tbody>
</table>
### Table 8-6  (Cont.) Dimension Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Storage</strong></td>
<td>Select a data storage option. The default is Never Share.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Store</strong>—Stores data values of members.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Dynamic Calc and Store</strong>—Calculates data values of members, and stores values.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Dynamic Calc</strong>—Calculates data values of members, and disregards the values.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Never Share</strong>—Prohibits members in the same dimension from sharing data values.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Label Only</strong>—Has no data associated with the member.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Shared</strong>—Allows members in the same dimension to share data values.</td>
</tr>
<tr>
<td></td>
<td>See Storage Options.</td>
</tr>
<tr>
<td><strong>Display Option</strong></td>
<td>Set application default display options for the Member Selection dialog box. Select Member Name or Alias to display members or aliases. Member Name:Alias displays members on the left and aliases on the right. Alias:Member Name displays aliases on the left and members on the right.</td>
</tr>
</tbody>
</table>
Table 8-6  (Cont.) Dimension Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy Type</td>
<td>Available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be Stored.</td>
</tr>
</tbody>
</table>

Note:
For members with a Stored hierarchy type, the only valid cube aggregation options are Addition or Ignore. In a stored hierarchy, the first member must be set to Addition. For members with a Dynamic hierarchy type, all cube aggregation options are valid. Stored hierarchy members that are not children of Label Only members must have Addition set as the consolidation operator. Children of Label Only members can be set to Ignore.

Custom Attributes
Click to Create or Synchronize custom attributes for a dimension.

Editing Member Properties in the Simplified Dimension Editor

Related Topics
- Accessing Edit Member Properties
- Adding Members in the Simplified Dimension Editor
- Editing Members in the Simplified Dimension Editor
Accessing Edit Member Properties

To access the Edit Member Properties tab in the Simplified dimension editor:

1. From the Home page, click Application, and then click Overview.
2. Click Dimensions, and then click the name of the dimension for which you want to view member properties.
3. Click Edit Member Properties.
4. To edit member properties in the Simplified dimension editor grid, click within a grid cell to edit text or to view a drop-down menu from which you can choose member properties. You can also drag and drop member property values over to rows and columns to fill in properties that are the same. See the Member Properties table below for property options and descriptions.

See Working with the Simplified Dimension Editor Grid

<table>
<thead>
<tr>
<th>Note:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Each column in the Simplified dimension editor grid represents a member property. The list of properties (columns) that initially displays on the grid can be different based on which dimension type you're editing. You can customize the layout of columns by hiding, unhiding, or resizing columns. You can also display the complete set of properties (all columns) by clearing the Default mode option. To customize the column layout in the Simplified dimension editor grid, see Customizing the Column Layout.</td>
</tr>
</tbody>
</table>

Members must conform to guidelines listed in Naming Restrictions. Shared members must be consistent with Adding Shared Members in the Simplified Dimension Editor.

Table 8-7 Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Member Name</td>
<td>A name that is unique across all dimension members</td>
</tr>
<tr>
<td>Description</td>
<td>Optional: A description of the member (you can use the same description for multiple members)</td>
</tr>
<tr>
<td>Alias Table</td>
<td>Optional: The alias table which stores the alias name. Enter an alternate name for the member in Alias. See Administering Alias Tables.</td>
</tr>
<tr>
<td>For Account members only: Account Type</td>
<td>Select Expense, Revenue, Asset, Liability, Equity, or Saved Assumption. For descriptions, see Account Types.</td>
</tr>
<tr>
<td>Property</td>
<td>Value</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>For Account members only: <strong>Variance Reporting</strong></td>
<td>If the account type is <strong>Saved Assumption</strong>, select <strong>Expense</strong> or <strong>Non-Expense</strong>. Designate the saved assumption as a revenue, asset, liability, or equity account.</td>
</tr>
<tr>
<td>For Account members only: <strong>Time Balance</strong></td>
<td>Select <strong>Flow</strong>, <strong>First</strong>, <strong>Balance</strong>, <strong>Average</strong>, <strong>Fill</strong>, <strong>Weighted Average - Actual</strong>, or <strong>Weighted Average - Actual_365</strong>. For descriptions, see <strong>Time Balance Property</strong>.</td>
</tr>
<tr>
<td>For Account members only: <strong>Skip</strong></td>
<td>If the account type is <strong>Asset</strong>, <strong>Equity</strong>, or <strong>Liability</strong>, select <strong>None</strong>, <strong>Missing</strong>, <strong>Zeros</strong>, or <strong>Missing and Zeros</strong>. For descriptions, see <strong>Setting Account Calculations for Zeros and Missing Values</strong>.</td>
</tr>
<tr>
<td>For Account members only: <strong>Exchange Rate Type</strong></td>
<td>Select <strong>Average</strong>, <strong>Ending</strong>, or <strong>Historical</strong>. For descriptions, see <strong>Data Type and Exchange Rate Type</strong>.</td>
</tr>
<tr>
<td><strong>Data Type</strong></td>
<td>Select <strong>Currency</strong>, <strong>Non-currency</strong>, <strong>Percentage</strong>, <strong>Date</strong>, or <strong>Text</strong>. For descriptions, see <strong>Data Type and Exchange Rate Type</strong>.</td>
</tr>
<tr>
<td>For Account members only: <strong>Distribution</strong></td>
<td>Sets the weekly distribution. Available for leaf Account members if the option was selected when creating the application and the base time period is 12 months.</td>
</tr>
</tbody>
</table>
### Table 8-7 (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy Type</td>
<td>Available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be stored.</td>
</tr>
</tbody>
</table>

**Note:**
For members with a stored hierarchy type, the only valid cube aggregation options are Addition or Ignore. In a stored hierarchy, the first member must be set to Addition. For members with a dynamic hierarchy type, all cube aggregation options are valid. Stored hierarchy members that are not children of Label Only members must have Addition set as the consolidation operator. Children of Label Only members can be set to Ignore.

**Data Storage**

The data storage property. Never Share is the default for new custom dimension members (except root members).

**Two Pass Calculation**

Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with **Dynamic Calc** or **Dynamic Calc and Store** properties.

*For Entity members only: **Base Currency***

For Standard multiple currency applications only, select the Entity member's base currency.
Table 8-7  (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cube Consol op</td>
<td>Select an aggregation option for each cube:</td>
</tr>
<tr>
<td></td>
<td>• Not used for Cube (member will be invalid for cube)</td>
</tr>
<tr>
<td></td>
<td>• Addition</td>
</tr>
<tr>
<td></td>
<td>• Subtraction</td>
</tr>
<tr>
<td></td>
<td>• Multiplication</td>
</tr>
<tr>
<td></td>
<td>• Division</td>
</tr>
<tr>
<td></td>
<td>• Percent</td>
</tr>
<tr>
<td></td>
<td>• Ignore</td>
</tr>
<tr>
<td></td>
<td>• Never (member doesn't aggregate, regardless of hierarchy)</td>
</tr>
</tbody>
</table>

You can select a source cube only if multiple cubes are valid for the member. Only cubes and aggregation options for which the member's parent is valid are available. If the parent isn't valid for a cube or aggregation option, neither is the child member. Deselecting a cube for an account or entity parent member deselects it for all descendents of that parent. For members with a stored hierarchy type, the only valid aggregation options are Addition or Ignore.

⚠️ Caution: ⚠️

Deselecting a cube for dimension members after data is entered into an application may result in loss of data when an application is refreshed. For account members, data is lost if the deselected cube is the source cube.

You can set usage by cube for members of custom dimensions and the Period dimension, similar to the Account and Entity dimensions.

For Entity members only: **Base Currency**

For Standard multiple currency applications only, select the base currency for the Entity member.
Table 8-7  (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>For Account members only: Source Cube</strong></td>
<td>Select the source cube for the member. A shared member is a pointer to the base member and isn't stored; this is disabled for shared members. The source cube of a shared Account member matches the source cube of the base member, even though Source Plan field is unavailable because it doesn't apply to shared members.</td>
</tr>
<tr>
<td><strong>Smart Lists</strong></td>
<td><strong>Optional</strong>: Select a Smart List to associate with the member.</td>
</tr>
<tr>
<td><strong>Enable for Dynamic Children</strong></td>
<td>Enables users to create children for this member by entering a member name in the runtime prompt for a business rule that has been configured with a dynamic parent member.</td>
</tr>
<tr>
<td><strong>Number of Possible Dynamic Children</strong></td>
<td>This option is available if <strong>Enable for Dynamic Children</strong> is selected. Enter the maximum number of dynamically-added members that users can create. The default is 10.</td>
</tr>
<tr>
<td><strong>Access Granted to Member Creator</strong></td>
<td>This option is available if <strong>Enable for Dynamic Children</strong> is selected. Determines the access that member creators have to dynamic members that they create with a runtime prompt:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Inherit</strong>—The member creator will inherit the closest parent's access to the newly-created member.</td>
</tr>
<tr>
<td></td>
<td>• <strong>None</strong>—The member creator will not be assigned any access to the newly-created member. (An administrator can later assign the member creator access to the members.)</td>
</tr>
<tr>
<td></td>
<td>• <strong>Read</strong>—The member creator will be assigned Read access to the newly-created member.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Write</strong>—The member creator will be assigned Write access to the newly-created member.</td>
</tr>
</tbody>
</table>

**Note:**

If an administrator changes these settings, they affect only future dynamic members; they don't retroactively affect dynamic members.
Adding Members in the Simplified Dimension Editor

Members must conform to guidelines listed in Naming Restrictions.

To add shared members, see Adding Shared Members in the Simplified Dimension Editor.

To add members in the Simplified dimension editor:

1. From the Home page, click Application, and then click Overview.
2. Click the Dimensions tab.
3. Click the down arrow next to Cube to filter the list of dimensions by cube.
4. Click the name of the dimension that you want to update.
5. Click the Edit Member Properties tab.

Tip:
To choose a different dimension, click the down arrow next to the dimension name at the top of the page.

6. Add members:
   - To add a child member, select the parent level member, click Actions, and then click Add Child.

   Note:
   Child members inherit the dimension properties of the parent member.
   - To add a sibling member, select a member, click Actions, and then click Add Sibling.
   - To add years to the calendar, select the Years dimension, click Actions, and then click Add Years. Enter the number of years to add, click Apply, and then confirm whether to add the years to the end of the calendar. Click Yes to add years after the End year. Click No to add years before the Start year.
   - To add an All Years parent member that includes all members of the Years dimension, select the Years dimension, click Actions, and then click Add "All Years".

The All Years parent member enables users to view the accumulated data across multiple years, for example, a project's total cost up to its end date. The All Years member doesn't include the No Year member, if one is defined for the application.

7. To set or change member properties, click a cell in the Edit Member Properties grid and make updates. See Editing Members in the Simplified Dimension Editor.
8. To undo the last change you made before saving, click Undo.
9. To undo all changes that were made since the last save, click Refresh.

Chapter 8
Editing Member Properties in the Simplified Dimension Editor
8-23
10. To save your changes, click **Save**.

11. To apply your changes across the application after a save, click **Refresh Database**.

12. After creating a dimension member, you typically complete these tasks:
   - Assign access. Click **Actions**, and then click **Assign Permissions**.
   - Assign custom attributes. See **Editing Dimension Properties in the Simplified Dimension Editor**.

---

**Editing Members in the Simplified Dimension Editor**

You can edit members directly in the Simplified dimension editor grid.

To edit members:

1. From the Home page, click **Application**, and then click **Overview**.
2. Click the **Dimensions** tab.
3. Click the down arrow next to **Cube** to filter the list of dimensions by cube.
4. Click the name of the dimension that you want to edit.
5. Click the **Edit Member Properties** tab.

**Tip:**

To choose a different dimension, click the down arrow next to the dimension name at the top of the page.

6. Take an action:
   - To add members, see **Adding Members in the Simplified Dimension Editor**
   - To navigate the dimension editor grid and to focus your editing on certain members, rows, or columns, see **Working with the Simplified Dimension Editor Grid**.
   - To modify member properties, click within a cell on the dimension editor grid and select an option from the drop-down list. For example, to edit the Account Type for an Account dimension member, click within a cell in the Account Type column. Click the down arrow that appears within the cell, and then select an Account Type option. For descriptions of member properties, see **Editing Member Properties in the Simplified Dimension Editor**.

**Note:**

To view all property columns in the grid, right-click any heading on the grid and scroll down until you see the **Default mode** check box. Clear the **Default mode** check box to view all the property columns in the grid.

- To delete members, see **Deleting Members in the Simplified Dimension Editor**.
7. To undo the last change you made before saving, click **Undo**.
Deleting Members in the Simplified Dimension Editor

Each data value is identified by a set of dimension member values and a cube. Deleting dimension members or deselecting the cube results in data loss when refreshing an application. Deleting entity members deletes all approval units (including data) associated with them.

Before deleting members, understand where in the application they are used (in which forms, approval units, exchange rates, and so on) by using Show Usage. See Showing Member Usage in an Application.

You must delete the entity member throughout the application before deleting it from dimension. For example, if the entity member is used in a form, you must delete it from the form before deleting it from dimensions.

When deleting a large subtree of entities, you can improve performance if you first exclude approval units for the subtree (by excluding the root member) for all scenarios and versions. See Approvals Process.

To delete members:

1. From the Home page, click Application, and then click Overview.
2. Click Dimensions.
3. Click the down arrow next to Cube to filter the list of dimensions by cube.
4. Click the name of the dimension that contains the member you want to delete.
5. Click Edit Member Properties.
6. From the dimension editor grid, select the member to delete.
7. Click Delete Member.
8. In the Delete Member query, click OK.
9. Update and validate business rules and reports.

Tip:
To choose a different dimension, click the down arrow next to the dimension name at the top of the page.

Note:
Deleting a base member also deletes its shared members.
Adding Shared Members in the Simplified Dimension Editor

Sharing members allow alternate rollup structures within an application. A base member must exist before you can create a shared member. You can create multiple shared members for the base member. A base member must display before its shared members in position from top to bottom.

Shared members are available for Entity, Account, and user-defined custom dimensions. Shared member values can be ignored to avoid double-counting values when you roll up the outline.

Shared members share some property definitions with base members, such as member name, alias name, base currency, and cubes for which members are valid. Shared members must have unique parent members and different rollup aggregation settings. Custom attributes, custom attribute values, and member formulas are not allowed for shared members. Renaming base members renames all shared members.

Shared members can't be moved to another parent member. You must delete shared members and recreate them under different parent members. The base member need not be level zero. You can enter data in shared members, and values are stored with base members.

To add shared members in the Simplified dimension editor:

1. View **Edit Member Properties**.  
   See Accessing the Simplified Dimension Editor.
2. Add shared members:
   a. Add a child member with the same name as the base member.
   b. For **Parent Name** of the new member, type a new parent name.
   c. For **Default Data Storage** of the new member, select **Shared**.
3. Click **Save**.
   Data storage properties for all other cubes will automatically be set to **Shared**.

Assigning Access to Dimension Members

Before you can assign access to members of user-defined custom dimensions, you must select the **Apply Security** check box on the dimension's **Edit Dimension Properties** page. See Editing Dimension Properties in the Simplified Dimension Editor.

To assign access to dimension members:

1. View **Edit Member Properties**.  
   See Accessing the Simplified Dimension Editor.
2. Select the member, and then click 
3. Click **Permissions**.
4. Add, edit, or remove permissions.
   • Adding, Editing, and Removing Access to Members
   • About Effective Access Permissions to Shared Members
Adding, Editing, and Removing Access to Members

You can specify which users and groups can access the selected member.

To assign, edit, and remove access permissions to members:

1. **View Edit Member Properties.**
   
   See Accessing the Simplified Dimension Editor.

2. Select the member, and then click 🕵️.

3. Click **Permissions**.

4. Click **Add User/Group**.

5. To add access:

   a. Click 🔍 to select the users and groups to access the selected member.

   Click **Users** to display all user names; click **Groups** to display all groups.

   b. **Optional**: Select a relationship.

   For example, select **Children** to assign access to the children of the selected member.

6. To modify access:

   a. Select the access type for the displayed users or groups.

   Click **Users** to display all user names; click **Groups** to display all groups.

   b. **Optional**: Select a relationship.

   For example, select Children to assign access to children of the selected member.

7. To remove access, click 🗑️.

8. Click **OK**.

9. Click **Close**.

About Effective Access Permissions to Shared Members

You can't assign access directly to a shared member. A shared member inherits access permissions from its base member, parent, or ancestor.

The application checks access permissions at each level, first by user, then by group, based on the member's access permissions inheritance relationship. If multiple access permissions exist, the least restrictive access permission is applied (for example, Write access takes precedence over Read access).
Sample Parent and Child Entity Members

- **United States**
  - CA (base)
  - NY
- **West**
  - CA (shared)
  - NV
- **Sales Region 1**
  - CA (shared)

Table 8-8  Example of Inherited Access to Shared Members

<table>
<thead>
<tr>
<th>Case</th>
<th>Access Permission</th>
<th>Effective Access for Base and Shared Member CA</th>
<th>Explanation</th>
</tr>
</thead>
</table>
| Case 1 | CA (base) = None  
  iDescendants (West) = Read | Read | CA inherits Read access from its West parent because Read is less restrictive than None. |
| Case 2 | iDescendants (United States) = None  
  iDescendants (West) = Read  
  iDescendants (Sales Region 1) = Write | Write | CA inherits Write access from its Sales Region 1 parent because Write is less restrictive than Read or None. |
| Case 3 | iDescendants (United States) = Write  
  iDescendants (West) = None  
  iDescendants (Sales Region 1) = Read | Write | CA inherits Write access from its United States parent because Write is less restrictive than Read or None. |

Setting Up Scenarios and Specifying Versions

**Related Topics**
- Setting Up Scenarios
- Specifying Versions

**Setting Up Scenarios**

Each scenario/version combination contains data for accounts and other dimensions of each entity. After users enter data for an entity for a scenario and version, they can submit or promote the data for the entity to other users for review and approval.

- About Scenarios
- Creating, Editing, and Deleting Scenarios
• Copying Scenarios

About Scenarios

Use scenarios to:
• Apply different planning methods
• Create forecasts
• Enter data into scenarios
• Associate scenarios with different time periods or exchange rates
• Assign user access rights by scenario
• Report on scenarios
• Compare and analyze scenarios

Time Periods

Assign each scenario a range of years and time periods, and specify the Beginning Balance time period. When users access forms, they can enter into that scenario only years and periods within the range. Years and periods outside of the range display as read-only. You can modify the time range.

Exchange Rate Table

If a Standard multicurrency application converts currencies, assign an exchange rate table to the scenario. By assigning different exchange rate tables to scenarios, you can model the effects of currency rate assumptions.

Access Permissions

Specify access permissions to Scenario dimension members for groups or users to determine who can view or modify data. A user or group can have only one of these access permissions: Read, Write, or None. Access permissions for a user can be combined based on groups to which the user belongs.

Creating, Editing, and Deleting Scenarios

To create, edit, or delete scenarios:

1. From the Home page, click Application, and then click Overview.
2. Click Dimensions, click the Scenario dimension, and then click the Edit Member Properties tab.
3. Choose an action:
   • To create a scenario, click , and then go to step 4.
   • To delete a scenario, select the scenario to delete, then click , and then confirm the deletion.
Note:

When you delete scenarios, all approval units that use the scenario (including data) are deleted. You can't delete scenarios used in approval units that are started, or assigned to an axis on a form. You must first remove references to scenarios from forms and assign different scenarios.

- To edit a scenario, click within the grid cells in the Simplified dimension editor grid to edit text or to view a drop-down menu from which you can choose member properties. Go to step 4.

Note:

Each column in the Simplified dimension editor grid represents a member property. The columns that initially display on the grid will differ based on which dimension type you're editing. You can customize the layout of columns in the Simplified dimension editor grid by hiding, unhiding, or resizing columns. You can also display the complete set of properties (all columns) by clearing the Default mode option. To clear the Default mode option, on the dimension grid, right-click any column heading, scroll down until you see the Default mode option, and then clear it.

See Working with the Simplified Dimension Editor Grid.

4. In the Member Name column, enter a name for the scenario.

5. In the Start Yr., End Yr., Start Period, and End Period columns, define the time periods to associate with the scenario.

6. You can also perform these optional steps:
   - In the Description column, enter a description for the scenario.
   - To update the alias, select one of the Alias Table columns (choose default or a language) to associate with the scenario, and enter an alias name.
   - In the Beg.Bal. as Time Period column, specify whether to include the Beg-Balance time period in this scenario for currency conversion.
   - In the Enabled for Process Management column, specify whether to include this scenario in approvals.
   - In the Exchange Rate Table, select an exchange rate table to associate with the scenario.

Note:

If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversions.

7. Click Save.
Copying Scenarios

Only scenario properties are copied. Data values and access rights associated with the original scenario are not copied to the new scenario.

To copy scenarios:

1. From the Home page, click Application, and then click Overview.
2. Click Dimensions, click the Scenario dimension, and then click the Edit Member Properties tab.
3. In the Simplified dimension editor grid, select the scenario to copy.
4. Click .
   The duplicate scenario is added to the Simplified dimension editor grid and _copy is appended after the member name.
5. In the Member Name column, enter a new name for the duplicate scenario.
6. In the Start Yr., End Yr., Start Period, and End Period columns, define the time periods to associate with the scenario.
7. You can also perform these optional steps:
   • In the Description column, enter a description for the scenario.
   • To update the alias, select one of the Alias Table columns (choose default or a language) to associate with the scenario, and enter an alias name.
   • In the Beg.Bal. as Time Period column, specify whether to include the Beg-Balance time period in this scenario for currency conversion.
   • In the Enabled for Process Management column, specify whether to include this scenario in approvals.
   • In the Exchange Rate Table, select an exchange rate table to associate with the scenario.

Note:
If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversions.

8. Click Save.

Specifying Versions

Use versions to group data used by an application.

• About Versions
• Target and Bottom Up Versions
• Creating, Editing, and Deleting Versions
• Viewing Version Types
About Versions

Use the Scenario and Version dimensions to create plans to be reviewed and approved. Each scenario/version combination contains data for accounts and other dimensions of each entity. After users enter data for an entity for a scenario and version, they can submit or promote the data for the entity to other users for review and approval. Use versions to:

- Allow multiple iterations of a plan
- Model possible outcomes based on different assumptions
- Manage dissemination of plan data
- Facilitate target settings

Target and Bottom Up Versions

You can create target and bottom up versions. With bottom up versions, you enter data into bottom level members; parent level members are display-only and don't permit data entry. Parent member values are aggregated from bottom level members.

For target versions, you can enter data for members at any level in the hierarchy. You can use business rules to distribute values from parent members to their descendants. Use target versions to set high-level targets for your plan. Planners working with bottom up versions can reference these targets when they enter plan data.

Target versions use top-down budgeting. Manage Approvals Tasks are not allowed, and children of target members must be blank (for example, #MISSING) to enable data input at the top level. Target members must be set to Store (Dynamic Calc overrides data input with sum of children).

Creating, Editing, and Deleting Versions

To create, edit, or delete versions:

1. From the Home page, click Application, and then click Overview.
2. Click Dimensions, click the Version dimension, and then click the Edit Member Properties tab.
3. Choose an action:
   - To create a version, click , and then go to step 4.
   - To delete a version, select the version to delete, then click , and then confirm the deletion.

Note:

You can't delete versions that are used in approval units that are started or are assigned to axes on forms. You must remove references to versions from forms and assign another version to axes. At least one version must remain in the application.
• To edit a version, click within the grid cells in the Simplified dimension editor grid to change the version name and access rights. Go to step 4.

**Note:**

Each column in the Simplified dimension editor grid represents a member property. The columns that initially display on the grid will differ based on which dimension type you’re editing. You can customize the layout of columns in the Simplified dimension editor grid by hiding, unhiding, or resizing columns. You can also display the complete set of properties (all columns) by clearing the **Default mode** option. To clear the **Default mode** option, on the dimension grid, right-click any column heading, scroll down until you see the **Default mode** option, and then clear it.

See *Working with the Simplified Dimension Editor Grid.*

4. In the **Member Name** column, enter a name for the version.
5. In the **Type** column, select the type of version to display for the application:
   • **Standard Target**—Values are entered from the parent level down
   • **Standard Bottom Up**—Values are entered at the lowest member level and aggregated upward
6. You can also perform these optional steps:
   • In the **Description** column, enter a description for the scenario.
   • To update the alias, select one of the **Alias Table** columns (choose default or a language) to associate with the version, and enter an alias name.
   • In the **Beg.Bal. as Time Period** column, specify whether to include the Beg-Balance time period in this scenario for currency conversion.
   • In the **Enabled for Process Management** column, specify whether to include this version in approvals.

**Note:**

This option isn't available for target versions.

7. Click **Save**.
8. Update and validate business rules and reports.

**Viewing Version Types**

To view version types:

1. From the Home page, click **Application**, and then click **Overview**.
2. Click **Dimensions**, click the **Version** dimension, and then click the **Edit Member Properties** tab.
3. In the Simplified dimension editor grid, the **Type** column displays these version types:

- **Standard Target**—Values are entered from the parent level down.
- **Standard Bottom Up**—Values are entered at the lowest member level and aggregated up.
Designing Dashboards

Design dashboards that show summary data to planners. Dashboards enable planners to chart, evaluate, highlight, comment on, and even change key business data.

Related Topics

• The Power of Dashboards
• Concepts in Designing Dashboards
• Designing Dashboards
• Creating Dashboards Containing Master Forms and Details
• About Your Dashboard’s Layout
• About the Gauge Chart Type
• About the Tile Chart Type
• Customizing Dashboard Colors
• Setting Line Width in Line and Combination Charts
• About Global and Local POVs
• Dashboard POVs and Valid Intersections

The Power of Dashboards

Dashboards typically provide an overview to planners at the beginning of their planning and forecast process by showing summary data. The versatility of dashboards enables planners to chart, evaluate, highlight, comment on, and even change key business data. They can change a driver such as Volume in a form that’s in a dashboard and immediately see its impact in other forms and charts:

You create dashboards by simply dragging and dropping a variety of objects from the design palette to the dashboard canvas.
With dashboards, you can:

- Include up to six forms that dynamically update, including their associated charts, as planners change data in the form.
- Include up to nine charts or tiles. Tiles display a specific value from the cube. To provide the value for each tile, you can specify a form or a cell intersection as a data source.
- Switch between using the design palette and runtime mode, so you can see exactly what the dashboard user will see.
- Depict data interactively with a wide variety of chart types such as area, bar, bubble, column, combination bar and line, doughnut, funnel, gauge, scatter, radar, and so on.
- Add links to dynamically display external Web pages.
- Finely control the dashboard’s layout. For example, two forms can consume the top half of a dashboard and three charts can each consume 33% of the bottom half.
- Depending on the form design, enable users to drill down into underlying detail and select which members to work with.
- Include user variables in the global POV bar and the local POV.
- Include explanations of data called Commentary in the dashboard.
- In certain charts:
  - Customize the colors and line width.
  - Show gridlines (the default is to hide gridlines).
- Tag a form as master and then filter members from the master form to detail forms or charts within the same dashboard

When planners use a dashboard (referred to as runtime), they can set many aspects of the object, such as the type of chart displayed, the dashboard’s title, and so on. A toolbar is available for each object type. Dashboard users can change and save data, run rules, and so on. However, changes made in runtime to the chart type options aren't saved for the next session. Clicking Save in runtime saves the data, but not the dashboard definition.

Administrators create, redesign, delete, and assign permissions to dashboards and dashboard folders.

Watch this tutorial video to learn how to create dashboards.

Tutorial Video

Related Topics:

- Concepts in Designing Dashboards
- Designing Dashboards
- Creating Dashboards Containing Master Forms and Details
- About Your Dashboard’s Layout
- About the Gauge Chart Type
Concepts in Designing Dashboards

Helpful information as you design dashboards:

- On the left is the design palette. Simply drag and drop objects from the palette to the canvas.

  **Tip:**

  Drag an object to a border line. The drag icon changes to a plus sign when you can drop the object in an allowed space. See About Your Dashboard’s Layout.

- On the top right are settings for the entire dashboard:

  ![Dashboard Settings](image)

  Use **Settings** to set these aspects of the dashboard:

  **Dashboard Settings**

  Use name as title ✓

  Borders 
  ○ Show  ○ Hide

  Layout 
  ○ Fixed  ○ Flexible

  POV Bars 
  ○ Show  ○ Hide

  Global POV Bar 
  ○ Enable  ○ Disable

  **Note:**

  When you create a new dashboard, the borders are hidden by default. To show borders in newly-created dashboards, you must change the Borders setting to Show.

  For more information about POVs, see About Global and Local POVs.

- At the top right of each object is a toolbar for that object that displays when you hover over it:
• As you create a dashboard, click Runtime so that you can immediately see how the dashboard looks and works to a dashboard user. To return to designer mode to continue designing the dashboard, click Designer.

• Missing or suppressed data is plotted as zeros in graphs.

• The listing page for dashboards supports folders. Folders enable you to assign permissions to all dashboards within a folder rather than assigning permissions to each individual dashboard. The dashboard listing page uses the same folder hierarchy as infolets and data entry forms and all artifacts and folders reside under a root folder called Library.

• On the listing page for dashboards, you can toggle between viewing dashboards by a flat view or a tree view:

Then you can search for dashboards using Search. The flat view displays only the dashboards that meet the search criteria, not the folders that contain them. The tree (or hierarchical) view displays dashboards in the context of the folders that contain them.

To search on another keyword, clear the search criteria by clicking X in the Search box.

Designing Dashboards

1. From the Home page, click Dashboards, and then Create.

2. Enter a name.
   • To change the default dashboard name, click its name, and enter a new name in the input box.
   • To give the dashboard a title with custom formatting, click Settings, clear Use name as title, and then enter the title and set formatting the dialog box.

3. From the design palette on the left, drag and drop objects onto the dashboard canvas.
   Select from these objects:
Table 9-1  Dashboard Objects

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forms</td>
<td>Select simple forms to include in the dashboard by navigating the forms folders or by searching for them by name.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The access permissions set for forms are honored in dashboards.</td>
</tr>
<tr>
<td>Chart Types</td>
<td>Select the chart types to include in the dashboard. When first added, a selected chart has sample data. You then associate it with a form as its data source. When you link a chart to a form, users can immediately see the impact of changing data in the form on the associated charts. The Combination chart type alternates displaying row data with vertical bars and lines in the chart. For example, the data in row 1 of a form is displayed as a bar and the data in row 2 as a line, with alternating chart types for even and odd-numbered rows. Although the Combination chart type can display up to 20 rows of data, it's particularly useful when you want to compare two categories of data. For example, you want to compare Germany and France's average exchange rates over several years, so the form has Germany rates in row 1 of the form, and France's rates are in row 2. For information on the Gauge chart type, see About the Gauge Chart Type.</td>
</tr>
<tr>
<td>Tile</td>
<td>Sometimes called performance tiles, a tile is a chart type that lets you select specific values from the cube to display. See About the Tile Chart Type.</td>
</tr>
<tr>
<td>Commentary</td>
<td>Select External Artifacts, and then Commentary. Enter text that explains the data or charts.</td>
</tr>
<tr>
<td>URL</td>
<td>Dynamic web page summary. Select External Artifacts, and then URL. Insert only external site URLs starting with the https:// security protocol. Don't use internal or relative URLs or URLs for unconsenting third party sites such as google.com.</td>
</tr>
</tbody>
</table>

4. Customize the dashboard using the dashboards settings and the objects' hover toolbar, and then click Save.

See About Your Dashboard's Layout.

You can easily duplicate and modify a dashboard using Copy As on the Dashboard list page. Select the Actions icon next to the dashboard you want to copy.

Creating Dashboards Containing Master Forms and Details

You can design dashboards that have one master form and multiple detail (or target) objects (forms or charts). When you do so, the selection of members in the master form automatically filters to the members in the target objects, and the target objects show only the details that are relevant to the members highlighted in the master form.
Considerations when designing dashboards with master forms and target objects:

- Master forms must always be a form (you cannot tag a chart as a master form).
- The target objects can be forms or charts.
- The dashboard containing a master form must always contain more than one object.
- Only one form on a dashboard can be tagged as master. If you want to tag a different form on a dashboard as master, you must first untag the existing master form.

To create a dashboard containing a master form and target objects:

1. Create a dashboard with at least one form and one or more target objects. See Designing Dashboards.
2. Ensure the dashboard is in designer mode by clicking Actions, and then clicking Edit.
3. Select the form on the dashboard that you want to designate as master, click the Settings icon for that form, and then select Tag as Master.

To filter the data in the target forms or charts that is relevant to the data in a master form, right-click the relevant data in the master form and select Apply Context.

About Your Dashboard's Layout

About setting up a dashboard layout:

- The first object you drag occupies the whole canvas.
- Then drag objects to the left, the right, the top, or the bottom of the existing object.
- The dashboard canvas provides two types of drop zones: One to place two objects next to each other, each occupying half the space, and the other to place three objects, each occupying a third of the space.
- You can design a dashboard with objects to display vertically, horizontally, and each can have its own size.
- To resize an object that uses Flexible layout, drag the object's border.
- To resize an object that uses Fixed layout, set its width or height percentage in Settings.
- The form layout can be asymmetric.
- In runtime mode, if a user doesn't have access to a form or if the form is missing, then an adjacent object takes its space. In designer mode, all empty objects are displayed so that the designer can choose to remove them.

About the Gauge Chart Type

Gauge chart types are handy for showing whether data values fall within an acceptable range or not. You set the maximum value, the range maximums, and the gauge displays ranges as red, yellow, and green to help you quickly assess a current value. So, gauge chart types help you identify problems in important data points or measures. For example, you could use a gauge to display the current sales, where the thresholds are set to represent the sales targets.
If the form has multiple values, you can display multiple gauges, up to a maximum of 36 (the values in the first 6 rows and the first 6 columns in the form). The remaining values in the form are ignored. If you want the gauge chart to display only one value, then associate it with a form that has only one cell value.

You can select either a dial gauge or a status meter gauge. You can display a status meter gauge using either horizontal or vertical bars.

Dashboard designers can set:

- **Maximum Value**: The highest value on the gauge. The dashboard designer sets the Maximum Value as a default, and then planners can temporarily change it at runtime. If the dashboard designer doesn't specify a maximum value, the application automatically sets the maximum value as greater than the value on the gauge.

- **Thresholds**:
  - Low, Medium, and High thresholds: To visually indicate whether a measure lies in the acceptable range or not, these thresholds enable you to display the gauge in red, yellow and green based on the specified values.
  - Thresholds where low values are desirable.
  - Appropriate labels for the thresholds that are displayed when hovering over the thresholds in the gauge.

**Settings**

For example, here's a form's data:
Here's the resulting dial gauge:

Here's the resulting status meter gauge with vertical bars:
About the Tile Chart Type

A tile is a chart type that lets you select specific values from the cube to display. In addition to using a form as a data source, you can directly enter a cell intersection that provides the value for a tile. You can have up to 6 tiles across, and 4 rows down in a dashboard, and give them a title. Until you associate a tile with data, it displays sample data.

With a form as the data source for a tile:

- You can have up to six tiles per object.
- The values in the first column (up to the sixth row) are used to create the tiles.

With a cell intersection as the data source for a tile, you can have only one tile per object.

Tip:

To select a tile chart type, expand the list of chart types by clicking the link at the bottom of the list.

Here are options you can set for a tile chart type. Note that you can set whether the displayed value is horizontally aligned in the tile to the left, the center, or the right.
Scaling Large Numbers

Especially useful for large numbers, you can scale how a currency value is displayed. For example, if the tile value is 1,689,000 and you select K as the scaling option, the tile displays the value as 1689K. Your scaling options:

- **None**—No scaling is applied.
- **Auto**—The value is displayed based on its range. For example, 1,500 displays as 1.5K, 1,689,000 displays as 1.69M, 42,314,531,211 displays as 42.31B, and 1,234,567,891,234 displays as 1.23T.
- **K**—The value is displayed as thousands units. For example, 1689000 displays as 1689K.
- **M**—The value is displayed as millions units. For example, 123,456,789 displays as 123M.
- **B**—The value is displayed as billions units. For example, 12,345,678,912 displays as 12B.
- **T**—The value is displayed as trillions units. For example, 1,234,567,891,234,567 displays as 1,234T.

Customizing Dashboard Colors

Maybe your company uses a standard set of colors in charts to denote different types of data. For example, dark blue might represent actual data versus light blue for budget data. When you customize dashboard colors, you select colors in the order of rows on the form. Series 1 represents the first row of data, and so on. You can assign each row in the form a color that represents its data in the chart.

You can customize dashboard colors in Bar, Line, Area, Bubble, Column, Combination, Doughnut, Pie, Radar, and Scatter chart types.

1. With the chart on the dashboard’s design palette, click Settings.
2. Click **Colors**.

3. Clear the **Default** check box, and then click the down arrow for the Series you want to change.

Check the chart in the background to see the data type that each series represents.

4. Click the colors you want for the selected Series, and then click **Close**.

**Note:**

To select more shades of colors than the ones initially displayed, click **Custom Color**.

Your selections apply only to the current chart. Follow these steps to change the colors of other charts in the dashboard.

**Setting Line Width in Line and Combination Charts**

You can set how thin or thick that lines display for Line and Combination chart types in a dashboard.

1. With a Line or Combination chart on the dashboard's design palette, click **Settings**.

2. Click the **Line Weight** counter to set the line width.
You can see the effect of your setting on the chart in the background.

**Note:**

The default width of lines in a Line and Combination chart type is 5 pixels. You can select from 1 to 12 pixels.

### About Global and Local POVs

A local POV on a form reflects the dimension members the form designer selected for that form. Dashboards and composite forms also support *global POV bars*, so that the local POVs that are common are combined in the global POV bar to avoid repeating them in each object. Here’s a dashboard that shows a global POV bar (showing Entity, Product, and Year) and a local POV (the Plan drop-down list showing Q2):
With a global POV bar, if you change a page in the global POV bar and then click GO, the page changes for all objects that are based on forms. The global POV bar displays at the top of the dashboard above all the objects, while the local POV bar displays within the object. User variables are supported in both global and local POVs in dashboards.

In dashboard Settings, you can set whether to show or hide POVs and whether to enable or disable global POV bars. (If you select Hide for POVs, and Enable for Global POV Bar, the Hide option overrides the Enable option.) The global POV bar is enabled by default; if you disable it, the complete POV is displayed for each local POV as applicable.

About global POV bars:

- The POV Bar is made up of local POV dimensions, Page dimensions and user variables.
- Depending on the local POV and page of each form on the dashboard, the global POV bar is automatically calculated.
- They are reflected in the other objects using forms in that dashboard. That is, they apply to forms in a dashboard, to charts that are linked to forms, and to tiles that use forms as a data source. So if the dashboard doesn't include a form as a data source, then neither the local nor global POV bar is available.

Here's an example of how the global POV bar is calculated, based on the local POV dimensions for two forms:

The global POV bar is disabled:

- Form A local POV: Year, Entity, Product
- Form B local POV: Year, Entity, Project

The global POV bar is enabled:

- Global POV bar: Year, Entity
- Form A local POV: Product
- Form B local POV: Project

Because not all dimensions and page selections may be common to all forms on a dashboard, the complete POV for a dashboard object may get split between the local and global POV bar. The global POV bar and the local POV together contain the complete intersection information for each form on a dashboard.

If there is only one object on the dashboard that uses a form as a data source, then the entire POV/page of the form can be moved to the global POV bar.

If there is more than one object on a dashboard that use forms as a data source, then this is how the application determines which dimensions go in the global POV bar or stay in the local POV:

- If the dimension is in the POV or page of all the forms, and the member selection is the same in all the forms, the dimension goes in the global POV bar.
- If the dimension is in the POV on one form and in the page of another form, then the dimension stays in the local POV.
- If the dimension is in the POV, then the same members must be selected in all the forms for the dimension.
• If the dimension is a page dimension, then the selected page members must be the same and display in the same order in all the forms.

The POVs in dashboards honor valid intersections by hiding invalid Page members. See Dashboard POVs and Valid Intersections.

Dashboard POVs and Valid Intersections

The POVs in dashboards honor the valid intersections by hiding invalid Page members. Just like in forms, the Page drop-down list is filtered for all selected members in the POV and Page dimensions. Because dashboards support both global and local POVs, the context for filtering the Page drop-down list depends on which POV the members are located. If the Page drop-down list is on a global POV, the filtering context is only the global POV dimensions. If the Page drop-down list is on the local POV, the filtering context is all the global dimensions plus the dimensions on a chart's local POV.

See also About Global and Local POVs.
Designing Infolets

Related Topics
• About Infolets
• Anatomy of an Infolet
• Determining Infolet Content
• Using the Infolets Designer
• Creating Infolets
• Working with Infolets
• Customizing the Application Interface to Access Infolets

About Infolets

Infolets enable users to view and interact with high-level, essential information generated from different sources so that they can quickly assess where to direct their attention. Administrators create, redesign, delete, and assign permissions to infolets.

Watch this overview video to learn how to design infolets.

What is an Infolet?

An infolet is a self-contained, interactive box-shaped container used to display information using text and charts. Infolets are interactive and use progressive disclosure to display high-level, aggregated, essential information for quick consumption at a glance, and then can be acted upon as needed. Infolets can be flipped and resized to display up to three charts or sets of values.
For more information about infolets, see Anatomy of an Infolet.

**How Can I Use Infolets?**

Use infolets to:

- Promote essential, easily consumable information
- Provide answers to your most critical questions:
  - What is new or what has changed?
  - What is the most important information that supports my work?
- Group key information by user role in a way that helps users quickly assess and prioritize their work
- Progressively display essential details and actions
  Display these additional details in the various infolet views accessed by flipping or expanding an infolet. However, a single infolet view is acceptable.
- Provide a visually rich means of displaying essential or summary information

Do not use infolets to feature highly complex information, such as reporting functions, or to present detailed visuals.

See Determining Infolet Content.

**What is an Infolet Page?**

An infolets page is a page that contains one or more infolets. It houses a container that manages the space occupied by the infolets and rearranges them depending on the size of the browser and the size of the infolets. Each infolet you create belongs to an infolet page. The Infolets card on the Home page provides a list of infolet pages.
Anatomy of an Infolet

Infolet Views

An infolet supports up to three views:

1. Front view (required)

2. Back view (optional)

3. Expanded view (optional)
Chapter 10
Anatomy of an Infolet

The front view is required, and:

• Provides a quick look or glimpse at high-level information that has a direct effect on your work; for example, the front view can display status, counts, totals, or the most recent updates
• Promotes a glancing action that helps you identify important information that you may want to explore a bit more
• Uses all infolet sizes except 3x2 (see information about infolet sizes below)
• Returns the expanded view to its original size in the front view or back view
• Includes an Actions menu icon available only on hover and either a (optional) flip to back icon or an expand icon in the lower-right corner

Note:
If only one view is presented, it must be a front view.

The back view is optional, and:

• Presents analytical information (for example, a graph)
• Promotes a scanning action that helps you explore or become more familiar with the information represented on the front view
• Is sized the same as the front view
• Includes an Actions menu icon available only on hover, a flip to front icon in the lower-left corner, and an (optional) expand icon in the lower-right corner

The expanded view is optional, and:

• Presents even more detailed information about the single data point or interdependent data set presented in the front and back views; for example, the expand-
ed view can display more details about an object or a list of recent items than what's displayed on either the front or back view

• Provides enough information to help you decide if you're ready to take action and move to a focused context on a work area page
• Transitions smoothly from other views. You see a smooth, seamless expansion, as one infolet pushes others into new positions.
• Must be sized bigger than the front or back views
• Includes an Actions menu icon available only on hover, and a collapse icon in the lower-right corner

Infolet views honor the access permissions assigned to the underlying forms and dimensions. Therefore, the same infolet may display varying views from user to user if they have different access permissions.

Infolet Sizes

Infolets can be sized as follows:

**Note:**

1x1 refers to a box that spans one column and row width (170 pixels).

• 1x1
• 2x1
• 3x1
• 2x2
• 3x2 (expanded view only)

The size of the front and the back views are always the same. Changing the size of the front view will automatically reset the size of the back view. Because the size of the expanded view must always be greater than the size of the front/back views, if the size of the front/back view of an infolet is enlarged, the expanded view automatically resets to a larger size than the front/back view.

**Note:**

Front and back views cannot use the 3x2 size. This size is applicable for the expanded view only.

The size, title, and subtitle of a view is set by the designer in the properties panel. See Using the Infolets Designer.

Navigating Between Infolet Views

An infolet can be created with one of the following view combinations:

1. Front view only
2. Front and back views
3. Front and expanded views
4. Front, back, and expanded views

Switching from one view to another is controlled by clicking a flip icon, an expand icon, or a collapse icon available in the bottom-right or bottom-left corner of the infolet. Hovering your cursor over the bottom corners reveals a flip, expand, or collapse icon.

Determining Infolet Content

Consider the following general tips when determining infolet content:

- Look to existing dashboards and work area pages.
  Dashboards and work area pages are excellent starting points because they display collections of information summaries.

- Apply the 10/90/90 principle.
  Seek out the most essential, easily consumable information that addresses frequently asked questions garnered from the top 10 percent of use cases. These use cases can come from across your enterprise—from business intelligence, social, transactional, external, and so on. Then focus this information to reveal what 90 percent of your users would benefit from viewing 90 percent of the time. You can apply this 10/90/90 percent principle to existing dashboard content, to existing work area page content, or generally, to percolate eligible information for an infolet.

- Restate your top use cases in the form of frequently asked business questions.
  Present the corresponding infolet content in such a way as to answer these business questions; for example, how many orders are in jeopardy, listed by status?

- Look for one point or a tightly related, interdependent set of points, instead of multiple points of information.
  The process of determining content for an infolet is similar to the process that is used to yield dashboard content—but to a deeper level of analysis. Look for information within a data point or data set that is suitable to be displayed in no more than three views of information hierarchy and that answers a critical business question.

- Start with the single most important point.
  An infolet displays aggregated information about a single data aspect or point of information in relation to an event that the user needs to know about or a task that the user needs to address.
  If a dashboard contains multiple aspects about one or more objects (for example, numeric totals and currency totals), start with the single most important point and add that as a simple overview (for example, as a total using a stylized numeric value) to the front view of an infolet. Then determine the content for the back view, if needed. Finally, determine the content for the expanded view, if needed.
  An infolet should have no more than three views. If there is only a single data point or only one tightly related, interdependent data set to display on an infolet, use only the front view.

Related Links
Designing Forms for Infolets
Designing Charts for Infolets

Use the title and subtitle in charts to show static context.

There are six types of charts that you can use in infolets:

- **Bar**—Displays a graphical summary of multiple data values for comparison purposes. Bar charts can be plotted vertically or horizontally. Up to eight bars are recommended for bar chart infolets.
- **Column**—Displays stacked bars that represent different data sets on top of each other. The height of the resulting bar shows the combined result of the data sets.
- **Doughnut**—A circular graph which is divided into segments to compare data sets to one another. The blank center displays the sum of all data sets. Up to six segment values are recommended for doughnut chart infolets.
- **Line**—Use to visualize a trend in data over intervals of time.
- **Pie**—A circular graph which is divided into slices to compare data sets to one another. Up to six slices are recommended for pie chart infolets.
- **Tile**—Lets you select specific values from a data set to display. No more than three values are recommended for tile chart infolets.

**Note:**

Tile charts can only use the 1x1 size. You cannot resize an infolet using the tile chart until the chart type is changed. If you drag and drop a tile chart to an infolet that is greater than 1x1, you will be prompted to change either the size of the infolet or the chart type.
Using the Infolets Designer

Administrators use the Infolets Designer to create infolets and infolet pages. The Infolets Designer enables easy toggling between runtime and designer views. To access the Infolets Designer, launch the infolets icon on the Home page and click Create, or click the Actions icon next to an infolets page in the listing, and then click Edit. Clicking the infolet name in the listing, launches the runtime version of the infolet page. You can toggle from runtime view to designer view by clicking .

Infolets Designer

On the upper right is the Infolets Toolbar.

— Adds a new infolet to the Infolets Designer

— Hides and unhides the Properties panel

— Click to perform these actions:
  - **Reset**—Resets the Infolets Designer to a previously saved state
  - **Refresh**—Refreshes the data from Essbase and updates the infolet definition from the database
  - **Runtime**—Hides all Infolets Designer elements and displays the infolet as it would appear to users during runtime
—From runtime mode, displays the Infolets Designer

**Infolets Designer Palette**

On the left is the Designer Palette. The Designer Palette has two tabs: Forms and Chart Types. Highlight objects in the Designer Palette and then drag and drop them onto the drop zone.

**Designer Palette objects:**

- **Forms**—Select simple forms to include in the infolet by scrolling through the forms or by searching for them by name. For details about forms, see [Designing Forms for Infolets](#).

  - Note: The access permissions set for forms are honored in infolets.

- **Chart Types**—Select the chart types to include in the infolet. Infolets display sample data in the charts until you associate the chart with a form as its data source. When you link a chart to a form, users can see the impact of changing data in the
form on the associated charts. To associate a chart with a form, highlight a chart and drag and drop it onto the drop zone, in the Properties panel, click Sample, and then click Forms to select the data source. For details about charts, see Designing Charts for Infolets.

Infolet Menu

The infolet menu contains the infolet delete and clear actions. To view the infolet menu, hover over the upper right corner of the infolet, and then click the down arrow to display the menu options:

- **Delete**—Removes the infolet from the page
- **Clear**—Clears infolet details

Properties Panel

The properties panel on the right side of the Infolets Designer enables you view and work with these infolet properties:

- **Header**
- **Subtitle**
- **Size**—Displays the infolet in the selected size
- **Chart Type**—Displays the infolet data as the selected chart type
- **Data**—Displays the associated data source (Sample or Form)
- **Form**—Displays the selected infolet form

Actions such as delete and clear are on the infolet menu.

By default, the front view of an infolet is displayed in the properties panel. You can see the other views by selecting them from the drop-down. If you flip or expand an infolet to view the back or expanded views, the properties for those views display in the properties panel. In addition, the properties for the corresponding chart type are also displayed in the properties panel.

Creating Infolets

To create infolets:

1. From the Home page, click Infolets, and then click Create.
2. Click Infolets Page Name, and enter a title for the new infolets page you are creating.
3. From the designer palette on the left, choose either the Forms tab or the Charts Type tab, highlight an object, and then drag and drop it onto the infolets drop zone.

4. Customize the infolet using selections made in the properties panel, and then click Save.

You can easily duplicate and modify an infolet using the Copy As action on the Infolet list page. From the list page, click the Actions icon next to the infolet you want to copy, and then click Copy As.

Working with Infolets

Once you’ve created an infolets page, it is displayed in the infolets listing on the Infolets card.

The listing page for infolets supports folders. Folders enable you to assign permissions to all infolets within a folder rather than assigning permissions to each individual infolet. The infolets listing page uses the same folder hierarchy as dashboards and data entry forms and all artifacts and folders reside under a root folder called Library.

To view and work with infolets:

1. From the Home page, click Infolets.
2. To work with infolets, perform an action:
   - On the listing page for infolets, you can toggle between viewing infolets by a flat view or a tree view:

Then you can search for infolets using Search. The flat view displays only the artifacts that meet the search criteria, not the folders that contain them. The tree (or hierarchical) view displays artifacts in the context of the folders that contain them.

To search on another keyword, clear the search criteria by clicking X in the Search box.

- To refresh the infolets listing, click Refresh.
- To create infolets, click Create. See Creating Infolets.
- To perform the following actions on infolet pages, click the Actions icon next to the infolet page, and then select:
  - Create Folder—Creates a folder in the listing
  - Edit—Opens the infolet page in the Infolets Designer
  - Rename—Renames the infolet page
  - Copy As—Duplicates an infolet page
  - Delete—Deletes the infolet page from the listing
  - Move To—Moves an infolet page to another folder
  - Default or Unmark—Default marks an infolet page as default and makes it accessible directly from the Home page by clicking the second infolet dot that is displayed beneath the global header on the Home page. Unmark removes the default designation from the infolet page.
### Note:

You can mark either an infolet page or a dashboard as default. If a dashboard is marked as default first and then you mark an infolet page as default, the default dashboard will be overwritten. Conversely, if an infolet is marked default first, then any dashboard marked default later will overwrite the default infolet.

- **Assign Permission**—Enables you to assign Read, Write, and None access permissions to infolet pages and folders for individual users or groups.

### Customizing the Application Interface to Access Infolets

You can customize the application interface to add links to infolet pages from the Home page using the Navigation Flow Designer. When you customize your application interface to access infolet pages, dots will appear on the Home page beneath the global header. Each dot that appears on the Home page represents an infolet page and hovering over each dot displays the name of the infolet page. Clicking an infolet dot launches the infolet page associated with that dot. You can define up to seven infolet dots on the Home page. If you've created connections to other EPM Cloud subscriptions, you can also add links to infolet pages in other EPM Cloud subscriptions.

Users will only see dots displayed on the Home page for infolet pages to which they have access. The types of infolet dots displayed are as follows:

- **Home dot**—This dot always appears first and it links to the Home page. There can only be one home dot. If you are not viewing the Home page, clicking the home dot will bring you back to the Home page.

- **User dot**—Links to an infolet page marked by an end user as the default infolet page. There can only be one user dot and it always appears after the Home dot on the user's Home page. User dots cannot be added using the Navigation Flow Designer. For more information about marking an infolet as the default, see Working with Infolets.

- **Customizable dot**—Links to infolet pages created by administrators. Customizable dots can be integrated into navigation flows and their visibility and the order in which they appear is determined by the navigation flow designer. There can be up to seven customizable dots and they always appear after the home and user dots.

To add infolet dots to your application interface using the Navigation Flow Designer:

1. From the Home page, click **Tools**, and then click **Navigation Flows**.
2. Select an inactive navigation flow in the listing, and then click the **Infolets** tab on the Navigation Flow Designer.
3. Click **+**.
4. In Manage Infolet, name the infolet dot, set visibility, and then click ![search_icon] to select an infolet in the Artifact Library.

**Note:**

You can select an infolet from another EPM Cloud subscription if you've created connections to other subscriptions. First select the subscription under **My Connections**, and then navigate to the infolet in that subscription.

5. Click **Save and Close**.

**Note:**

Infolets can also be associated with a tab or a card in a navigation flow. While adding or updating a tab or a card, select an infolet in the Artifact Library.

To view design time changes to the navigation flow, activate the navigation flow, and then from the Home page, click the down arrow next to the user name (upper right corner of the screen) and click **Reload Navigation Flow**.

To learn more about designing navigation flows, see "Designing Custom Navigation Flows" in your Administering guide.

To learn more about connecting EPM Cloud subscriptions, see "Connecting Subscriptions in EPM Cloud" in your Administering guide.
Managing Sandboxes

Enable planners to use sandboxes to keep their work private while experimenting with various outcomes as they develop their plans.

Related Topics

- About Sandboxes
- Enabling Sandboxes
- Enabling Version Members
- How Sandboxes Work with HSP_View and Version Members
- What Happens When a Sandbox is Published?
- Managing Sandboxes
- Sandboxes and Other Functionality

About Sandboxes

Sandboxes enable planners to keep their work private (from other non-administrators) as they develop their plans (see "Building a Plan Privately" in the Working with Planning). Planners may want to privately experiment with various outcomes in a sandbox without having others see their numbers until they've completed their analysis. When they're ready to make their numbers public (to others having access to them), they publish the data in the sandbox.

Some basics about sandboxes:

- Sandboxes are enabled at the cube level. You can select the Sandboxes option for block storage cubes that are seeded when you create the application. You can also enable sandboxes when you create a new block storage cube. You can't later change the sandbox option for existing cubes, including cubes created by default such as Plan1. See Enabling Sandboxes.

- By default, forms and ad hoc grids are in base view. When a planner starts working in a sandbox, the form switches from base view to sandbox view. In sandbox view, the planner's data is saved independently of the application's data. When planners are done privately modifying and analyzing data in the sandbox, they publish the sandbox. When published, the new and changed data is saved to the application. The form is then back in base view. See What Happens When a Sandbox is Published?.

- To provide access to sandboxes for users with the Planner role, you must enable the Apply Security option for the Version dimension. See also Enabling Sandboxes and Enabling Version Members.

- All access permissions are in effect in sandboxes, including security on approval units. For example, if the Version member is part of an approval unit that is in the approvals process, planners have write access only when they are the current owner of the approval unit.
• Planners can have multiple sandboxes and switch among them.

**Note:**
If a planner switches to a sandbox from a composite form, all the forms that comprise the composite form also switch to sandbox view.

• Multiple planners can simultaneously work in multiple sandboxes without affecting each other's work.

• Administrators can see the names of (and data in) all sandboxes, which version member they're based on, who created each one, when they were last modified. Administrators can also delete anyone's sandbox. To manage sandboxes, from the Home page, click **Application**, then click **Overview**, then **Actions**, and then **Manage Sandboxes**.

• Because dynamic members are automatically calculated, planners can immediately see the effect of their changes.

• When a cube is enabled for sandboxes, a dimension called HSP_View is added, with three members: **BaseData**, **SandboxData**, and **ConsolidatedData**. See **How Sandboxes Work with HSP_View and Version Members**.

• When working in a sandbox, planners can add members, but those members aren't added to the base version when the sandbox is published.

## Enabling Sandboxes

To enable sandboxes:

1. When you create an application, a block storage cube is created, which you can enable for sandboxes by selecting the **Sandboxes** option. You can also enable sandboxes when you create custom block storage cubes by selecting **Enable Sandboxes**.

   You can enable sandboxes only in Standard applications and for custom cubes created in Enterprise applications, not for Reporting applications. After you enable sandboxes for a cube, you can't later disable that option. If you enable sandboxes for a cube, Version members have the **Enable Sandboxes** option.

2. Create and enable Version members for sandboxes, as described in **Enabling Version Members**.

3. If needed, modify member formulas so that they calculate correctly in both base view and sandbox view.

   Member formulas may be impacted by the new HSP_View dimension. To ensure that member formulas use the correct data for all versions, they must reference “ConsolidatedData”/Sandbox.

   See:
   - The topics in **How Sandboxes Work with HSP_View and Version Members**
   - **Working with Member Formulas**
In Data Maps, all data mapping occurs at the HSP_View BaseData intersection. The base view is selected by default.

4. Design or modify forms that you want to support sandboxes so that either:
   • The Version dimension is on the Page axis.
   • The Version dimension is a POV that has a user variable.

Forms that don't meet one of the above conditions do not display sandbox options.

In a sandbox, the HSP_View member is fixed on the POV and can't be changed.

The only business rules that planners can run in a sandbox are the default Calculate Form and Calculate Currencies rules.

You can't import or export when you're in the sandbox view.

### Enabling Version Members

To make versions available to sandboxes, administrators create Version members and then select the member property **Enable Sandboxes**.

When planners switch to sandbox view, they select an enabled Version member, and then the data in the sandbox is tied to that version. When the planner switches back to base view, the base version data is displayed. If a planner publishes their sandbox data, the modified data in the sandbox is saved to the application and displays in the base view. See How Sandboxes Work with HSP_View and Version Members and What Happens When a Sandbox is Published?.

**Tip:**

To edit member properties, from the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.

Note that the **Dimensions** link is available only if you're accessing the application from the desktop.
How Sandboxes Work with HSP_View and Version Members

Related Topics
- The HSP_View Dimension
- Sample Formula for the ConsolidatedData Member
- Sandbox Version Members
- Sandbox Implications on Member Formulas

The HSP_View Dimension

When a cube is enabled for sandboxes, the HSP_View dimension is created with the following members.

- The **BaseData** member defines the data intersection when users are working in the base view in a form. When planners are not working in a sandbox, data is stored at the intersection of the BaseData member.
- The **SandboxData** member stores the data when a user works with data in a sandbox.
- The **ConsolidatedData** member is dynamically calculated; it retrieves data from the SandboxData member if it’s available. Otherwise, the ConsolidatedData member retrieves data from the BaseData member of the base version.

**Note:**
- You can't change the HSP_View dimension or its members. You should not change the order of the HSP_View dimension in **Performance Settings**. To view **Performance Settings**, from the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
- In a form, the HSP_View member is fixed on the POV and can't be changed.
- Aggregations at the top levels of sparse dimensions are not supported in Sandbox view.

Sample Formula for the ConsolidatedData Member

Assuming that Market is a sparse dimension and that Working is a base version, here is a sample formula for the ConsolidatedData member:

```
IF(@ISLev("Market",0))
IF(@ISLev("Entity",0))
IF (@ISLev("Period", 0))
IF(NOT (@ismbr(@relative("Sandboxes",0))))
BaseData;
```
ELSE
  IF (SandboxData == #MISSING )
  IF (@ISCHILD("Sandboxes_Working"))
    "Working"->BaseData;
  ENDIF
  ELSE
    SandboxData;
  ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF

Sandbox Version Members

When an application is enabled for sandboxes, a Sandbox member is created under the Version dimension. When a sandbox is created, a Version member is added under the Version Sandbox member, with the name given by the creator of the sandbox. Example:

```
Version
  ➤ SandBox
    ➤ SandBox 1
    ➤ SandBox 2
    ➤ SandBox 3 ...
```

- The data in the base form is stored at the intersection of the HSP_View BaseData member and the base version (for example, Working).
- The data in the sandbox is stored at the intersection of the respective member from the Version dimension (for example, Sandbox 1) and the HSP_View member SandboxData.

Sandbox Implications on Member Formulas

When you create or modify member formulas, for data to be calculated correctly in both the base and sandbox views, member formulas must refer to the intersection of the HSP_View ConsolidatedData member and the sandbox Version member. Example:
What Happens When a Sandbox is Published?

When a user publishes the data in a sandbox:

- The modified data in the sandbox is saved to the application only for those inter-
  sections that are visible in the base view at the time the data is published and only
  for those cells to which the planner has write permission. Sandbox data is saved to
  the base view in all affected forms and cubes. Even if the planner has modified da-
  ta in multiple forms or different page/POV combinations, all modified data, support-
  ing detail, and comments are moved to the application.

- The application deletes the sandbox. (Administrators can delete sandboxes before
  the data in them is published. See Managing Sandboxes.)

- If the publish action takes longer than the three minute default, then the job runs in
  the background and is displayed in the Job.

**Note:**

If multiple planners publish sandbox data to the same member in the base
view, the most recently published data overwrites previously published data.

Managing Sandboxes

To manage sandboxes:

1. Click Application, then click Overview, then Actions, and then Manage Sand-
   boxes.
2. To filter the list of sandboxes, click , and then complete any of these fields:
   - Sandbox Name
   - Base Version
   - Created By
3. To delete sandboxes, click their names, and then **Delete**.

**Sandbox and Other Functionality**

- **Oracle Smart View for Office**:
  - Sandbox functionality is available in Smart View, though you can't create, delete, or publish data from Smart View.
  - In ad hoc analysis, users can see the HSP_View members, which includes sandbox and base data.
  - Selecting the ConsolidatedData member displays the data entered in both the sandbox and the original base data.
  - Sandboxes are supported for Smart Slices, but the HSP_View dimension is hidden.
  - In Planning Admin Extensions, the HSP_View dimension is displayed under **Dimensions**, but you can't change member properties. Likewise, you can't change the member properties of sandbox Version members.

- **Financial Reporting**: To report on base view data, select the BaseData member of the HSP_View dimension. To report on sandbox data, select the ConsolidatedData member for HSP_View.

- **Data Management**: The HSP_View dimension is visible in the mappings, but you can map only the BaseData member.

- **Migration**: If you clone an application using Migration, Oracle recommends that you select the **Sandbox Changes** option. If you're importing sandboxes and data, you must select **Sandbox Changes** if you want to publish the sandbox data.

- **Oracle Hyperion Public Sector Planning and Budgeting**: Decision Packages and Budget Requests don't support sandboxes.
Defining Valid Intersections

Define valid intersection rules which filter certain cell intersections to users when they enter data or select runtime prompts.

Related Topics
- Understanding Valid Intersections
- Creating Valid Intersections
- Managing Valid Intersections
- Suppressing Invalid Data in Forms
- Clearing Invalid Data
- Working with Valid Intersections

Understanding Valid Intersections

Valid intersections enable you to define rules, called valid intersection rules, which filter certain cell intersections to users when they enter data or select runtime prompts. For example, you can specify that certain programs are valid only for some periods or departments.

After valid intersections are defined, cells containing invalid data are read-only. This restriction speeds the planning process and optimizes the information available to users.

To better understand how valid intersections affect behavior in forms and in runtime prompts, see Working with Valid Intersections.

Watch this tutorial video to learn how manage valid intersections.

Tutorial Video
- Valid Intersection Groups
- Valid Intersection Rules
- Anchor and Nonanchor Dimensions
- Valid Intersection Examples
- Redundancy or Overlap in Valid Intersection Rules
- Shared Members and Valid Intersection Rules
- Substitution Variables and Valid Intersection Rules
- Evaluation Order
Valid Intersection Groups

Valid intersection groups define:

- Dimensions to be included
- One of those dimensions as the anchor dimension
- Whether nonanchor dimensions are required or not
- Whether the anchor dimension members not specified or referenced will be valid or invalid

Valid Intersection Rules

Valid intersection rules:

- Must use the same dimensions that were defined within their valid intersection group
- Define only valid intersections
- Valid intersection rules within the same valid intersection group that produce an apparent conflict or overlap, are marked valid if either valid intersection rule condition is met
- Valid intersection rules in different valid intersection groups that produce an apparent redundancy or overlap, are marked valid if they satisfy the requirements of all valid intersection groups

Thus, if any valid intersection group marks an intersection invalid, regardless of other valid intersection groups making it valid, the system will mark the intersection invalid. Invalid groups override valid group results.

Note:

If you want to remove valid intersections regardless of what other valid intersection groups allow, then this rule must be in a different valid intersection group.

See Valid Intersection Examples.

Anchor and Nonanchor Dimensions

Anchor and nonanchor dimensions:

- Anchor dimensions are always required dimensions in the cube that is used in the valid intersection evaluation.
- Nonanchor dimensions are either required or not:
  - If a nonanchor dimension is required, any cube that doesn't use that dimension will ignore any valid intersection group where that dimension is tagged as required as it evaluates the valid intersections.
  - If a nonanchor dimension isn't required, any cube that doesn't use that dimension will still evaluate any valid intersection group that includes that dimension.
as not required and evaluate the intersections of any other dimensions in the valid intersection group in use in the cube.

- Unselected anchor dimension members are valid by default, but you can mark them invalid by clearing the **Unselected Members are Valid** option. This option marks all intersections with anchor dimension members not selected in this rule as invalid.

See [Valid Intersection Examples](#).

### Valid Intersection Examples

This section provides valid intersection group and valid intersection rule examples to illustrate a few simple, complex, and edge-case scenarios.

#### Example: Anchor and Nonanchor Dimensions

The choice of the anchor dimension is critical. Consider the following example, which produces a different result based on the anchor dimension definition:

- Valid intersection group 1 defines Entity as the anchor dimension and Product as a nonanchor dimension.
- Valid intersection group 2 reverses this definition with Product as the anchor dimension and Entity as the nonanchor dimension.

<table>
<thead>
<tr>
<th>Table 12-1</th>
<th>Example - Anchor Dimension is Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Valid Intersection Group</strong></td>
<td><strong>Anchor Dimension - Entity</strong></td>
</tr>
<tr>
<td>1</td>
<td>DESC(500 - Manufacturing) - Unselected members are valid</td>
</tr>
</tbody>
</table>

Group 1 means entities that are descendants of Manufacturing are valid only with descendant products of Computer Equipment. No other products are valid with descendants of Manufacturing. All other entities besides descendants of Manufacturing are valid with all products, including descendants of Computer Equipment.

<table>
<thead>
<tr>
<th>Table 12-2</th>
<th>Example - Anchor Dimension is Product</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Valid Intersection Group</strong></td>
<td><strong>Anchor Dimension - Product</strong></td>
</tr>
<tr>
<td>2</td>
<td>DESC(P_TP1 - Computer Equipment) - Unselected members are valid</td>
</tr>
</tbody>
</table>

Group 2 means products that are descendants of Computer Equipment are only valid with descendant entities of Manufacturing. No other entities are valid with descendants of Computer Equipment. All other products besides descendants of Computer Equipment are valid with all entities, including descendants of Manufacturing.
Caution:

The choice of anchor dimension is significant. You'll get dramatically different results if you choose the wrong anchor dimension.

Example: Required Dimension

In the following example, if a nonanchor dimension isn't required, then the application evaluates all remaining dimension intersections in the valid intersection group for a cube that doesn't contain the nonrequired dimension. This behavior could result in the evaluation of a valid intersection group with only one effective dimension.

<table>
<thead>
<tr>
<th>Valid Intersection Group</th>
<th>Anchor Dimension - Entity</th>
<th>Nonanchor Dimension - Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DESC(500 - Manufacturing) - Unselected members are valid</td>
<td>DESC(P_TP1 - Computer Equipment) - Not required</td>
</tr>
</tbody>
</table>

In Group 1, the product dimension isn't required, and unselected entities are valid. Therefore, if the cube of the form or business rule, at runtime, doesn't include the product dimension, the application evaluates the entity dimension selections to mark all entities as valid for a cube that doesn't contain the product dimension.

<table>
<thead>
<tr>
<th>Valid Intersection Group</th>
<th>Anchor Dimension - Entity</th>
<th>Nonanchor Dimension - Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>DESC(500 - Manufacturing) - Unselected members are invalid</td>
<td>DESC(P_TP1 - Computer Equipment) - Not required</td>
</tr>
</tbody>
</table>

In Group 2, the product dimension isn't required, and unselected entities are invalid. Therefore, if a cube doesn't include the product dimension, then the application evaluates the entity dimension selections to mark all entities except descendants of Manufacturing as invalid. Thereafter, any cube that doesn't use the product dimension will only allow data entry in the descendants of Manufacturing entities.

Caution:

Carefully consider whether a nonanchor dimension is required or not, especially if the result leaves a valid intersection group with only one effective dimension. Additionally, selecting the Unselected Members are Valid option for anchor dimension members also plays a significant role in the system behavior for valid intersections. See Valid Intersection Examples.
Example: Unselected Members are Valid

In the following example, two intersection groups are valid. In one group, the anchor dimension unselected members are invalid (this option is cleared). In the other group, the anchor dimension unselected members are valid (this option is selected).

Table 12-5  Example - Unselected Members are Valid

<table>
<thead>
<tr>
<th>Valid Intersection Group</th>
<th>Anchor Dimension - Account</th>
<th>Nonanchor Dimension - Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IDESC(BS - Balance Sheet) - Unselected members are invalid</td>
<td>000 - No Department</td>
</tr>
<tr>
<td>2</td>
<td>IDESC(GP - Gross Profit) - Unselected members are valid</td>
<td>IDESC(403 - Sales)</td>
</tr>
</tbody>
</table>

Because Group 1 defines all unselected members as invalid, the application marks noninclusive descendants of Balance Sheet invalid. Gross Profit isn't an inclusive descendant of Balance Sheet. So even though Group 2 explicitly states inclusive descendants of Gross Profit are valid with inclusive descendants Sales entities, the invalid definition from Group 1 overrides any further valid intersections of the same anchor dimension member set.

Example: Redundant or Overlapping Valid Intersection Rules Within the Same Valid Intersection Group

When valid intersection rules are within the same valid intersection group and produce any redundancy or overlap, the system marks an intersection valid if either of the valid intersection rule conditions are met.

Table 12-6  Example - Redundant or Overlapping Valid Intersection Rules Within the Same Valid Intersection Group

<table>
<thead>
<tr>
<th>Valid Intersection Rule</th>
<th>Anchor Dimension - Account</th>
<th>Nonanchor Dimension - Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IDESC(GP - Gross Profit) - Unselected members are valid</td>
<td>IDESC(403 - Sales)</td>
</tr>
<tr>
<td>2</td>
<td>IDESC(NI - Net Income) - Unselected members are valid</td>
<td>IDESC(TD - Total Department)</td>
</tr>
</tbody>
</table>

Because Gross Profit is a descendant of Net Income and Sales is a descendant of Total Department, inclusive descendants of Gross Profit are valid with any inclusive descendant of Total Department. Rule 1 is a subset of Rule 2, so Rule 1 is effectively a "No operation" rule and is unnecessary. There is no restriction on inclusive descendants of Gross Profit accounts only being valid for inclusive descendants of Sales Entities.

Example: Redundant or Overlapping Valid Intersection Rules in Different Valid Intersection Groups

When valid intersection rules are in different valid intersection groups and produce any redundancy or overlap, the system marks an intersection valid only if it satisfies the requirements of all valid intersection groups.
In the following example, there are redundant or overlapping rules in different groups:

**Table 12-7  Example - Redundant or Overlapping Valid Intersection Rules in Different Valid Intersection Groups**

<table>
<thead>
<tr>
<th>Valid Intersection Group</th>
<th>Anchor Dimension - Account</th>
<th>Nonanchor Dimension - Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IDESC(GP - Gross Profit) - Unselected members are valid</td>
<td>IDESC(403 - Sales) - Required</td>
</tr>
<tr>
<td>2</td>
<td>IDESC(NI - Net Income) - Unselected members are valid</td>
<td>IDESC(TD - Total Department) - Not required</td>
</tr>
</tbody>
</table>

Because Group 1 is further restrictive for inclusive descendants of Gross Profit accounts being valid with inclusive descendants of Sales entities, the application enforces this group for these intersections. Other, non-Gross Profit accounts can still use all inclusive descendants of Total Department entities, but inclusive descendants of Gross Profit accounts must use inclusive descendants of Sales entities.

**Redundancy or Overlap in Valid Intersection Rules**

Valid intersection rules within the same valid intersection group, which produce any apparent conflict or overlap, are marked valid if either valid intersection rule condition is met.

If different valid intersection groups share the same attributes, including the anchor dimension, required and not required nonanchor dimensions, and **Unselected Members are Valid** attribute, they will be treated as rules of the same valid intersection group.

**Shared Members and Valid Intersection Rules**

Shared members are supported in valid intersection rules. If a base member is selected for a valid intersection rule, any shared members are also included in the rule. Conversely, if a shared member is selected for a valid intersection rule, the base member is also included in the rule.

**Substitution Variables and Valid Intersection Rules**

You can use substitution variables in valid intersection rules. User variables are not supported. Substitution variables can be set on the Essbase server, application, or database level. The same substitution variable can exist on multiple levels; the application uses the first one it finds as it searches in this order:

1. Database (cube)
2. Application
3. Server

**Evaluation Order**

Evaluation order for valid intersection groups orders invalid results sets as quickly as possible, increasing the speed and efficiency of the overall valid intersection evaluation.
For example, the application evaluates the first valid intersection group in the list, then the second group, and so on. If the application finds an invalid intersection in the second group in the list, it will stop evaluating the rest of the list because, once an intersection is defined as invalid, it will override other valid intersection rule results.

To change the order in which the groups are evaluated, see Changing the Valid Intersection Group Evaluation Order.

Creating Valid Intersections

To create a valid intersection:

1. Click Application, and then click Valid Intersections.
2. If it isn't already selected, click Setup.
3. Create the valid intersection group:
   a. Click Create.
   b. Enter a name and description for the intersection.
   c. To select the anchor dimension, click next to Select Anchor Dimension.
      
      Optional: By default, the anchor dimension members that are not specified in the valid intersection rule are marked valid. To clear this option, click next to the anchor dimension, and then click Unselected members are valid.
      
      For an explanation of this option and for an example of how it's used, see Anchor and Nonanchor Dimensions.
   d. To select additional dimensions (called nonanchor dimensions), click Add Dimension.
      
      Optional: By default, nonanchor dimensions are not required. To make a nonanchor dimension required, click next to the nonanchor dimension, and click Required.
4. Define the valid intersection rule:
   a. Click Add Rule.
   b. To select the range of members to include, exclude, or remove in the valid intersection, click next to the dimensions in the new rule:
      
      • Click Edit to open the Select Members page and select the members, substitution variables, and attributes to include in the valid intersection rule. See Using the Member Selector.
      
      • Click Add Exclusion to define an exclusion in the rule. You can exclude a subset of what is included for that dimension.
      
      • Click Clear to clear the selection.
   c. To delete a rule, click .
5. Click Save and Close.

The new valid intersection group is added to the end of the valid intersections list. To reorder the rules in the list, see Changing the Valid Intersection Group Evaluation Order.
Managing Valid Intersections

Related Topics

• Viewing Valid Intersections
• Changing the Valid Intersection Group Evaluation Order
• Disabling and Enabling Valid Intersection Groups
• Editing Details for a Valid Intersection Group
• Duplicating Valid Intersection Groups
• Deleting a Valid Intersection Group

Viewing Valid Intersections

To view valid intersections:

1. Click Application, and then click Valid Intersections.
2. If it isn't already selected, click Setup.
3. Perform a task:
   • Create a valid intersection group. See Creating Valid Intersections.
   • Reorder valid intersection groups. See Changing the Valid Intersection Group Evaluation Order.
   • Disable and enable valid intersection groups. See Disabling and Enabling Valid Intersection Groups.
   • Edit details for a valid intersection such as adding or removing dimensions in a valid intersection group. See Editing Details for a Valid Intersection Group.
   • Duplicate an existing valid intersection group so you can quickly create a new one. See Duplicating Valid Intersection Groups.
   • Delete valid intersection groups. See Deleting a Valid Intersection Group.

Changing the Valid Intersection Group Evaluation Order

Evaluation order for valid intersection groups orders invalid results sets as quickly as possible, increasing the speed and efficiency of the overall valid intersection evaluation.

To learn more about evaluation order, see Evaluation Order.

To change the position of a valid intersection group in a list:

1. Click Application, and then click Valid Intersections.
2. If it isn't already selected, click Setup.
3. To the right of the valid intersection, click 
4. Select Move Up or Move Down.
Disabling and Enabling Valid Intersection Groups

Valid intersection groups, by default, are enabled at the time of creation. If you don't want a valid intersection group to be evaluated or used, you can disable it on the **Valid Intersections** page. When a valid intersection is disabled, the valid intersection rule for that group no longer applies when viewing application forms, business rules, or runtime prompts. You can also reenable a disabled valid intersection group.

To disable and enable a valid intersection group:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click **Setup**.
3. In the Enabled column of the valid intersection list, click the check mark next to the valid intersection group that you're disabling or enabling.

**Note:**
The check mark is green if the group is enabled.

4. Ensure that any remaining groups that are enabled are still listed in the correct evaluation order in the valid intersections list. If they are not, then move them up or down in the order.

See **Changing the Valid Intersection Group Evaluation Order**.

Editing Details for a Valid Intersection Group

To edit valid intersection group details, you work with dimension members in the member selector. You can also define exclusions in valid intersection rules.

To edit valid intersection group details:

1. Click **Application**, and then click **Valid Intersections**.
2. If it isn't already selected, click **Setup**.
3. Click the name of the valid intersection group you want to edit.

   - To edit dimension details, next to the dimension, click a to select the members to include, exclude, or remove in the valid intersection rule:
     - Click **Edit** to open the **Select Members** page and select members, substitution variables, and attributes to include in the valid intersection rule. You can also type in the members or functions. See **Using the Member Selector**.
– Click **Add Exclusion** to define an exclusion in the rule. You can select members to exclude; for example, you can select or include all children of YearTotal except children of Q1 by excluding children of Q1.

– Click **Clear** to clear the selection.

• To delete a dimension from a valid intersection group, next to the dimension, click 
[ ![ ]](image) , and then click 
[ ![ ]](image) .

• To remove a rule from a valid intersection group, click 
[ ![ ]](image) .

• To add a dimension or a rule to a valid intersection group, click **Add Rule** or **Add Dimension**.

4. Click **Save and Close**.

**Duplicating Valid Intersection Groups**

To speed valid intersection group creation, you can duplicate an existing valid intersection and then edit it.

To duplicate a valid intersection group:

1. Click **Application**, and then click **Valid Intersections**.

2. If it isn't already selected, click **Setup**.

3. Click 
[ ![ ]](image) to the right of a valid intersection group you want to duplicate, and then select **Duplicate**.

The duplicate group is added to the end of the valid intersections list with the word “Copy” appended to the name.

4. Open the valid intersection group and edit it.

5. Reorder the valid intersection groups, if needed. See **Changing the Valid Intersection Group Evaluation Order**.

**Deleting a Valid Intersection Group**

After a group is deleted, the valid intersection groups are reordered. If there are three valid intersection groups, and the second one in the order is deleted, the third valid intersection group becomes number two.

To delete a valid intersection group:

1. Click **Application**, and then click **Valid Intersections**.

2. If it isn't already selected, click **Setup**.

3. Click 
[ ![ ]](image) to the right of a valid intersection group that you want to remove, and then select **Delete**.

4. Reorder the remaining valid intersections, if needed. See **Changing the Valid Intersection Group Evaluation Order**.

To delete a valid intersection rule from a valid intersection group, see **Editing Details for a Valid Intersection Group**.
Suppressing Invalid Data in Forms

Suppressing invalid data hides rows or columns in application forms that contain invalid data. If this option isn't selected, then the application displays rows or columns that contain cells with data that are invalid. Cells with invalid data are read-only.

To suppress invalid data in forms:

1. From the Home page, click Navigator and then under Create and Manage, click Forms.
2. Select the form, click , and then click Layout.
4. Click Save.

Clearing Invalid Data

Related Topics
- About Invalid Data
- Working With Invalid Intersection Reports
- Clearing Data at Invalid Intersections

About Invalid Data

If data already exists at intersections, then adding or modifying valid intersections invalidates the existing data. Creating a valid intersection rule or modifying an existing valid intersection rule doesn't clear data in the invalid intersections. You must generate a valid intersection rule report, which will show where data exists at invalid intersections, and then determine whether to clear the data.

Note:

Data may remain at an invalid intersection for historical purposes or for use in forward looking scenarios. Therefore, it's not a requirement to clear data at invalid intersections.

Working With Invalid Intersection Reports

You can view reports that show data at invalid intersections on the Reports tab of the Valid Intersections page. Invalid Intersection Reports lists existing reports, their status, and the last time they were run.

To work with invalid intersection reports:

1. Click Application, then Valid Intersections, and then click Reports.
2. Perform a task:
   • To refresh the listing, click **Refresh**.
   • To create a report, click **Create**, name the report, select the cube, choose the scope of the report, and then select when to run the report. You can run the report now or run the report later by saving it as a job. To save a report without running it, click **Save and Close**. To remove a report after it's run, click **Remove Reports**.

   **Note:**
   Invalid intersection reports are not supported for aggregate storage cubes. Aggregate storage cubes are not listed in the **Cube** drop-down list.

   • To edit a report, click **Edit**, and then **Edit**.
   • To copy a report, click **Duplicate**, and then **Duplicate**.
   • To run a report, click **Run**, and then **Run**.
   • To delete a report, click **Delete**, and then **Delete**.

### Clearing Data at Invalid Intersections

Users with appropriate permissions can clear the invalid data if the data isn't needed. To clear invalid data, run the report, and then click **Clear Invalid Intersections**.

### Working with Valid Intersections

**Related Topics**

- Working with Valid Intersections in Application Forms
- Working with Valid Intersections in Calculation Manager Rule Runtime Prompts

### Working with Valid Intersections in Application Forms

Using valid intersections prevents data entry for invalid intersections as defined in the applicable valid intersection group. The affected cells in the form display as read-only following standard, read-only color coding. If you hover the cursor over an invalid intersection, a tool tip displays indicating the cell is read-only because it's defined as an invalid intersection.

The valid intersection group applies first to the form point of view and page axis. If the point of view intersections are all invalid, then a warning message is displayed, and the form doesn't render a data grid until a valid intersection is selected.

If the point of view has valid intersections, then the rows and columns are filtered to restrict data entry at invalid intersections. If the **Suppress Invalid Data** option for the
form is enabled, then the form suppresses invalid rows, columns, or both, as appropriate.

Any rows or columns, which consist of a mix of valid and invalid intersections, display those intersections as valid or invalid, as appropriate. Invalid intersections are displayed with standard, read-only shading and preclude data entry.

Application users can reset the point of view to the default, unfiltered list without closing and reopening the form by clearing the selections. Application users can also clear a selection, thus opening up more selections for other dimensions. They will not be able to render a form with a dimension cleared, because valid members must be selected for each dimension.

In the member selector, invalid members are suppressed due to valid intersection rules. Application users can display invalid members in the member selector using the Show Invalid Members option. Invalid members are displayed but are unavailable for selection.

Note:
Valid intersection groups don’t grant access to dimension members. Valid intersection groups further restrict the valid intersections of dimension members already granted to an application user.

### Table 12-8  Form Behavior if Valid Intersections are Applied

<table>
<thead>
<tr>
<th>Action</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open a form</td>
<td>The form renders with member selections as defined in the form definition, adhering to the user’s access rights for dimensions, and applies valid intersection groups with the most recently used as current selections.</td>
</tr>
</tbody>
</table>
### Table 12-8  (Cont.) Form Behavior if Valid Intersections are Applied

<table>
<thead>
<tr>
<th>Action</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select members from a point of view dimen-</td>
<td>The application:</td>
</tr>
<tr>
<td>sion</td>
<td>• Enables users to select a member on the point of view</td>
</tr>
<tr>
<td></td>
<td>• In the member selector for a point of view dimension, enables users to select from a filtered list of remaining valid intersections, which is based on the members that were selected for the other point of view dimensions</td>
</tr>
<tr>
<td></td>
<td>• Ignores the order in which point of view dimension members are selected because selecting a member from any dimension included in a valid intersection group dynamically filters the remaining dimension members lists for those dimensions included in the valid intersection group, as appropriate, when that dimension is selected</td>
</tr>
<tr>
<td></td>
<td>• Provides the option to hide invalid members from dimension lists or display them as unselectable in the point of view</td>
</tr>
<tr>
<td></td>
<td>• Provides the ability to reset the point of view to the fully unfiltered list without closing and reopening the form by clearing the selections</td>
</tr>
</tbody>
</table>

**Note:**
Ad hoc forms, both in Web and Oracle Smart View for Office, will not filter page or point of view members according to valid intersection groups.

Select **Go** to render a form based on point of view selections. You can also click the right arrow in the form point of view.

Enter and save data

The form data is entered and saved.
Valid intersection groups apply to runtime prompts when launched from within the context of the application. Runtime prompts will prevent users from selecting invalid intersections as defined in the valid intersection groups.

Filtering according to valid intersection groups isn't supported in Oracle Smart View for Office forms. The rule, however, will not launch if you choose an invalid intersection in the runtime prompts both in the Web and in Smart View.

**Table 12-9  Runtime Prompt Behavior if Valid Intersections are Applied**

<table>
<thead>
<tr>
<th>Action</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Launch a Calculation Manager rule runtime prompt</td>
<td>The application:</td>
</tr>
<tr>
<td></td>
<td>• Prevents the user from selecting invalid intersections within the runtime prompt based on the valid intersection group</td>
</tr>
<tr>
<td></td>
<td>• Prevents the Calculation Manager rule from executing if there are invalid intersections in the runtime prompts</td>
</tr>
<tr>
<td>Enter valid intersections</td>
<td>The valid intersection is allowed to be entered.</td>
</tr>
</tbody>
</table>
Defining Data Maps

Create data mappings between a source application and a target reporting application.

Related Topics

- Understanding Data Maps
  Use data maps to move data, comments, attachments, and supporting detail from source cubes and smart lists to reporting cubes to consolidate data.
- Creating Data Maps
- Refreshing Data Maps
- Managing Data Maps
- Moving Data

Understanding Data Maps

Use data maps to move data, comments, attachments, and supporting detail from source cubes and smart lists to reporting cubes to consolidate data.

You can map dimensions between a source application database and a reporting database to enable:

- Reporting on source application data in a reporting application
- Aggregating and querying on Smart Lists, which are converted to regular dimensions in the reporting application
- Linking application data to multiple reporting applications for various consolidations
- Pushing data between Strategic Modeling and Planning (see Pushing Data Between Strategic Modeling and Planning Using Data Maps in Administering Planning Modules)

Watch this tutorial video to learn how to move data for reporting using data maps.

Tutorial Video

Subtopics:

- About Applications
- About Reporting Cubes
- Unsupported Features with Data Maps
- Data Maps and Substitution Variables
About Applications

An application is a related set of dimensions and dimension members used to meet a set of planning needs. Each application has its own accounts, entities, scenarios, and other data elements.

For more information about applications see, Managing Applications.

About Reporting Cubes

A reporting cube is a cube on which you can report on and aggregate Planning data. The primary use cases:

• The reporting cube contains data from any source, such as a data warehouse. You want to move new Planning data to it and report on the data.
• You want to report on Smart Lists in Planning, which you can't do in Planning.

Unsupported Features with Data Maps

Mapping a Planning application to a reporting application doesn't support:

• Aggregate storage database outlines with the “Duplicate Members Allowed” option selected
• Attribute dimensions
• Attribute member selections

Data Maps and Substitution Variables

When mapping an application that contains substitution variables, note:

• Substitution variables are checked when you move data, not while data maps are defined.
• The member selector displays the substitution variables that were defined for the Planning application.
• For dimension-to-dimension mappings:
  – For the Planning application, the variable name that is selected or typed is passed and evaluated when you move data.
  – For the reporting application, the variable name is evaluated against the Planning application, and then the value is used when you choose to clear data in a Push Data operation.
• For unmapped dimensions, for the reporting application, the member selector displays the substitution variables defined only for the reporting application. It's evaluated against the reporting application, and the evaluated value is validated against the respective reporting dimension for the single member and no member function.
• If metadata in a reporting application has been modified, click Synchronize before editing or moving data to synchronize reporting dimensions and members with Planning. For example, if a dimension or member was added to a reporting application, clicking Synchronize makes the member visible in Planning. Synchroniz-
ing changes from reporting application metadata may cause mappings to become invalid.

Creating Data Maps

To create a data map that maps a Planning application to a reporting application:

1. Create the reporting application.
   
   For information on cubes, see Managing Cubes. For information on adding members, see Editing Dimensions in the Simplified Dimension Editor.

2. Click Application, then Data Exchange, and then click Data Maps.

3. Click Create.

4. Enter a name and description for the data map.

5. Make your Source and Target selections as follows:
   
   • Under Source, select a cube from the available cubes for the current application.
   
   • Under Target, select an application and a cube from the available applications and cubes for the target reporting application.

   The current mappings for the selected source and target cubes are displayed under Source and Target, and any unmapped dimensions are displayed under Unmapped Dimensions.

   Note:
   
   If you are creating a data map to push data between Strategic Modeling and Planning, you will see additional Source and Target options:
   
   • For Source, Select Model and Select Consolidation
   
   • For Target, Select Model
   
   See Pushing Data Between Strategic Modeling and Planning Using Data Maps in Administering Planning Modules.

6. Optional. Change the current mappings and map any unmapped dimensions.
   
   To change the current mappings, see Changing Mapping Information.

   To map unmapped dimensions, see Mapping Unmapped Dimensions.

7. Define data map options.
   
   See Defining Data Map Options.

8. Click Save and Close.
   
   The new data map is added to the end of the data maps list.

Refreshing Data Maps

To refresh data maps:

1. Click Application, then Data Exchange, and then click Data Maps.
2. Click Refresh.

Managing Data Maps

Related Topics

- Changing Mapping Information
- Mapping Unmapped Dimensions
- Defining Data Map Options
- Editing Details for a Data Map
- Duplicating a Data Map
- Deleting a Data Map
- Synchronizing Smart Lists in Reporting Applications
- Setting Data Options
- Validation Rules for Default Members

Changing Mapping Information

To change the mapping information for the dimensions in a data map:

1. Click Application, then Data Exchange, and then click Data Maps.
2. Click an existing data map.
3. Select a dimension under Source, and then click to select a new Source Dimension or Smart List to use for the mapping.

If all of the dimensions are mapped, you can only map an existing source dimension to a Smart List (only "Smart List" is displayed when you click ). To map a source dimension to another dimension, you must first unmapped a dimension. To unmap a dimension so that it can be remapped to another dimension, select the dimension under Source, and then click . The dimension is moved to Unmapped Dimensions.

Mapping Unmapped Dimensions

To map unmapped dimensions in a data map:

1. Click Application, then Data Exchange, and then click Data Maps.
2. Do one of the following:
   - Click Create to create a new data map.
   - Click an existing data map to edit the mapping in an existing data map.
3. For any dimension under Unmapped Dimensions, click next to the dimension.

This moves the dimension out of Unmapped Dimensions and under the Target dimensions above.
4. Find the dimension that you just moved under **Target**.
   The corresponding dimension under Source will say **Select**.

5. Click ✗ next to **Select** under **Source**, and then select the Source Dimension or Smart List that you want to map to the unmapped Target dimension.

6. Click **Save and Close**.

### Defining Data Map Options

To define data map options:

1. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
2. Click an existing data map.
3. Click **Options**.
4. For **Select Items to Copy**, select the items for which you want to copy the corresponding relational data. You can copy:

   - **Comments and Attachments**
     
     If you select **Collate**, then:
     
     - If there is one-to-one mapping between the source cells and the target cells, then Comments and Attachments are copied from the source cells to the target cells.
     
     - If there isn't a one-to-one mapping between the source cells and the target cells, then the Comments and Attachments from the source cells are combined and saved in the corresponding target cells.

     If you do **not** select **Collate**, then:
     
     - If there is one-to-one mapping between the source cells and the target cells, then Comments and Attachments are copied from the source cells to the target cells.
     
     - If there isn't a one-to-one mapping between the source cells and the target cells, then last source cell's Comments and Attachments are copied into the corresponding target cell. "Last source cell" means the last source cell among multiple source cells that has non-empty relational data.

   - **Supporting Detail**
     
     - If there is a one-to-one mapping between the source cells and the target cells, then the Supporting details are copied from the source cells to the target cells.
     
     - If there isn't a one-to-one mapping between the source cells and target cells, then the Supporting details are not copied to the target cells.

---

**Note:**

The relational data (**Comments and Attachments** and **Supporting Detail**) selected to be copied in the **Data Map Options** dialog box applies only to moving data using Smart Push. See **Moving Data**.
5. Choose a **Smart List Dimension**.

6. Select **Exclude Dynamic Calc Members** if you don't want to move the dynamically calculated member data during the data map process.

    **Note:**

    **Exclude Dynamic Calc Members** is supported for data map push only and not for Smart Push.

7. **Allow Smart Push Simultaneously** allows data map and Smart Push processes to run at the same time. Click **Yes** to enable this option.

    See **Running Data Mapping and Smart Push Processes Simultaneously**.

**Editing Details for a Data Map**

When you edit data map details, you can change data mappings and map unmapped dimensions.

To edit data map details:

1. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
2. Do one of the following:
   - Click a data map.
   - Click **...** to the right of the data map that you want to edit, and then select **Edit**:
     - To edit the mapping between dimensions, see **Changing Mapping Information**.
     - To map an unmapped dimension, see **Mapping Unmapped Dimensions**.
     - To define data map options, see **Defining Data Map Options**.
     - To delete a data map, to the right of the Source and Target dimension, click **x**.
3. Click **Save and Close**.

**Duplicating a Data Map**

To duplicate a data map:

1. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
2. Click **...** to the right of the data map that you want to duplicate, and then select **Duplicate**.
3. In the **Save As** dialog box, enter a name for the data map, and then click **OK**.

    The duplicate data map is added to the data map list. It has all the details of the original data map, but is saved under the new name.
Deleting a Data Map

To delete a data map:

1. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
2. Click *** to the right of the data map that you want to remove, and then select **Delete**.

Synchronizing Smart Lists in Reporting Applications

Synchronizing Smart Lists in reporting applications identifies dimensions in reporting applications to which Smart Lists are mapped, and adds level 0 members of the selected dimensions as new Smart List entries to the selected Smart Lists. See **About Applications**.

To synchronize Smart Lists in reporting applications:

1. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
2. Click *** to the right of the data map that contains the Smart List that you want to synchronize, and then select **Synchronize**.

During synchronization, values from reporting applications in all existing mappings are appended after the last Smart List item in the appropriate Smart list. If a Smart List is mapped to two dimensions, all members from the first mapping are inserted first, and then members from the second mapping are inserted. If a member already exists in a Smart List, it's not added again. Smart List members in the Planning Smart Lists are not deleted, even if the corresponding dimension members on the reporting application are deleted.

**Note:**

If Account is mapped as Smart List to Dimension, all of the level 0 members in the Account dimension are brought in as Smart List entries when the Smart List is synchronized. For example, Smart Lists may include entries such as HSP_Average and HSP_Ending. If this occurs, delete the extra entries from the Smart List.

3. Click **OK** in the dialog box that says "Data map synchronization is complete."

**Note:**

Smart List names can't have spaces in them. If you're synchronizing Smart Lists in a reporting application, ensure that any new members don't have spaces in the name.

Setting Data Options

Launch **Options** to define how to merge relational data such as comments, attachments, and supporting detail when moving data.
To set data options:

1. Click Application, then Data Exchange, and then click Data Maps.
2. To the right of the data map that you want to edit, click ..., and then select Edit.
3. Click Options, select items to copy, and choose a Smart List dimension.
4. If you don't want to move the dynamically calculated member data during the data map process, select Exclude Dynamic Calc Members.

Validation Rules for Default Members

Default members in the reporting cube store the data that is moved from the source Planning application. If any of the following constraints are not met, or if a dimension in either application isn't mapped and has no valid default member, then a mapping isn't valid and an error message is displayed.

Rules:

• If the reporting cube is an aggregate storage database, then the default members must be level 0 members.
• If the reporting cube is a block storage database, then the default members can be any members with the Store Data property.
• If the Planning application has only dimension to dimension mappings, then the default members can be any level or data storage type.
• If the Planning application has Smart List to dimension mappings, then default members must be only level 0. In addition, the source cube must contain a dense Account dimension with at least one member associated with a Smart List.
• If Descendants (Acct_Default) is selected in a mapping, the Acct_Default member must exist in the reporting cube.

Note:

Mappings that were once valid can become invalid if dimensions, members, or Smart Lists are renamed, removed, or added. If a target cube has a change in dimensionality, you must select the corresponding application mapping on the Map Reporting Application screen and click ⬬ to refresh the data.

Moving Data

Related Topics

• Moving Data to a Reporting Cube
• Moving Data from One Cube to Another Cube Using Smart Push
• Running Data Mapping and Smart Push Processes Simultaneously
Moving Data to a Reporting Cube

After setting up data maps, you can move data to a reporting cube. Planning validates the selected data maps, and then moves the mapped Planning dimension data to the reporting cube dimensions. You can also check the Job for the job status.

Note:

If you enabled parent members for dynamic children in both a block storage and aggregate storage cube and added a dynamic member, you can use **Push Data** to successfully move data from the block storage to the aggregate storage cube without refreshing the database. After the database refresh, however, the administrator must synchronize the data map. See **Synchronizing Smart Lists in Reporting Applications**.

To move data to a reporting cube:

1. Create the reporting cube.
2. Create the data map.
3. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
4. To the right of the data map, click **...**, and then select **Push Data**.

Confirm whether you want to clear the data before moving it:

- Click **No** to add the new data to the existing data in the reporting cube. No data is cleared from the target cube when you use this option.
  
  If you map Smart Lists to dimensions, Smart List labels must match either member names or aliases in the reporting cube. **Push Data** doesn't work with Smart List entry names.

- Click **Yes** to clear the data in the target cube before moving data to it.

When the target is an aggregate storage reporting cube, note the following when clearing and moving data:

- Members with nonmatching names in the target reporting cube are ignored.

- This option works only with member names, not with member aliases.

- Use caution when using member relationships (such as Children) when selecting members for the data map, because using this option can cause the calculation script to exceed its length limit.

- If you use member relationships, this option expands the level 0 member list in the source Planning cube. If at least one member name in the source cube matches a member in the reporting cube, this option proceeds without error. If at least one member doesn't match, the option can't proceed.

When the target is a block storage reporting cube, to proceed successfully, clearing and moving data requires these conditions:
If you use member relationships, all member names in the source cube must match all member names in the reporting cube.

If you map Smart Lists to dimensions, all Smart List entries in the source cube must match all member names in the reporting cube.

If you map Smart Lists to dimensions, the Smart List entry label in the source Planning cube must match the member name in the reporting cube. If a Smart List entry label doesn't match a member name in the reporting cube, then the Smart List entry name must match the reporting cube member name.

If the previous conditions are not met, the clearing and moving data operation can't proceed.

### Moving Data from One Cube to Another Cube Using Smart Push

#### Related Topics
- About Smart Push
- Considerations When Using Smart Push
- Configuring Smart Push for a Form
- Configuring Merge Options
- Moving Data After Configuring Smart Push
- Viewing the Smart Push Status

#### About Smart Push

For more meaningful and complete reporting, planners can instantly move comments, attachments, and supporting detail from source cubes to a reporting cube while working in forms. Planners can then do more analysis on the planning data coming from the different cubes.

For example, assume that you have expense planning in one cube and revenue planning in another cube. Assume further that you use a reporting cube to consolidate your reporting needs. Without Smart Push, the data from your cubes would be moved to the reporting cube through scheduled jobs set up by administrators. Smart Push allows the data to be moved immediately to the reporting cube by planners.

You can move application data from:
- Block Storage cubes to aggregate storage cubes
- Block Storage cubes to block storage cubes

#### Considerations When Using Smart Push

When using Smart Push, keep in mind the following considerations:
- Smart Push honors metadata and approvals security.
- Smart Push isn't supported for forms from source aggregate storage cubes.
- Smart Push requires that at least one of the dense dimensions, account or period, is set as a dimension-to-dimension mapping in the data map definition.
• Numeric data across all members selected for dimensions listed in the unmapped section of the source cube in the data map isn't listed in the Smart Push definition.

• Whenever a target cube has a change in dimensionality, you must select the corresponding data map on the Data Maps page, and then click Actions and then Synchronize to refresh the data.

Configuring Smart Push for a Form

• Configuring Smart Push for a Simple Form
• Configuring Smart Push for a Composite Form

Configuring Smart Push for a Simple Form

To configure Smart Push for a simple form:

1. From the Home page, click Navigator and then under Create and Manage, click Forms.

2. Expand Forms, and then select a simple form.

3. With the simple form selected, select .

4. Click Smart Push.

   The defined mappings for the form are displayed. Click to add new mappings if desired.

5. Expand each mapping.

6. For each dimension shown under a mapping, specify the Smart Push region for the dimension:

   • Select Use Form Context to move data for all the members selected for the dimension on Layout.

   • Clear Use Form Context, and then click next to Overwrite Selection to select a member or members in the Member Selection dialog box.

   **Note:**

   You can override a mapping definition on source cube dimensions only. You can override dimension members if you take the context from a form by selecting Use Form Context. You can also override dimension members if you specify members in the Member Selection dialog box, or if you leave Overwrite Selection blank.
Tip:

Smart Push makes incremental updates to the reporting cube. To achieve this, you can select the **Use Form Context** or **Overwrite Selection** options to limit the amount of data being moved to the current context or to limit the data movement to the section that you are updating or calculating.

If the Smart Push region isn't specified using one of the above options, then member selection is taken from the data map.

7. For each mapping, define whether Smart Push will be automatic or manual:
   - Select **Run After Save** to automatically move the data when the simple form is saved.
   - Leave **Run After Save** cleared to manually move the data.
     
     For information on how to manually move the data, see [Moving Data After Configuring Smart Push](#).

8. Select from these additional Smart Push options:
   - **Run Smart Push in Background**—Allows you to continue using the application while Smart Push runs in the background
   - **Use Database Suppression**—Enables you to apply row suppression at the database level instead of at the application level, which eliminates the impact on query thresholds and improves Smart Push wait times
     
     The **Use Database Suppression** option is recommended if you are using Smart Push to load large amounts of data from a block storage cube and while using the missing block and row suppression options. This option is available for simple forms only.

     If you use Groovy rules to configure Smart Push with the **Use Database Suppression** option, there is an additional parameter to specify this option for Smart Push execution:

     ```java
     public void execute(Map<String, String> overrideMembersMap, boolean suppressMissingRowsNative)
     ```

9. Click **Save** to save the Smart Push configuration for the form.

Configuring Smart Push for a Composite Form

To configure Smart Push for a composite form:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Forms**.
2. Expand **Forms**, and then select a composite form.
3. With the composite form selected, click .
4. Click **Smart Push**.

All the simple forms for which Smart Push details are provided are displayed.
5. Select a simple form in order to use its Smart Push details in the composite form.

6. For each simple form selected, define whether Smart Push will be automatic or manual.
   - Select Run After Save to automatically move the data when the composite form is saved.
   - Leave Run After Save cleared to manually move the data.

If you select Run After Save to move the Smart Push details for a simple form when the composite form is saved, then regardless of whether you selected Run After Save for each mapping in the simple form, all of the mappings selected for Smart Push in the simple form are executed.

Similarly, if Smart Push is configured to be manual, all of the simple form’s mappings are executed regardless of whether Run After Save is selected for a mapping in a simple form.

7. Select Run Smart Push in Background if you want to continue using the application while Smart Push runs in the background.

8. Click Save to save the Smart Push configuration for the form.

### Configuring Merge Options

Merge options for comments, attachments, and supporting detail are configured on the Data Options tab available when mapping a cube for reporting. See Defining Data Map Options.

> **Note:**

When using Smart Push, relational data such as Supporting Details, Date, Smart List, and Text can’t be merged. Smart List, Date, and Text are moved only in cases of a one-to one-mapping between source cells and target cells. Empty cells are not considered for a relational data move.

### Moving Data After Configuring Smart Push

See Moving Data for information on how to move data.

How the data is moved depends on whether you selected Run After Save when you configured Smart Push for a form.

- If you selected Run After Save when you configured Smart Push for a form, the data is moved when you save the form.
- If you left Run After Save unchecked:
  1. Click Data.
  2. Click a form to open it.
  3. With the form open, select Actions, and then click Smart Push Details.
  4. Click the link in the Smart Push dialog box to move the data.
Viewing the Smart Push Status

To view the status of the data move:

1. Click Application.
2. Click Jobs.

The Smart Push job will be displayed in the Jobs console. If an error occurs, click the error link to troubleshoot the problem. If the data is moved, the Run Status will display either as Complete or Completed with Warnings. Click the job name to review the log for further details.

Running Data Mapping and Smart Push Processes Simultaneously

While moving data to a reporting cube, you can minimize the time the system is unavailable to planners. This can be achieved by moving only the required data to the reporting cube using runtime prompts in a Groovy rules-based Data Map push. For a Data Map push through EPM Automate, you can use substitution variables and set up substitution variable values before moving data using Data Maps. See these topics for more information:

- To find documentation and examples for Groovy business rules, see Frequently Asked Questions.
- For information about EPM Automate, see Working with EPM Automate for Oracle Enterprise Performance Management Cloud.

You can also further minimize downtime by allowing planners to run the Smart Push and Data Map processes simultaneously by enabling the Allow Smart Push Simultaneously option in Data Map Options.

Consider the following when running simultaneous Data Map and Smart Push executions:

- Smart Push will wait for data to clear in the reporting cube before pushing data.
- There are limits on the amount of data that can be processed simultaneously. Check the job output and design accordingly.

For information about data map options, see Defining Data Map Options.
Managing Jobs

Schedule jobs to perform common administrative tasks and lighten your workload. You can start jobs right away or schedule jobs to run later at intervals.

Related Topics
• How Jobs Save You Time
• Viewing Pending Jobs and Recent Activity
• Scheduling Jobs
• Editing and Canceling Jobs
• Canceling Rules Jobs and Ruleset Jobs
• Downloading Export Files From Your Outbox

How Jobs Save You Time

Jobs are actions such as exporting data or refreshing the database, which you can start right away or schedule to run later at intervals. To lighten your workload, define jobs to perform common administrative tasks such as:
• Import and export data
• Import and export metadata
• Refresh the database
• Map cubes

Viewing Pending Jobs and Recent Activity

To view the job listings in the Jobs console:
1. Click Application, and then click Jobs.
2. Perform any task:
   • To filter jobs and activity by criteria such as date or job type, click
   • To search for job by name, enter text in the Search, and then click
   • To view a job’s details, click the job.
Scheduling Jobs

To schedule jobs:

1. Click Application, then Jobs, and then click Schedule Jobs.
2. Specify the following:
   - The type of job you’re creating. For example:
     - **Rules**—Runs a rule that you select
     - **Import Data**—Performs a data import that was saved as a job
     - **Cube Map**—Performs a mapping operation
     - **Invalid Intersection Reports**—Runs a report that shows where data exists at invalid intersections
     - **Clear Cube**—Performs a cube clearing operation
   - When to run the job. If scheduling the job:
     - In **Schedule starting from**, select the starting date and time, including the time zone.
     - In **Name**, specify a name for the job. The name you specify is displayed with a system-generated name; such as **MyWeeklyCubeRefresh : Refresh Database**.
     - In **Recurrence Pattern**, specify the frequency with which to run the job:
       - **Hourly**—The job will start running in one hour and continue to run each hour until the ending date and time; for example, if the current time is 3:11, the job will run at 4:11, 5:11, and so on.
       - **By Minute**—Set the **Frequency** for 15 or 30 minutes. The job will start running in 15 or 30 minutes and continue to run at the selected frequency until the ending date and time; for example, if you set the job to run every 15 minutes and the current time is 3:15, the job will start to run at 3:30, then 3:45, and so on. Jobs cannot be scheduled to run in increments smaller than 15 minutes.
       - **Run Once**—The job will run once at the starting date and time.
       - **Yearly**—The job will run at the starting date and time and continue to run each year thereafter until the end date.
       - **Monthly**—The job will run at the starting date and time and continue to run each month thereafter until the end date.
       - **Weekly**—The job will run at the starting date and time and continue to run each week until the end date.
* Daily—The job will run at the starting date and time and continue to run each day until the end date.
  – If the job is recurring, select an ending date and time in End Date.

3. Click Next.

4. Select a job from the job list:
   • For details about the clear cube job options, see Creating Clear Cube Jobs.
   • For details about the following job types, see Improving Cube Performance:
     – Restructure Cube
     – Compact Outline
     – Merge Data Slices
     – Optimize Aggregation
   • For rules jobs, note the following:
     – You can filter the rules list by cube and by rule type.
     – You must click the check mark next to the rule you want to run before you can proceed.
     – For rules jobs with runtime prompts, clicking the check mark next to a rule will display the runtime prompt parameters. Set the runtime prompt values with which to run the rule in the job scheduler, and then click OK.

Note:

Hidden runtime prompts will pick up the default values that were set at design time in Calculation Manager.

• You can run up to five import or export jobs at one time.

5. Click Next.

6. Review your choices. If satisfactory, click Finish.

To edit or cancel a job after scheduling it, see Editing and Canceling Jobs.

Editing and Canceling Jobs

You can edit the schedule for pending jobs, and delete pending and complete jobs. You can't modify or delete jobs that are processing.

To edit or delete jobs:

1. Click Application, and then click Jobs.

2. To change when a job runs, click , then Edit and modify the schedule.
Note:

When changing the frequency of a job, the job will not run at the new frequency until the next time it runs at the previous frequency; for example, if the job is set up to recur daily and you change the frequency to hourly, the job won’t start running at the new hourly frequency until after the scheduled daily job runs the next day. If you want the job to begin running at the new frequency sooner, Oracle recommends that you delete the scheduled job and create a new one.

3. To delete jobs, select them, click 

Deleting Rules Jobs and Ruleset Jobs

You can cancel ruleset or rule jobs that are processing and display in Recent Activity. To cancel these jobs, click Application, then click Jobs.

To cancel a rules job, click beside the job, then 

and then Cancel.

To cancel a ruleset, click , then on Job Details, and then Cancel.

Downloading Export Files From Your Outbox

After running a metadata or data export job, you can download the export file from the Outbox.

To download files:

1. Click Application, and then click Jobs.

2. Under Recent Activity, click the job.

3. At the top of Job Details, click the export file option to select a download location.
Auditing Tasks and Data

Related Topics

- Auditing Overview
- Enabling Audit Tracking
- Viewing Audit Details

Auditing Overview

Use the Audit feature to view data tasks performed by users. You can filter audited tasks by audit type (example, Data, Approvals, or Clear Cell Details), date range (for example, Yesterday or Last 60 Days), and user name.

You must be a Service Administrator to enable audit tracking, and to view and export audit information.

These are the types of user activities the system can log in the task audit:

Table 15-1  Tasks That Can be Audited

<table>
<thead>
<tr>
<th>Audit Types</th>
<th>Tracked Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension Administration</td>
<td>• Dimension hierarchy: Adding a member or dimension; moving, deleting, or changing properties; renaming a member or dimension</td>
</tr>
<tr>
<td></td>
<td>• Performance settings: Resetting a dimension's dense or sparse setting, changing the order of dimensions</td>
</tr>
<tr>
<td></td>
<td>• Currencies: Adding or deleting currencies, setting a triangulation or reporting currency</td>
</tr>
<tr>
<td>Alias Table Administration</td>
<td>Changes to alias tables: Creating, copying, renaming, deleting, and clearing</td>
</tr>
<tr>
<td>Data</td>
<td>• Cell values</td>
</tr>
<tr>
<td></td>
<td>• Supporting detail</td>
</tr>
<tr>
<td></td>
<td>• Account annotations</td>
</tr>
<tr>
<td></td>
<td>• Cell-level documents</td>
</tr>
<tr>
<td>Launch Business Rules</td>
<td>Updates from calculation scripts and business rules (including runtime prompts)</td>
</tr>
<tr>
<td>Form Definition</td>
<td>Forms: Creating, modifying, adding rows (The audit record doesn't record how the design changed.)</td>
</tr>
<tr>
<td>Form Folder Administration</td>
<td>Folders: Created, moved, or deleted</td>
</tr>
<tr>
<td>Approvals</td>
<td>Approvals: Approval unit owners and status</td>
</tr>
</tbody>
</table>
Table 15-1  (Cont.) Tasks That Can be Audited

<table>
<thead>
<tr>
<th>Audit Types</th>
<th>Tracked Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy Version</td>
<td>Versions copied, including supporting detail, cell text, cell attachments, and data without any details. The audit record doesn’t record details (such as data and supporting detail) of the copied version.</td>
</tr>
<tr>
<td>Security</td>
<td>Access permissions to dimension members, forms, form folders, business rules, and task lists</td>
</tr>
<tr>
<td>Users Administration</td>
<td>Users added, changed, or deleted</td>
</tr>
<tr>
<td>Groups Administration</td>
<td>Groups added, changed, or deleted; users added or removed</td>
</tr>
<tr>
<td>Task List</td>
<td>Task lists: created, updated, saved, moved, and deleted</td>
</tr>
<tr>
<td>Copy Data</td>
<td>Users’ selections for Static Dimensions, Source Dimension, and Destination Dimension, including supporting detail, cell text, cell attachments, and data without any details</td>
</tr>
<tr>
<td>Clear Cell Details</td>
<td>Users’ selections for clearing supporting details, comments, and attachments</td>
</tr>
</tbody>
</table>

The Audit page displays the following information:

- Audit (task)
- Source
- Action
- User
- Name
- Date
- Details
- Property
- Old Value
- New Value

All columns can be sorted in ascending or descending order.
Enabling Audit Tracking

Audit tracking isn't enabled by default. Service Administrators must enable auditing so data changes can be tracked.

To enable auditing:

1. On the Home page, click Tools, and then click Audit.
2. From the Audit page, click Enable Auditing.
3. From the Enable Auditing page, select an audit task or tasks, and then click Save and Close.

Viewing Audit Details

To view audit details:

1. On the Home page, click Tools, and then click Audit.

   If auditing is enabled, by default the Audit page displays the audit records for Data tasks for the Last 7 Days. To enable auditing, see Enabling Audit Tracking.

2. To filter, click , and select filter criteria:
   • Audit Types—Select one or more, or All. For a list of audit types and descriptions, see Auditing Overview.
   • Date Range—Select Today, Yesterday, Last 7 Days, Last 30 Days, Last 60 Days, or Last 180 Days.

   Note:
   You can't select a date range prior to 180 days from today. The date range you select must occur between today and 180 days prior to today (including today).

   • User Name—Enter a user name or click to search for the user.

3. When you're done selecting filter criteria, click Apply.

   The grid displays the first 200 records from the audit table that match the filter criteria. You can view a legend at the bottom of the table if the filter criteria has more than 200 records in it.

4. To export the audit information to a Microsoft Excel spreadsheet, click Export, and follow the download instructions.

   When you select the Export option, the system exports all of the records matching the filter criteria to an xlsx file (format for Microsoft Excel versions 2007 and later).
Note:

- Audit information is maintained for 6 months. If you want to access audit information beyond 6 months, download it and maintain a copy of the audit records.
- If data auditing is enabled, users can see what data has changed by selecting Actions, and then Change History.
Creating and Updating Data Validation Rules

To implement business policies and practices, you can build data validation rules that are checked when conditions are met in forms. Validation messages can be generated if entered data violates validation rules. You can also build limits on submitted approval unit data using validation rules, and designate a reviewer, owner, or notifier to review data that meets some condition.

For example, data validation can ensure that a department’s capital expenses adhere to company policies by preventing planners from submitting budgets that contain capital expenditures that fall outside the company’s guidelines. Sample scenarios that can be addressed using data validation rules are described in Data Validation Rule Scenarios.

Defining data validation rules involves these main tasks:

- Identifying the data cells or location that you want to display with validation messages or in different colors when conditions are met.
- Identifying the cells that need to participate during rule evaluation, and defining the rule accordingly.
- Creating the data validation rule at the location identified, as described in this topic.

To create and update validation rules:

1. On the Home page, click **Navigator** and then under **Create and Manage**, click **Forms**.
2. Create or edit a form, and then on the **Form and Ad Hoc Grid Management** page, click **Layout**.
3. In **Layout**, right-click the grid, row heading, column heading, or cell for which you want to add or update the validation rule.
4. Select Add/Edit Validation Rules to create or update rules.

5. To add a rule, click and enter a name and description for the rule. If necessary, move the rule by selecting an option from Location. To create a rule similar to an existing rule, click and then update the rule. To view rules, click View Rule. See Viewing Data Validation Rules.

6. Update the rule.
   a. For Condition, select an option to begin the condition statement: If, Else If, Else, Then, Check Range, or Range. The first part of a rule must include an If condition. Rules must also contain some form of Then condition. See Conditions Supported by the Rule Builder.
   b. For Source Type, select an option for evaluation by the rule.
      The Source Type list displays the appropriate options for the Condition. For example, if conditions can include Current Cell Value, Cell Value, Column Value, Row Value, Member Name, Member, Cross-Dim Member, Account Type, Version Type, Var Reporting Type, UDA, or Attribute. For detailed information about each type of condition, see Conditions Supported by the Rule Builder.
   c. If applicable for the selected Source Type, enter a value in Source Value by selecting an option or entering a free form value.
   d. Select an operator for the evaluation: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, or Ends With, In, or Not In. For examples, see Data Validation Conditional Operators.
   e. Select an option for the appropriate Target Type for the rule.
   f. Update conditions by clicking an icon in the Actions column to the right of the condition builder:
      • —Add a condition next to the current row.
      • —Delete a condition at the current row.
   g. Select conditions or condition blocks to update.
      To update condition blocks, click an icon in the Condition area, at the top of the condition builder:
      • —Add a condition block within the validation rule starting with If. You can expand or collapse the condition. See Conditions Supported by the Rule Builder.
      • —Delete the selected condition block.
• Delete selected conditions, and copy them to paste in a new location.

• Copy selected conditions.

• Paste selected conditions to a new location.

• Group the selection within a condition, and add a grouping parenthesis. In addition to groupings that you set, the If statement in the grouping block is grouped when a condition is grouped, and the Custom Grouping option is enabled.

• Ungroup the selected conditions. The grouping parentheses are removed from the selected condition. One grouping is removed from the condition each time Ungroup is selected.

You can also set your own grouping for conditions by selecting Custom Grouping, then setting up the grouping in the rule definition area.

Selected conditions are displayed as shaded. To clear selected conditions, click once more to the left of the Condition column.

7. Click in the rightmost column to add processing instructions.

Note:
If the icon isn't displayed, ensure that the rule is valid and that it permits cell processing instructions. For example, cell processing instructions are included for Else, Range, and Then conditions. The icon isn't displayed until all required columns are selected for a rule.

See Formatting Cells and Setting the Promotional Path.

8. When you're ready to enable the rule to make it available in the form, select Enable Validation Rule.

Tip:
While you're building a rule, you can save the rule without enabling it. After any errors are resolved and the rule is ready to be validated and used, you can enable and save the rule to make it available in the form. You can temporarily disable a rule by clearing Enable Validation Rule.

9. When you finish updating the rule, validate the rule:
   a. Click Validate.

   The validation status displays at the top of the dialog box. You must correct errors before you save changes. If you close the dialog box without validating rules and fixing errors noted during validation, updates are not saved.
b. After fixing any errors noted during validation, ensure that **Enable Validation Rule** is selected above the rule definition area to enable the rule for the application.

c. After the rule is validated, click **OK**.

10. **Optional**: In the **Form and Ad Hoc Grid Management** page, view and update rules:

- In the **Validation Rules** pane on the right side of the **Form and Ad Hoc Grid Management** page, add, edit, or delete rules by clicking **»**, **», or **✗**.

- If multiple rules are defined at the same location, you can change the order in which rules are processed when rules have the same precedence. To move a rule up, down, or to the top or bottom of the list, select the rule and click the arrows. See **Order of Evaluation and Execution for Data Validation Rules**.

- Select form validation rules options:

  **Table 16-1   Form Validation Rules Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validate only for users with access to this form</td>
<td>If the currently logged-in user does not have access to the form, do not execute validations associated with the form when validating the Approval unit.</td>
</tr>
<tr>
<td>Validate only for pages with existing blocks</td>
<td>When enabled, the system figures out which page combinations have potential blocks and runs the validations only for those page combinations. There are a few exceptions to this. If a page combination has any Dynamic Calc, Dynamic Calc and Store, Label only, or Store with one child member, then that page is always loaded.</td>
</tr>
<tr>
<td>Validate only for cells and pages the user has access to</td>
<td>When enabled, validations are run as the currently logged-in user and not as the administrator, which means the user’s security will be applied to the form members.</td>
</tr>
</tbody>
</table>

11. In the **Form and Ad Hoc Grid Management** page, preview and validate the form, resolve any validation errors, and then save changes.

For forms that have data validation rules enabled, rules are validated when the form is loaded or saved. Data validation rules are saved when the form is saved. See **Creating Simple Forms**.

When users open the form, they can view and resolve validation messages using the **Data Validation Messages** pane. See **Working with Planning**.

### Formatting Cells and Setting the Promotional Path

After a rule is set up, use the Process Cell dialog box to set how cells display in forms, and update the promotional path based on data validations.

To format cells and set the promotional path:

1. In the Data Validation Rule Builder dialog box, click **»** in the right-most column.
2. In the Process Cell dialog box, set how the cell should appear in forms if the conditions defined by this rule are fulfilled.

When the rule is enabled, it doesn't validate unless you specify at least one of these options: a cell background color, a validation message, or a promotional path option.

- To add or update the cell background color, click 🌈. To remove the cell background color, click 🌈.

- To display a validation message for the cell, enter the message in the Validation Message field. Users see this text when they select Show Data Validation Messages in the context menu that appears when you hover over the cell in the form. It also appears as a link in the Data Validation Messages pane if data cells are flagged by the validation rules and the Display message in the Data Validation Messages pane check box is selected. For information on viewing and resolving data validation errors, see Working with Planning.

- To update the approval unit promotional path based on the data validation rule specified for the cell, select an Approvals option.

You can leave the option as None to specify no changes to the promotional path, if the data cell is in a technically valid state and you're only updating the cell background color or specifying a validation message. You can also select Update Promotional Path or Do Not Promote (to prevent the approval unit from being promoted if the condition is fulfilled). See Modifying the Approval Unit Promotional Path.

3. Click OK.

Updates for the rule are displayed in the Process column for the rule. If you specified a cell color, that color is displayed. You can preview a validation message by hovering the cursor over the Process column.

**Viewing Data Validation Rules**

After data validation rules are set up with processing instructions, you can use the View Rule dialog box to view all rules that apply to the selected grid, row, column, or cell.

To view data validation rules:

1. In the Data Validation Rule Builder dialog box, click View Rule to view all the rules at this level (grid, row, column, cell) for this location in the form.

2. Select the rule name, and then double-click the rule or click OK to view details.
Order of Evaluation and Execution for Data Validation Rules

For data validation rules in forms, the precedence for rule evaluation depends on condition priority, location of the rule, and position of the rule in the rule list (if multiple rules exist in the same location). First, cell-level rules are processed. Next, rules at the column level are processed, and then row-level rules are processed. Finally, rules at the grid level are processed. The rules are evaluated based on their position in the rule list within each level.

Location and position determine the order in which the rule will be processed. However, the priority of the processing instructions determines which rule is applied to the data cell. So, if a cell-level rule contains processing instructions with priority 4 and a grid-level rule contains processing instructions with priority 5, the grid-level rule is applied to the data cell. If all rules have processing instructions with the same priority, the first processed rule wins. The priority is based on whether the cell processing instructions specify a validation message, a color, the Do Not Promote promotional path option, or a combination of these settings.

<table>
<thead>
<tr>
<th>Default Condition Priority</th>
<th>Validation Message</th>
<th>Color</th>
<th>Do Not Promote</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (lowest)</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>1</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>5 (highest)</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Conditions Supported by the Rule Builder

These conditions are supported by the data validation rule builder: If, Else, Else If, Then, Check Range, and Range.

For details and examples of the values supported by these conditions, see these sections:

- **If, Else, Else If**—If Condition Values
- **Then**—Then Condition Values
- **Check Range, Range**—Range Condition Values

If Condition Values

For information on other conditions, see Conditions Supported by the Rule Builder.
Current Cell Value

Action:
The action is performed when the value in the current data cell on which the rule is invoked satisfies this condition.

Operators:
The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, or Ends With. These operators act on the selected target value, which can be a free form value, cell value, column value, row value, or cross-dimension member.

Condition Definition:

Condition Evaluation:
With the condition shown in the previous figure, the cells in Row 1 with member Row_Member1 will turn red when the condition is evaluated.

<table>
<thead>
<tr>
<th></th>
<th>Column_Member 1</th>
<th>Column_Member 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0</td>
<td>2.0</td>
</tr>
<tr>
<td>2</td>
<td>5.0</td>
<td>6.0</td>
</tr>
<tr>
<td></td>
<td>9.0</td>
<td>10.0</td>
</tr>
<tr>
<td>2</td>
<td>13.0</td>
<td>14.0</td>
</tr>
<tr>
<td></td>
<td>17.0</td>
<td>18.0</td>
</tr>
</tbody>
</table>

Cell Value

Action:
The action is performed when the value for the specified cell satisfies the condition.
Operators:

The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, Ends With. These operators act on the target value selected, which can be a free form value, cell value, column value, row value, or cross-dimension member.

Condition Definition:

<table>
<thead>
<tr>
<th>IF</th>
<th>Cell Value</th>
<th>A</th>
<th>1</th>
<th>2</th>
<th>Value</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Condition Evaluation:

A design-time cell can expand to one or more data cells at data entry time, as shown in the following figure. The value for the cell is the sum of values in all the expanded data cells. For example, the value for cell A1 is the sum of the values in the cells outlined in purple (1+2+5+6+9+10=33), and the value for cell A2 is the sum of the values in the cells outlined in blue (13+14+17+18=62).

<table>
<thead>
<tr>
<th></th>
<th>Column Value</th>
<th>A</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Column Value

Action:

The action is performed when the value for the specified column satisfies the condition.

Operators:

The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, or Ends With. These operators act on the target value selected, which can be a free form value, cell value, column value, row value or cross dim member.

Condition Definition:
Condition Evaluation:

A design-time column can expand to data cells at data entry, as shown in the following figure. The value for a column is the sum of the values in all the expanded data cells in that column at the current row location. The current row changes as the current cell for which the rule is being evaluated changes within the grid.

For example, the value for column A is the sum of the values in the cells outlined in purple (1+2=3), when the rule is evaluated for any cell in row 1 with member Row_Member1. The value for column A is the sum of values in cells outlined in blue (9+10=19) when the rule is evaluated for any cell in row 1 with member Row_Member3. Similarly, the value for column A is the sum of values in cells outlined in green (17+18=35) when the rule is evaluated for any cell in row 2 with member Row_Member5, and so on.

<table>
<thead>
<tr>
<th></th>
<th>Column_Member 1</th>
<th>Column_Member 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Row_Member 1</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>Row_Member 2</td>
<td>5.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6.0</td>
</tr>
<tr>
<td></td>
<td>Row_Member 3</td>
<td>9.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10.0</td>
</tr>
<tr>
<td>2</td>
<td>Row_Member 4</td>
<td>13.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14.0</td>
</tr>
<tr>
<td></td>
<td>Row_Member 5</td>
<td>17.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18.0</td>
</tr>
</tbody>
</table>

Row Value

Action:

The action is performed when the value for the specified row satisfies the condition.

Operators:

The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, Ends With. These operators act on the target value selected, which can be a free form value, cell value, column value, row value, or cross-dimension member.

Condition Definition:

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Condition Evaluation:

A design-time row can expand to one or more data cells at data entry time, as shown in the following figure. The value for a row is the sum of the values in all of the expanded data cells in that row at the current column location. The current column changes as the current cell for which the rule is being evaluated changes within the grid.
For example, the value for row 1 is the sum of the values in the cells outlined in purple (1+5+9=15), when the rule is evaluated for any cell in column A with member Column_Member1. Similarly, the value for row 2 is the sum of values in cells outlined in blue (14+18=32) when the rule is evaluated for any cell in column A with member Column_Member2, and so on.

<table>
<thead>
<tr>
<th>Cross Dim Member</th>
<th>Action:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The action is performed when the value in the data cell referenced by the cross-dimension member satisfies the condition. The current data cell members are used to fully qualify the cell for dimensions whose members are not specified in the cross-dimension. This member name is a free form entry.</td>
<td></td>
</tr>
</tbody>
</table>

| Operators: |
| The operators available for this function: =, !=, <, <=, >, >=, Equals, Not Equals, Contains, Starts With, or Ends With. These operators act on the target value selected. The target value can be a free form value, cell value, column value, row value or cross-dimensional member. It can include one member only from each dimension, and must include only members for dimensions on rows or columns. |

| Condition Definition: |
| When the previous rule is applied at grid level, the rule is invoked on each cell in the form, and the value in that cell is compared with the value in the cell outlined in purple. Thus, the cell at Row_Member 5->Column_Member 2 will turn red. |
Member Name

Action:

The action is performed if the current data cell on which the rule is invoked has the specified dimension member in its intersection. The member name for the selected dimension should be in the cube for which the form is created.

Operators:

The operators can be Equals, Not Equals, Contains, Starts With, or Ends With. The target value, which is the member name, is selected or entered free form.

Condition Definition:

<table>
<thead>
<tr>
<th>A</th>
<th>Column_Member 1</th>
<th>Column_Member 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Row_Member 1</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Row_Member 2</td>
<td>5.0</td>
</tr>
<tr>
<td></td>
<td>Row_Member 3</td>
<td>9.0</td>
</tr>
<tr>
<td>2</td>
<td>Row_Member 4</td>
<td>13.0</td>
</tr>
<tr>
<td></td>
<td>Row_Member 5</td>
<td>17.0</td>
</tr>
</tbody>
</table>

Member

Action:

The action is performed if the current data cell on which the rule is invoked has the specified dimension member (or any one of the members that result from evaluating the specified function) in its intersection.

Operator:

The available operators are In and Not In. The target value, which is the member, is selected or entered free form.

Including Attributes

Rules can include attribute values. If Source Type is Attribute, the available operator is Is, and you can type an attribute value directly in the Target Value field. If Source Type is Member, and you select the In or Not In operator in the Target Value field, you can select an attribute by clicking Variables in the Member Selection dialog box. You can use the member selection function selector to select functions for the attribute, such as NotEqual and GreaterOrEqual.

When using attribute values in data validation rules, keep in mind the way attributes are evaluated. If a rule references one or more attributes from one or more dimen-
tions, they are evaluated as an OR for attribute values from the same attribute dimension, and as an AND for attributes from different attribute dimensions. For example, if the rule includes attributes IN Red, Blue, True, Big, then all members are selected that are either (Red OR Blue) AND True AND Big. For additional information, see Selecting Attribute Values as Members.

**Condition Definition for Member Source Type**

| IF | Member | Account | In | Value | [Descendants\(\text{Total Cost}\)] | Red |

**Condition Definition for Attribute Source Type**

| IF | Attribute | Entity | Is | Value | Red |

**Account Type**

**Action:**
The action is performed if the current data cell on which the rule is invoked has an account with the specified account type in its intersection. Refers to all the currently supported Account Types: Expense, Revenue, Asset, Liability, Equity, and Saved Assumption.

**Operator:**
The available operator is Is.

**Condition Definition:**

| IF | Account Type | Is | Expense |

**Version Type**

**Action:**
The action is performed if the current cell on which the rule is invoked has a version with the specified version type in its intersection. It refers to the version types currently supported, standard bottom-up and standard top down.

**Operator:**
The available operator is Is.

**Condition Definition:**

| IF | Version Type | Is | Standard Bottom |

*ORACLE*
Variance Reporting Type

Action:
The action is performed if the current cell on which the rule is invoked has an account with the specified variance reporting type in its intersection. Refers to the available variance reporting types, Expense and Non-Expense.

Operator:
The available operator is Is.

Condition Definition:

UDA

Action:
The action is performed if the current cell on which the rule is invoked has this UDA associated with the specified dimension’s member in its intersection. The UDA reference is selected based on the selected dimension. The condition is based on the UDA for this dimension being equal to the selected value. You must select the UDA value from the drop-down list.

Operator:
The available operator is Is.

Condition Definition:

Attribute

Action:
The action is performed if the current cell on which the rule is invoked has this attribute associated with the specified dimension’s member in its intersection. The attribute reference is selected based on the selected dimension. The condition is based on the attribute for this dimension being the selected target value.

Operator:
The available operator is Is.

Condition Definition:
Then Condition Values

**Action:**
The Then conditions supported by the data validation rule builder support Process Cell conditions only. To enter Process Cell Conditions, see Formatting Cells and Setting the Promotional Path.

**Condition Definition:**

![Condition Table]

For information on other conditions, see Conditions Supported by the Rule Builder.

Range Condition Values

The Check Range and Range conditions are used together. These conditions can be used in the data validation rule builder in a Then clause or standalone.

For information on other conditions, see Conditions Supported by the Rule Builder.

**Check Range**

**Action:**
Defines the value that needs to be in a specific range.

**Value:**
This value can be the Current Cell Value or the value in a particular Row, Column or Cell.

**Range**

**Action:**
Defines a valid range for the value defined in the Check Range condition. This range includes all values that are $\geq$ the minimum value and $<$ the maximum value. If the value specified in the Check Range condition is within this range, then the processing instructions defined by this condition are applied to the data cell on which the rule is being invoked. You can define multiple ranges of values and provide different processing instructions for each range.

**Value:**
The minimum and maximum values for the range can be defined using Cell Value, Current Cell Value, Row Value, Column Value, Cross-dimension Value, or by entering a free-form value. For example, the following rule ensures that the current cell value is $\geq$ 5 and $< 10$. If this condition is met, the cell is turned red.

**Condition Definition:**
Conditional operators in the data validation rule builder can include these types of comparisons:

- Numeric comparisons, using these operators: =, !=, <, <=, >, >=.
- String value comparisons, using these operators: Equals, Not Equals, Contains, Starts With, Ends With, In, and Not In.

Rules can compare cells with different data types; for example, text and Smart List. The data type of the cell is honored if the referenced value always comes from one cell. This is the case when using Current Cell Value and Cross Dim Member to refer to a cell value. In cases where the value being compared comes from multiple cells (such as row value, column value, and cell value), the data type is defaulted to double.

When comparing values for these data types:

- For double, a string representation is used for the double value, such as "123.45." If the double is a whole number with no fractional part, such as 123.00, the integer value is used, for example, "123."
- For text, rules use only the text value for comparison.
- All other data types (currency, non-currency, percentage, and date) are treated as double.

### Table 16-3  Examples of Results for Starts With, Ends With, and Contains

<table>
<thead>
<tr>
<th>Operator</th>
<th>Compare Value</th>
<th>Compare To Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starts With</td>
<td>2.0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>1234.0</td>
<td>12.0</td>
</tr>
<tr>
<td></td>
<td>101.0</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>&quot;2.0&quot;</td>
</tr>
<tr>
<td></td>
<td>&quot;YearTotal&quot;</td>
<td>&quot;Year&quot;</td>
</tr>
<tr>
<td>Ends With</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>&quot;.5&quot;</td>
</tr>
<tr>
<td></td>
<td>&quot;YearTotal&quot;</td>
<td>&quot;al&quot;</td>
</tr>
<tr>
<td></td>
<td>&quot;YearTotal&quot;</td>
<td>&quot;Total&quot;</td>
</tr>
<tr>
<td>Contains</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>2.0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>&quot;.5&quot;</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>2.5</td>
</tr>
</tbody>
</table>
Table 16-3  (Cont.) Examples of Results for Starts With, Ends With, and Contains

<table>
<thead>
<tr>
<th>Operator</th>
<th>Compare Value</th>
<th>Compare To Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>23.567</td>
<td>23.567</td>
<td>3.5</td>
</tr>
<tr>
<td>23.567</td>
<td>23.567</td>
<td>67</td>
</tr>
<tr>
<td>23.567</td>
<td>23.567</td>
<td>&quot;23.&quot;</td>
</tr>
<tr>
<td>23.567</td>
<td>23.567</td>
<td>&quot;.56&quot;</td>
</tr>
<tr>
<td>&quot;YearTotal&quot;</td>
<td>&quot;YearTotal&quot;</td>
<td>&quot;al&quot;</td>
</tr>
</tbody>
</table>

Data Validation Rule Scenarios

These scenarios provide examples of how data validation can help implement business policies.

Scenario 1

John is hired by a company called Acme, Inc. as a consultant to design forms and implement data validation rules that enforce some of the company policies. He is asked to implement a validation rule that flags Actual amounts in red if the Total Cost in actuals exceeds the budgeted amount. This test must be repeated for each year and time period in the application. John designs the form and adds a data validation rule at cell level using a cross-dimension member, as shown in the following figures.

Form Layout at Design Time:

Data Validation Rule at Design Time:
Form at Data Entry Time with Data Validations Applied:

<table>
<thead>
<tr>
<th>Period</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sep</th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>15</td>
<td>12</td>
<td>24</td>
<td>23</td>
<td>70</td>
<td>5</td>
<td>14</td>
<td>7</td>
<td>26</td>
</tr>
<tr>
<td>Rate</td>
<td>5</td>
<td>6</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td>9</td>
<td>54</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>Total Cost</td>
<td>20</td>
<td>10</td>
<td>10</td>
<td>15</td>
<td>100</td>
<td>190</td>
<td>180</td>
<td>360</td>
<td>180</td>
<td>200</td>
<td>300</td>
<td>312</td>
</tr>
<tr>
<td>Actual</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>13</td>
<td>12</td>
<td>24</td>
<td>23</td>
<td>70</td>
<td>5</td>
<td>13</td>
<td>7</td>
<td>25</td>
</tr>
<tr>
<td>Rate</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>15</td>
<td>15</td>
<td>9</td>
<td>52</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>Budget</td>
<td>Total Cost</td>
<td>12</td>
<td>24</td>
<td>100</td>
<td>100</td>
<td>360</td>
<td>159</td>
<td>360</td>
<td>25</td>
<td>32</td>
<td>305</td>
<td></td>
</tr>
</tbody>
</table>

Tips:

- John can split Total Cost into its own segment and apply the data validation rule at that segment for a slight performance gain. However, doing so would increase maintenance as new accounts and scenarios were added to the form.

- If the requirements changed such that only the YearTotal Period in Actual had to be flagged in red, John would have two options. The best option is to add an IF entry to check if the Period member is YearTotal. Another option is to split the YearTotal member into a separate column for better performance. However, doing so would break the spreading logic, the column header for Year would repeat, and the form would be harder to maintain as new years were added.

Scenario 2

After reviewing the form designed by John in Scenario 1, Acme decides that they want Budget on the column instead of the row. To implement this requirement, John can move members within the axes to change the form layout. However, he doesn't need to update the data validation rules. John updates the form as shown in the following figure.

Form Layout at Design Time:
Scenario 3

Following the successful rollout of these forms, John is asked to implement the next policy, which is to ensure that this year’s Budget amounts are not significantly higher than previous year’s Actual amounts. If the difference is greater than 5%, then flag the difference in red.

John decides to use a member with a member formula to calculate the variance between this year’s Budget and the previous year’s Actual amount. He adds this member formula:

@varper(@Prior("Actual", 1, @Relative("Year", 0)), budget)/100;

John designs the form and adds a data validation rule at cell level, as shown in the following figure. He uses Member Name to apply the validation only to Total Cost.

Form Layout at Design Time:
Data Validation Rule at Design Time:

```
<table>
<thead>
<tr>
<th>Condition</th>
<th>Source Type</th>
<th>Source Value</th>
<th>Operator</th>
<th>Target Value</th>
<th>Target Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF</td>
<td>Member Name</td>
<td>Account</td>
<td>equals</td>
<td>Value</td>
<td>Total Cost</td>
</tr>
<tr>
<td>THEN</td>
<td>Current Cell Value</td>
<td>&gt;</td>
<td>Value</td>
<td>6.05</td>
<td></td>
</tr>
</tbody>
</table>
```

Form at Data Entry Time with Data Validations Applied:

<table>
<thead>
<tr>
<th>FY09 Actual</th>
<th>FY10 Budget</th>
<th>FY10 Variance Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>YearTotal</td>
<td>Q1</td>
<td>Q2</td>
</tr>
<tr>
<td>Units</td>
<td>70.0</td>
<td>65.0</td>
</tr>
<tr>
<td>Rate</td>
<td>54.0</td>
<td>24.0</td>
</tr>
<tr>
<td>Total Cost</td>
<td>3780.0</td>
<td>1440.0</td>
</tr>
</tbody>
</table>

Tips:

• If John isn’t allowed to change the outline, or if he experiences performance issues related to member formulas, he can use a formula column. See Designing Forms with Formula Rows and Columns.

• John defines the rule at the Variance Percent column for these reasons.
  – It improves performance. The rule is evaluated only on the cells in the Variance Percent column. If the rule had been assigned to YearTotal, it would have to be evaluated for all Time Periods for the current year budget.
  – It helps users respond to the data validation message. John can add a message to the Variance Percent column stating that the variance is higher instead of adding it to YearTotal. This way, users don’t have to look for Variance Percent to determine the difference.

• John could have flagged both YearTotal and Variance Percent in red if this had been part of the requirement.

Scenario 4

In addition to flagging the cell in red, the rule is also required to prevent anyone from promoting the approval unit if this year’s Budget is significantly higher (> 5%) than the previous year’s Actual amounts. To implement this requirement, all John needs to do
is edit the data validation rule’s processing instructions and select **Do Not Promote**, as shown in the following figure.

**Data Validation Rule at Design Time:**

![Data Validation Rule](image)

**Scenario 5**

Finally, John is asked to design a data validation rule to validate that the total compensation for employees in a particular department is within the allowed range. The rule evaluates Existing Employees in the Operations department. It validates that, if Total Compensation is > than Min allowed, and is \( \leq \frac{3}{4} \) of the compensation range for the employee’s grade, no action is needed.

If Total Compensation is greater than \( \frac{3}{4} \) of the compensation range, a validation message is provided, and the approval units must be approved by a human resource manager. If the value is less than Min and greater than Max, an error is generated, and users can't promote their approval units.

John opens the Employee Expenses Summary form in the Form Management dialog box. The form has employees and departments on the page, accounts (such as Total Compensation) on the row, and time period on the column. To make validations easier to build, John adds a calculated row to calculate \( \frac{3}{4} \) of the compensation range, and adds Min Compensation and Max Compensation members to the form, as shown in the following figures. Min Compensation and Max Compensation for the employee’s grade are calculated using member formulas.

**Form Layout at Design Time:**

![Form Layout](image)
Data Validation Rule to Stop Promotion of Approval Units:

![Data Validation Rule Diagram]

Data Validation Rule to Add the Human Resources Manager as Reviewer:

![Data Validation Rule Diagram]
Form at Data Entry Time with Data Validations Applied and Validation Messages Shown:
Managing Application and System Settings

Set application defaults and system settings such as number formatting, notifications, number of items displayed on a page, aliases, date formatting, and assigning application ownership.

Related Topics

• What Application and System Settings Can I Specify?
• Defining User Variables
• Customizing Your Display
• Announcing Upcoming Events
• Specifying Artifact Labels

What Application and System Settings Can I Specify?

You can control many aspects of the application and the system, such as:

• How to display thousands, decimals, and negative numbers in forms
• Approvals actions to be taken when you’re out of the office
• Actions about which you want to be notified
• Display the full names of users rather than user IDs

To change application and system settings:

1. Click Application, and then click Settings.
2. Specify defaults for the following application settings:
   • Alias Setting—For option descriptions, see Specifying a Default Alias Table and Setting Member and Alias Display Options
   • Number Formatting—For option descriptions, see Number Formatting
   • Approvals—Select whether to display aliases, show the approval units that are not started, and show approval units as aliases in approvals notifications
   • Notifications—Enable notifications for task lists, approvals, and job console
   • Page—Set defaults for indenting members on a page and setting the number of items on the page drop-down
Note:

The **Number of Items on the Page Drop-down** option lets you shorten the member list so that the **Search** box can be more easily seen. If the list is shortened to 10 members, for example, then you won’t need to scroll to see the **Search** box.

- **Other Options**—Set options for date format, attribute dimension date format, partial grid fetch size, whether to suppress application management options in Smart View, and whether to enable data loads for ad hoc read-only roles.

- **Predictive Planning**—Predictive Planning options for specifying the scenario/version to use for historical data values, and the prediction interval to use for worst case and best case values. Note that if no version member is specified for historical data, Predictive Planning uses Actual ([Current Form]) as the basis for historical data, which results in using the first version on the form for the historical data.

3. Specify options for the following system settings:

- **Display Users’ Full Names**—If selected, the system displays the user’s full name (for example, Victoria Hennings). If cleared, the system displays the user’s ID (for example, VHennings).

- **Include Shared Members in Cube Refresh**

- **Email Character Set**

- **Business Rules Notification**—If set to **Yes**, notifies users or groups when rules (which are enabled for notification in Calculation Manager) are completed or encounter errors. In **Notify These Users**, select the users or groups to notify.

- **Allow Drill Down on Shared Members in Ad Hoc Form**—**Yes** enables drilling on shared members in an ad hoc grid. **No** disables drilling on shared members in an ad hoc grid.

- **Minimize Approval Process Emails**—Reduces the number of emails a user receives when using Approvals. The default is **No**.

  If **Yes** is selected, only one email notification (for the approved parent entity) is sent to the new owner of the planning unit. Separate email notifications aren’t sent for every child entity in the planning unit hierarchy. If **No** is selected, owners that are set at parent nodes will receive emails for the selected node as well as an email for each child node.

- **Enable Use of the Application for**—Determines whether users can access the application in administration mode, such as during backups. When you select **Administrators**, if any nonadministrative users are logged on to the application, they are forced off the system and will not be able to log on. To restore access to an application for all users, select **All users**.

- **Assign Application Owner**—Assign ownership of the application to another administrator.

- **Enable the Display of Substitution Variables**—Set how substitution variables display in the Member Selection dialog box when users respond to runtime prompts in business rules. **Display All** displays all substitution variables.
Display None displays no substitution variables. Enable Filtering displays only substitution variables that are valid for the runtime prompt.

- **Smart View Suppression Behavior**—Choose a suppression behavior in Oracle Smart View for Office for cases where rows and columns contain missing data or zeroes.
  - **Legacy** (default)— Suppresses rows, or columns, or both that contain No Data/Missing or Zero, but not both.
  - **Standard**—Suppresses rows, or columns, or both that contain both No Data/Missing and Zero.

- **Smart View Ad Hoc Behavior**—Choose to enable enhanced ad hoc features and behaviors in Smart View.
  - **Native** (default)—Does not enable enhanced ad hoc features.
  - **Standard**—Enables enhanced ad hoc features.

The enhanced ad hoc features and behaviors are:

- **In-grid POV**—POV members are placed on the grid instead of in the POV toolbar.

- **Submit without refresh**—Using the default Submit Data button in the Smart View ribbon, all cells in a grid are submitted, including all data cells that have been explicitly modified (made dirty) and those that were not modified. For this operation, all data cells are marked dirty and submitted. Once the submit operation is complete, the entire grid will be refreshed.

- **Free-form support**—Supports empty columns and rows anywhere in a grid and changing the alias table. Additionally, supports member auto-refresh where deleted members are returned to the grid upon refresh.

- **Multiple-grid ad hoc**—Supports multiple ad hoc grids on the same Excel worksheet. With multiple-grid ad hoc, you can submit data from any grid on the sheet. Grids based on aggregate storage cubes and block storage cubes are supported on the same sheet. Each grid is independent; for example, if required, you can change the alias table for only one grid on the sheet.

- **Enable currency calculation based on scenario time period**—Choose whether to enable currency calculations in forms and batch currency rules based on the scenario time period. Selecting Yes restricts currency calculations to the range defined for the scenario time period. Selecting No calculates everything based on exchange rates and reporting currency and is not restricted based on the scenario time period range. Note that the behavior of the currency conversion script is dependent on this application setting at the time the rule is executed.

- **Export Planning Smart List textual data during daily maintenance for incremental data import**—Choose whether to perform a complete export during the daily maintenance process or to create an application backup:
  - **Yes**—Performs a complete export, such that data, including the Planning Smart List data, can be incrementally imported to an application (this option may lengthen the maintenance process duration)
  - **No** (default)—Creates an application backup during the maintenance process, such that data can be used as part of a full restoration.

For more information, see Setting the Daily Maintenance Process Time.
• **Link Accounts by Default**—For block storage (input) cubes, select whether to XREF linked account members by default.
  
  – **Yes** (default)—XREFs will be created on account members, and the application will work the same way it has in earlier releases.
  
  – **No**—XREFs will not be created for account members, which may improve the application’s performance. With **No** selected, after **Cube Refresh** is run, all existing XREFs on account members will be deleted, and non-source cubes will no longer show data from the source cube.

**Note:**

HSP_LINK and HSP_NOLINK UDAs on specific account members override the XREF setting for those account members. For example, if this option is set to **No** and you use the \@XREF function to look up a data value in another cube to calculate a value from the current cube, you can add the HSP_LINK UDA to such members to create the \@XREF function only for these specific members. If this option is set to **Yes**, HSP_NOLINK works the same way it worked in earlier releases and prevents XREFs from being created on specific members.

### Defining User Variables

You can define user variables to help planners focus on particular members, such as their department's expenses. For example, you can create a form with entities on the rows, and a user variable called Department. You can limit the number of rows displayed on the form by selecting a member for the Department user variable, such as Sales. Later, you can select another value for Department, such as Marketing.

To update user variables:

1. Click **Tools**, and then click **Variables**.
2. Click the **User Variables** tab.
3. Click **next to the variable to change.**
4. On **Member Selection**, select members.

### Customizing Your Display

Want to change the theme of your display or add your company logo to the Home page?

Use **Appearance** to customize the appearance of the application display.
To customize your display:

1. Click **Tools**, and then click **Appearance**.
2. Specify a logo or background image or select another theme.

   Note the following:
   - Both the logo and background image can be customized. Any logo image smaller than 125px wide and 25px high can fit without scaling. For large image logos, Oracle recommends that you maintain a 5:1 ratio so the image is scaled without distortion.
   - The default size for the background image is 1024x768. You can use a larger background image, however the image is scaled to fit the resolution setting of your display and the image is centered horizontally. If you want your background image to fit both a browser and a mobile device, Oracle recommends that you size the image so that it fits your biggest screen (or highest resolution device).
   - The logo and background image files must be accessible by URL. Importing image files isn't supported.

Announcing Upcoming Events

Create and send announcements to alert users about upcoming events, such as system maintenance or the running of jobs. Announcements are displayed in the Announcements area on the application's Home page.

For more information about the Announcements area, see [About the Home Page](#).

To create an announcement:

1. Click **Tools**, and then click **Announcement**.
2. Click **Create**, and enter information such as:
   - A subject that summarizes the purpose of the announcement
   - The start date; when to send the announcement. The end date is optional.
   - The content. You may need to select an editing mode (rich text or source code) first.

Specifying Artifact Labels

The **Artifact Labels** page on the **Tools** cluster enables administrators to customize artifact labels (artifact names, descriptions, and so on) based on the user's browser locale.

A few examples:

- If you create a form with a cryptic name that you don't want displayed to the user, you can define a meaningful name for the form that is displayed in the language of the user.
- If you want to create a useful instruction for an artifact that only displays in the language of the user, for example:
  
  "This formula calculates the number of regular employees away on a Leave of Absence."
Related Links

• Which Artifact Labels Can be Localized?
• Working With the Artifact Labels Grid
• Adding Languages and Defining Localized Artifact Labels
• Exporting and Importing Artifact Labels for Editing

Which Artifact Labels Can be Localized?

The application supports changing the language for the following artifacts:

• Card
• Cluster
• Dashboard
• Data Map
• Data Validation Rule
• Dimension
• Folder
• Form
• Member
• Menu
• Menu Item
• Navigation Flow
• Cube
• Approval Unit Hierarchy
• Report
• Rule
• Ruleset
• Smart List
• Smart List Entry
• Tab
• Task
• Task List
• Template
• User Variable
• Valid Intersection
Working With the Artifact Labels Grid

The Artifact Labels page displays an Excel-style spreadsheet grid that is filtered by artifact and property type.

The row axis of the grid displays the artifacts and their properties.

The column axis of the grid displays the following columns:

- **Artifact**—The type of artifact (for example, Task List or Rule)
- **Property**—The artifact's property type (for example, Name, Description, and so on)
- **Default**—Displays the artifact labels that were defined when the artifact was created.

When a language is added, a new column displays to the right of the Default column.

To view and filter the Artifact Labels grid:

1. Click **Tools**, and then click **Artifact Labels**.
2. To filter:
   a. Click , and then select the artifacts you want to work with. For some artifacts, you can further filter by property type.
   b. Click **Apply** to close the Filter window and display the artifact grid filtered by artifact type and property type.

Adding Languages and Defining Localized Artifact Labels

Administrators can add a language for a given artifact to the Artifact Labels grid from a list of supported languages. You can select only one language at a time. When you add a language, a new column for that language is added to the grid to the right of the Default column. The cells in the language-specific column are editable.

**Tip:**

Use this method to add labels directly in the artifact labels grid. This method is ideal if you only need to add or update a few labels at a time. For bulk changes or edits on artifact labels; for example, terminology changes that affect multiple labels, use the export feature to edit in Excel, then import. See Exporting and Importing Artifact Labels for Editing.

To add a language:

1. Click **Tools**, and then click **Artifact Labels**.
2. Click , and then select the artifacts you want to work with. For some artifacts, you can further filter by property type.
3. Click **Apply**.
4. Click **Add Language**.

5. Select from the list of supported languages.

6. In the language-specific column, enter artifact labels into the editable cells for each artifact property (Name, Description, and so on).

7. Click **Save**.

**Note:**

When you define a localized artifact label for the Default navigation flow (for example, editing the name of an icon on the Home page), your update will automatically propagate to all navigation flows. However, if you define a localized artifact label for another navigation flow that isn't the Default flow, then that update will override the label coming from the Default flow.

---

### Exporting and Importing Artifact Labels for Editing

You can export all the artifact labels in a given language to edit them. The labels are exported in an Excel file format (XLSX). After you edit the labels, you can import them back into the application.

**Tip:**

Use this method for bulk changes or edits on artifact labels by language; for example, terminology changes that affect multiple labels. For updates to individual artifact labels, you can edit them directly in the artifact grid. See **Adding Languages and Defining Localized Artifact Labels**.

To export and import artifact labels for editing:

1. Click **Tools**, and then click **Artifact Labels**.

2. Click **Filter**, and then select the artifacts you want to work with. For some artifacts, you can further filter by property type.

3. Click **Apply**.

4. Click **Actions**.

   - To export artifact labels:
     - Click **Export**.
     - Select the target environment of the export file:
       - **Local**—Saves the export file to a location on your local computer.
       - **Outbox**—Saves the export file to the server. See **Uploading and Downloading Files Using the Application Inbox and Outbox**.
     - Choose a language.
     - Click **Export**.
To import artifact labels:

a. Click Import.

b. Select the location of the import file:
   - Local—Loads the import file from a location on your computer. For Source File, click Browse to select the import file on your computer for the artifact you're importing.
   - Inbox—Loads the import file from the server. Enter the name of the file in Source File. See Uploading and Downloading Files Using the Application Inbox and Outbox.

c. Click Import.
Accessing More Administrative Tasks

Related Links
About the Navigator Menu
Administering Data Load Settings
Importing Using Data Management
Administering Action Menus
Administering Alias Tables
Administering Dimensions
Administering Forms
Administering Rules
Administering Rules Security
Administering Smart Lists
Administering Task Lists
Setting User Preferences
Administering Variables
Clearing Cell Details
Copying Data
Administering the Application Monitor
Managing Approvals
Managing Approval Unit Hierarchies
Assigning Approval Unit Hierarchy Scenario and Version Combinations

About the Navigator Menu

Clicking Navigator on the Home page displays a list of links that connect you to more application functionality.

Note:

Some of the links are available only if you're accessing the application from the desktop.
Administering Data Load Settings

Specify parameters to enable data to be loaded directly into an application database. Optionally, you can use advanced settings if you want to load details to child members of parent dimension members based on unique identifiers for the driver dimension.

For example, a company might load the Employee dimension member with account data for Start Date, Position, Salary Basis, and Pay Type. Because the human resource data includes placeholders for new and existing employees, the company could set up the following advanced settings:

- Data load dimension parents: New Employees, Existing Employees
- New Employees unique identifiers: Start Date, Position
- Existing Employees unique identifiers: Salary Basis, Pay Type

During data load, child members of New Employees and Existing Employees are evaluated for data updates. The unique identifiers Start Date, Position, Salary Basis, and Pay Type determine if existing data load dimension values are updated, or if new values are added: If the unique identifier's data values are the same, data is updated. If the data value is different, the next available child member is used.

To specify parameters for loading data:

1. From the Home page, click Navigator and then under Integration, click Data Load Settings.
2. For Data Load Dimension, select the dimension (such as Employee) for which data is loaded for the application.
3. For Driver Dimension, click to select the dimension into which data is loaded.
   For example, if you're loading data to Employee, the driver dimension might be Account.
4. Select the members of the driver dimension.
   For example, if the driver dimension is Account, driver dimension members might include Start Date, Grade, Position, Salary Basis, and Pay Type.
5. Optional: To use advanced settings, complete these steps.
   a. Add a row by clicking .
   b. To the right of the new field, click and select a parent member.
      For information about selecting members, see Using the Member Selector.
   c. To the right of the parent member, under Driver Dimension Unique Identifiers, select members as unique identifiers. (Members selected for this field must be included in the list of selected Driver Dimension members at the top of the page.)
      Each parent member must include at least one unique identifier member. These members determine if existing data load dimension values are updated, or if new values are added.
d. If necessary, continue adding rows by repeating the previous steps.

e. To duplicate or delete a row, click within a row, and click 📝 or ✗.

Importing Using Data Management

The basic process for importing data includes:

1. Exporting data from the source system to a CSV file
2. Defining the relationship between the source system data to members using data load mappings
3. Defining data load rules for the Location; you can create multiple data load rules for the application so that you can import data from multiple sources into a target application
4. Importing the data into the application by executing the data load rules

Importing Data Using Data Management

To import data using Data Management:

1. From the Home page, click Navigator and then under Integration, click Data Management.
2. Import data into the application, using the information in the following sections and the Administering Data Management for Oracle Enterprise Performance Management Cloud.

See these examples:
- Import Scenario Case 1: Importing Data when Member Names Match
- Import Scenario Case 2: Importing Data when Member Names Do Not Match

Import Scenario Case 1: Importing Data when Member Names Match

This example walks you through the steps to importing data into an application when the member names in the source system match the member names in the application.

See Review the Import Format.

Import Scenario Case 2: Importing Data when Member Names Do Not Match

When there are differences between the source dimensions and the target dimensions, use the mapping feature to relate the dimensions. Examples where source data and target data are in different formats:
- The source Chart of Account structure consists of Company, Account, Sub Account, Department, and Product whereas the target application uses only Department, Account, and Product.
- The source Chart of Accounts has multiple levels in a Division, whereas the target application uses only the Division level.
- The source account numbers use eight characters, whereas the target application uses only the first six characters of the account.
This section gives an example of how to import data into a target application when the member names in the source system don't match the member names in the target.

Refer to these steps and then perform the following procedure:

- Review the Import Format
- Review the Default Location

**Note:**

The following example assumes the source system has multiple products in the Electronics product line (for example, Televisions, Smart Phones, Computers), whereas the target application stores data only at the aggregated Electronics level. You want to import data from all Electronics products in the source system, but aggregate them in the target into the Electronics level in the Product dimension.

<table>
<thead>
<tr>
<th>In Source System</th>
<th>In Target Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electronics</td>
<td>110 - TVs</td>
</tr>
<tr>
<td></td>
<td>120 - Phones</td>
</tr>
<tr>
<td></td>
<td>130 - Computers</td>
</tr>
<tr>
<td>Appliances</td>
<td>210 - Dishwashers</td>
</tr>
<tr>
<td></td>
<td>220 - Ovens</td>
</tr>
</tbody>
</table>

**Table 18-1 Electronics and Appliances Hierarchies**

To set up the data load mapping for this example:

1. On the Home page, click **Navigator**, and then under **Integration**, click **Data Management**.
2. Click **Workflow**, and then click **Data Load Mapping**.
3. From **Dimensions**, select **Product**.
4. Click **Like**, enter "1*" in the **Source Value** field to import data for all Electronics (product codes that begin with "1").
5. In the **Target Value** field, enter "Electronics".

You can create multiple mappings to import and aggregate other product lines. For example, you could create another mapping and enter "2*" in the **Source Value** field to import all Appliances (product codes that begin with "2"), and enter "Appliances" in the **Target Value** field.

6. Click **Save**.

When the data load rule is executed (see Create and Execute the Data Load Rule), data for all electronic products that begin with the code "1" are imported and aggregated at the Electronics level in the target application.
Review the Import Format

When you create an application, a default import format is automatically created with the name appname_1. You use this import format to map a source data file to the application. This import format assumes that the order of dimensions in the CSV file are: Account, Entity, Amount, and then any custom dimensions in alphabetical order. If you want to use a different file format, adjust the format.

To review the default import format:

1. On the Home page, click Navigator, and then under Integration, click Data Management.
2. Click Setup, and then click Import Format.
3. Confirm the settings.

**Note:**

- Under Details, File Delimiter specifies the character that separates values in the file.
- Under Mappings, Field Number specifies the order of fields in the load file. The import format assumes the order of dimensions are Account, Entity, Amount, and then any user defined dimensions. All dimensions must be in alphabetical order.

4. Click Save.

See Review the Default Location.

Review the Default Location

The location is used as a container to load data from Data Management. When you create an application, a default location is automatically created with the name appname_1.

To review the default location:

1. On the Home page, click Navigator, and then under Integration, click Data Management.
2. Click Setup, and then click Location.
3. Review the Location settings. For example:
You can update the location as needed to include different selections for such aspects as the check rule. For more information, see *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

4. Click **Save**.

See **Create the Data Load Mapping**.

**Create the Data Load Mapping**

Data load mappings define relationships between source dimension member values and target dimension members in one dimension. The mappings are used to derive the target members for each dimension based on the source value. Data Management uses the data load mappings during the data load to dimensionalize the data that is loaded to the target dimension. You must create a data load mapping for each target dimension.

To create the data load mapping:

1. From the Home page, click **Navigator** and then under **Integration**, click **Data Management**.
2. Select the **Workflow** tab, and then click **Data Load Mapping**.
3. From the **Dimensions** drop-down list, select and then define the relationship between the **Source Value** and the **Target Value** for each dimension.

You must specify the dimensional intersection for every data value that you want to import.

For example:

a. Because the source and target member names match in this example, on the **Like** tab, enter * in the **Source Value** and **Target Value** text boxes.

b. Under **Rule Name**, enter the name of the rule.
c. Click **Save**.

See **Create and Execute the Data Load Rule**.

**Create and Execute the Data Load Rule**

After creating a data load mapping, you define a data load rule. Data load rules are defined within locations.

To create and execute the data load rule:

1. From the Home page, click **Navigator**, and then under **Integration**, click **Data Management**.
2. On the **Workflow** tab, click **Data Load Rule**.
3. Click **Add**.
4. Under **Details** on the **Data Load Rule** window, you define the rule.
5. In this example, you’ll browse to a file:
   a. Name and describe the rule.
   b. For **Category**, select the application scenario.

   **Note:**
   When you create an application, a default Category mapping is created to map to the Scenario dimension and a default Period mapping is created based on the Period dimension.

   c. Select the **Cube** for which you want to load data.
   d. Click **Select** to browse to the data load file.
   e. Click **Save**.
6. To run the data load rule, using the mapping that you created, click **Execute**.

**Scheduling Jobs in Data Management**

You can set execution times for data load rules by scheduling them. For example, you may want actual revenue data to be imported into the application on Friday of each week.

To schedule a job:

1. From the Home page, click **Navigator**, and then under **Integration**, click **Data Management**.
2. Select **Workflow** and click **Data Load Rule**.
3. Complete the load rule information.
4. Click **Schedule**.
5. Specify the type of scheduling and select the associated date and time parameters.
### Table 18-2 Schedule Details and Parameters

<table>
<thead>
<tr>
<th>Schedule Details</th>
<th>Data and Time Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td>Hours</td>
</tr>
<tr>
<td></td>
<td>Minutes</td>
</tr>
<tr>
<td></td>
<td>Seconds</td>
</tr>
<tr>
<td></td>
<td>Select (AM/PM)</td>
</tr>
<tr>
<td>Hourly</td>
<td>Minutes</td>
</tr>
<tr>
<td></td>
<td>Seconds</td>
</tr>
<tr>
<td>Daily</td>
<td>Hours</td>
</tr>
<tr>
<td></td>
<td>Minutes</td>
</tr>
<tr>
<td></td>
<td>Seconds</td>
</tr>
<tr>
<td></td>
<td>Select (AM/PM)</td>
</tr>
<tr>
<td>Weekly</td>
<td>Monday-Sunday</td>
</tr>
<tr>
<td></td>
<td>Hours</td>
</tr>
<tr>
<td></td>
<td>Minutes</td>
</tr>
<tr>
<td></td>
<td>Seconds</td>
</tr>
<tr>
<td></td>
<td>Select (AM/PM)</td>
</tr>
<tr>
<td>Monthly (day of month)</td>
<td>Monthly Date</td>
</tr>
<tr>
<td></td>
<td>Hours</td>
</tr>
<tr>
<td></td>
<td>Minutes</td>
</tr>
<tr>
<td></td>
<td>Seconds</td>
</tr>
<tr>
<td></td>
<td>Select (AM/PM)</td>
</tr>
<tr>
<td>Monthly (week day)</td>
<td>Day of Month</td>
</tr>
<tr>
<td></td>
<td>Day</td>
</tr>
<tr>
<td></td>
<td>Hours</td>
</tr>
<tr>
<td></td>
<td>Minutes</td>
</tr>
<tr>
<td></td>
<td>Seconds</td>
</tr>
<tr>
<td></td>
<td>Select (AM/PM)</td>
</tr>
</tbody>
</table>

6. Click OK.

### Drilling Through to Source Data

You can drill through on values from the application to the Data Management landing page. This landing page displays General Ledger accounts and the hyperlinked balances that were used to populate the cells in the application. If a URL has been specified in the Import Format option for the application, data is displayed in the granularity with which it was loaded in your browser. Drill through based on a URL requires that users be connected to the server on which the data resides.

### For More Information

For more information, see the following sections in Administering Data Management for Oracle Enterprise Performance Management Cloud.
Defining Category Mappings

When you create the application, a default Category mapping is created to map the Scenario dimension to a Data Management category. To update the default mapping, see *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

Defining Periods

When you create the application, a default Period mapping is created based on the Period for the application. To update the default mapping, see *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

Creating Data Load Mappings

See *Administering Data Management for Oracle Enterprise Performance Management Cloud* for information on the following topics:

- Creating Mappings Using the Explicit Method
- Creating Mappings Using the Between Method
- Using Special Characters in the Source Value Expression for Like Mappings
- Using Special Characters in the Target Value Expression
- Ignoring Member Mappings
- Importing Member Mappings
- Exporting Member Mappings

Defining Data Load Rules

After you define data load mappings, you define data load rules for the application. Data load rules are defined for locations.

See *Administering Data Management for Oracle Enterprise Performance Management Cloud* for information on the following topics:

- Creating the Data Load Rule
- Defining the Data Load Rule Details
- Defining Data Load Rule Details for a File-Based Source System

Managing Data Load Rules

After data load rules are created, you can edit, run, or delete them.

See *Administering Data Management for Oracle Enterprise Performance Management Cloud* for information on the following topics:

- Editing Data Load Rules
- Running Data Load Rules
- Checking the Data Load Rules
- Deleting Data Load Rules
Using the Data Load Workbench

Data Load Workbench enables you to perform the data load processes interactively from sources in Data Management. In addition, users can view and review the data imported from the source system.

Key features include:

- Interactive Load Process with options for import and export (a 4-step wizard that performs import, validate, export, and check)
- Provision to view Source (All)/Source (Mapped)/Target/Source and Target values
- PTD YTD Value display for ready reference
- Display options for Valid, Invalid, Ignored, and All Data
- Online and Offline Load process
- Export to Excel
- Drill through to source from the Data Load Workbench

For more information, see *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

Integrating Oracle Financials Cloud Data in Planning

Data Management facilitates data loading and write-back between a Oracle Financials Cloud application and a Planning application. Oracle Financials Cloud users who specialize in general ledger and commitment control operations can use Planning as a financial planning application. Users can generate a trial balance in the Oracle General Ledger application, and then pick that file up and load it to Planning using a few simple constructs in Data Management. In addition, data from Planning can be written back to the Oracle Financials Cloud. Both processes are supported using the WebCenter Content Management for the Cloud business process with data provided by both Oracle General Ledger and Planning.

For more information, see *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

Process Description

At a high level, the steps for integrating data in Planning:

1. Create an application that includes the dimensions to which to map General Ledger balances. See *Creating an Application*.

2. Extract the General Ledger file from Oracle Financials Cloud and move it to the application. See "Extracting the General Ledger File" in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

3. Register and configure a source connection for the General Ledger file. See "Configuring a Source Connection in Data Management" in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

4. Load data from the General Ledger file to the application. Then after mapping data to the application dimensions, write back the data to Oracle Financials Cloud. See "Writing Back to the General Ledger" in *Administering Data Management for Oracle Enterprise Performance Management Cloud*. 
5. Optionally, you can extract the data from the integrated application to create a custom target application. See "Extracting the Integrated EPM Cloud and Oracle General Ledger Data to a Flat File" in Administering Data Management for Oracle Enterprise Performance Management Cloud.

Administering Action Menus

Related Topics

- Creating and Updating Action Menus
- Working with Action Menu Items
- Defining Action Menu Items

Creating and Updating Action Menus

Administrators can create right-click (or action) menus and associate them with forms, enabling users to click rows or columns in forms and select menu items to:

- Launch another application, URL, or business rule, with or without runtime prompts
- Move to another form
- Move to Manage Approvals with a predefined scenario and version
- Open Job or Copy Version

The context of the right-click is relayed to the next action: the POV and the Page, the member the user clicked on, the members to the left (for rows), or above (for columns).

When designing forms, use Other Options to select menus available for Form menu item types. As you update an application, update the appropriate menus. For example, if you delete a business rule referenced by a menu, remove it from the menu.

To create, edit, or delete action menus:

1. From the Home page, click Navigator, and then under Create and Manage, click Action Menus.
2. Perform an action:
   - To create an action menu, click +, enter the menu's name, and then click OK.
   - To edit details of an action menu, select it, and then click . See Working with Action Menu Items.
   - To delete action menus, select them, click , and then click OK.

Working with Action Menu Items

The Edit Menu page displays menu items on the current action menu, including names, labels, required dimensions, icon, and type, such as URL, Form, Business Rule, Manage Approvals, Menu Header, Form, Job, and Copy Version.

To work with action menu items:
1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Action Menus**.

2. Select a menu, and then click ✍️.

3. **First time only:** To add the first item to the menu, click **Add Child** and **Save**.

4. Select a menu item and:
   - To add menu items below the selected item, click **Add Child** (available for Menu Header menu types).
   - To add menu items at the same level as the selected item, click **Add Sibling**.
   - To edit menu items and define the menu item properties, click **Edit Menu Item**.
   - To delete menu items, click **Delete Menu Item**.
   - To change the order of menu items within the same level, click **Move Up** or **Move Down**. You can move multiple items.

5. Click **Save**.

   Click **Save As** to save the current selections under a new menu name.

### Defining Action Menu Items

To define action menu items:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Action Menus**.

2. Select a menu and then click ✍️.

3. Select the menu item, and then click **Edit Menu Item** or **Add Sibling**.

4. Define the menu item:

   **Table 18-3   Edit Menu Item Options**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Menu Item</strong></td>
<td>Enter a unique name containing only alphanumeric and underscore characters, with no special characters or spaces</td>
</tr>
<tr>
<td><strong>Label</strong></td>
<td>Enter text to be displayed when the menu is selected. Spaces and special characters are allowed. Menu labels display in the user interface. Labels can be text or can reference a resource variable by name. For example, to set a menu's label to File, set it to <strong>File</strong> directly or to the name of a resource, such as <strong>LABEL_FILE</strong>, which can be localized.</td>
</tr>
</tbody>
</table>
Table 18-3  (Cont.) Edit Menu Item Options

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Icon</td>
<td><strong>Optional:</strong> In context of the application server, enter the path and filename to a graphic to display by the menu. (In other words, the graphic or image referenced must be within the Planning Web application root folder.) For example: Images/green.gif</td>
</tr>
<tr>
<td>Type</td>
<td>Select the menu item type to determine available properties.</td>
</tr>
</tbody>
</table>

**Note:**

No properties are available for Menu Header.

- **URL**—Navigate to the specified URL.
- **Form**—Launch a selected form. The member selection context for the member, page, and POV is retained when users right-click in the source form. If the target form contains these dimension members on the page, its page is set to match the context.
- **Business Rule**—Launch the selected business rule.
- **Manage Approvals**—Move to Manage Approvals to work with approval units
- **Menu Header**—Create a menu under which you can create children menu items. To display a separator bar on the menu at this item, enter one hyphen as the Label. In this case, the Required Dimension list isn't available.
- **Previous Form**—Create a menu that returns the user to the previous form.
- **Copy Version**—Create a menu that opens Copy Version to enable the user to copy data for the current form.

**Required Parameters**

Select a dimension, or select an option for where the menu item displays: Point of View, Page, Row, Column, Members Only, Cell Only. For example, if you select Account, users can right-click Account members on a form to open the menu. If you select Row, the menu is available when users right-click a row. Selecting None makes the menu available whenever the user right-clicks in the form.

5. Define menu item properties, which differ for menu item types:
### Table 18-4 Options for Menu Item Types

<table>
<thead>
<tr>
<th>Type</th>
<th>Options</th>
</tr>
</thead>
</table>
| **URL**              | a. In **URL**, enter the complete URL to which to direct the user. For example: `http://server name/HFM/Logon/HsvLogon.asp`.  
|                      | b. Select **Use Single Sign-on** to append the SSO token to the URL.  
|                      | c. Select **Include Context in URL** to include the context.  |
| **Form**             | a. In **Form Folder**, select the folder containing the destination form.  
|                      | b. In **Form**, select the form.  |
| **Business Rule**    | a. In **Cube**, select the cube for which the business rule is available.  
|                      | b. In **Business Rules**, select the business rule to launch.  
|                      | c. In **View Type**, select how to display runtime prompt pages:  
|                      | | • **Classic View**—Use the default application view  
|                      | | • **Streamline View**—Display each runtime prompt on a different line  
|                      | d. **Optional**: In **Window Title**, enter a title to display instead of Runtime Prompts.  
|                      | e. **Optional**: In **OK Button Label**, enter the text to display for the OK button.  
|                      | f. **Optional**: In **Cancel Button Label**, enter the text to display for the Cancel button.  
|                      | g. **Optional**: In **Launch Confirmation Message**, enter text to display when the business rule is invoked, but before it's launched. This option enables administrators to provide meaningful messages to planners about the consequences of launching business rules.  |
| **Manage Approvals** | Specify the approval unit to which the user is directed by selecting a scenario and a version. |
| **Previous Form**    | Enter the name of the menu item that will return the user to the previous form. |
Table 18-4  (Cont.) Options for Menu Item Types

<table>
<thead>
<tr>
<th>Type</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy Version</td>
<td>Enable end users to use Copy Version to copy form data for the current form, including supporting details, annotations, cell text, and cell documents to another version. Select the following default values:</td>
</tr>
<tr>
<td></td>
<td>a. In Scenario, select the scenario from which to copy.</td>
</tr>
<tr>
<td></td>
<td>b. In Copy From, select the version that contains the data to copy.</td>
</tr>
<tr>
<td></td>
<td>c. In Copy To, select the version to which to copy the data.</td>
</tr>
</tbody>
</table>

6. Click **Save**.

Administering Alias Tables

Related Topics

- About Aliases
- About Alias Tables
- Working with Alias Tables
- Specifying a Default Alias Table and Setting Member and Alias Display Options

About Aliases

You can assign alternate names, or aliases, to Account, Currency, Entity, Scenario, Period, Version, Years, and user-defined dimension members. The application allows up to 30 aliases per dimension member, including the default alias. Aliases can have the same name within an alias table and across alias tables.

Aliases can also have:

- The same name as a member
- The same alias on members that are parent and child
- The same name for members from different dimensions or from the same dimension
Note:

- You can't have the same alias for two members that are siblings because there would be no way to uniquely identify the member. This rule is enforced by the application for base members, but not for shared members.

  This rule isn't enforced for shared members because you can't directly set aliases for shared members; shared member aliases are inherited from the alias of their base member. It's possible to create an alternative hierarchy where you can have two shared members that have the same alias and are siblings. However, this situation is discouraged if you want to reference these members by their aliases from an adhoc grid because, when you type the alias into the grid and submit it to the application, the application can't uniquely resolve this member and it will return an error. If you're using the alias for display purposes only then there will be no issue, but this design is discouraged because, visually, there is no way to differentiate between these two members.

- Although aliases can have the same name as a member, be careful not to set the alias of member1 to be the same name as member2. This can lead to unintended results and will cause confusion on the form grids.

- Member names must be unique so that they can be used in rules and form designs.

About Alias Tables

You can create and update alias tables, and set a default alias table for the application. Follow naming conventions in Naming Restrictions.

Multiple alias tables support, for example, these language combinations:

- English, French, German, Spanish, and Italian
- Japanese and English
- Korean and English
- Turkish and English

Note:

Alias table support isn't restricted to these language combinations.

You can set alias tables to display members in an application. Planners can set alias tables in preferences.

Working with Alias Tables

You can add, edit, rename, and delete alias tables or clear alias table values. You can also copy alias table content from one table to another.
To work with alias tables:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Alias Tables**.

2. Choose a task:
   - If adding an alias table, click **Add - Alias Table**, and then in **Add - Alias Table**, enter a name.
   - If editing or renaming alias tables, select the alias table, then click **Edit - Alias Table**, and then for **Edit - Alias Table**, enter a name.
   - If deleting alias tables, select the alias table, and then click **Delete**.

   **Note:**
   You can't delete the Default alias table.

   - If clearing the values in an alias table, select the alias table to clear, then click **Clear Values**.

   **Note:**
   Clearing the alias table removes the content of the table, but doesn't remove the table.

   - If copying alias table content, select the alias table, click **Copy**, then select the destination alias table, and then click **Copy**.

   **Note:**
   The destination alias table must exist. Copying doesn't create tables.

3. Click **OK**.

**Specifying a Default Alias Table and Setting Member and Alias Display Options**

If you create alias tables with aliases for Account, Currency, Entity, Scenario, Period, Version, Years, and user-defined dimensions and members, you can select a default alias table for the application. Users can set preferences for which set of aliases (stored in an alias table) to use for displaying member and dimension names.

To select the application's default alias table:

1. From the Home page, click **Application**, and then click **Settings**.

2. For **Alias Table**, select an alias table.

3. For **Display Member Label as**, select the option that enables the kind of member data to be displayed in the member selector throughout the application:
• **Default**—The data determined by the form, grid, or dimension settings
• **Member Name**—Just member names
• **Alias**—Just member aliases, if defined
• **Member Name:Alias**—Names followed by aliases, if defined
• **Alias:Member Name**—Alias, if defined, followed by the names

4. Click **Save** or **Reset**.

## Administering Dimensions

This section documents how to edit dimensions using the Classic Dimension Editor which is accessed by using the **Dimensions** link in the Navigator menu. In the 17.05 (May 2017) update, we released the Simplified Dimension Editor.

For detailed information about using the Simplified Dimension Editor, see **Editing Dimensions in the Simplified Dimension Editor**.

### Related Links

- **About Dimensions**
- **Working with Dimension Hierarchies**
- **About Custom Dimensions, Entities, Accounts, Periods, and Cubes**
- **Adding or Editing User-Defined Custom Dimensions**
- **Working with Members**
- **Working with Attributes**
- **Working with Attribute Values**
- **Customizing Calendars**
- **Setting Up Currencies**
- **Setting up Dynamic Time Series Members**
- **Working with UDAs**
- **Working with Member Formulas**

## About Dimensions

Dimensions categorize data values. Seven dimensions are included with the application: Account, Entity, Scenario, Version, Period, Years, and Currency. You can create up to 13 user-defined custom dimensions.

Members are components of dimensions.

For complete information about dimensions, see **Dimension Overview**.
Working with Dimension Hierarchies

Related Topics

- Filtering the Dimension View by Cube
- Finding Dimensions or Members
- Sorting Members
- About Assigning Access to Members
- Viewing a Member’s Ancestors
- Determining Where Members Are Used in an Application

Filtering the Dimension View by Cube

You can filter the dimension view by cube. When you select a cube, only dimensions used in that cube are displayed on the Dimensions page.

To filter the dimension view by cube:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. For Cube, select the cube.

The application displays only the dimensions used in the selected cube.

Finding Dimensions or Members

To find dimension members in dimension hierarchies:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. For Dimension, select the dimension for the member.
3. For Search, select Name, Alias, or Both.
4. Enter the search text (member name, alias, or partial string) for which to search.
5. Click or .

Sorting Members

You can sort members in ascending or descending order, by children or descendants. Sorting members affects the outline.

To sort members:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. For Dimension, select the dimension for the members.
3. On Dimensions, select the members whose children or descendants you want to sort.
4. For **Sort**, select children or descendants. Sorting by children affects only members in the level immediately below the selected member. Sorting by descendants affects all descendants of the selected member.

5. Click 🔄 to sort by ascending order or 🔄 to sort by descending order.

6. Click **OK**.

   The next time you create or refresh the database, the outline is generated with members in the order that is displayed.

**About Assigning Access to Members**

Administrators can assign permissions to members.

You can assign permissions to members by selecting the dimension property **Apply Security**. If you omit or clear the **Apply Security** setting, all users can access the dimension's members. By default, the Account, Entity, Scenario, and Version dimensions are enabled for access permissions. Optionally, you can enable this option for Period, Years, and Custom dimensions.

For complete information, see Assigning Access to Dimension Members. To enable access to members, see Editing Dimension Properties in the Simplified Dimension Editor.

**Viewing a Member’s Ancestors**

To view a member’s ancestors:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select a dimension.
3. Select the member in the dimension hierarchy.
4. Click 📈.
5. Click **OK**.

**Determining Where Members Are Used in an Application**

To view where members are used in an application:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension whose member’s usage you want to view.
3. Click 📈.

**About Custom Dimensions, Entities, Accounts, Periods, and Cubes**

See the following topics for more information:
• About Custom Dimensions
• About Entities
• About Accounts
• Accounts, Entities, Periods, and Cubes

Adding or Editing User-Defined Custom Dimensions

User-defined custom dimensions must conform to guidelines listed in Naming Restrictions.

Table 18-5 Properties for User-Defined Custom Dimensions

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension</td>
<td>Enter a name that is unique across all dimensions.</td>
</tr>
<tr>
<td>Alias</td>
<td>Optional: Select an alias table. Enter an alternate name for the dimension. See About Aliases.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional: Enter a description.</td>
</tr>
<tr>
<td>Valid for Cubes</td>
<td>Select cubes for which the dimension is valid. Clearing this option makes all members of the dimension invalid for the deselected cube.</td>
</tr>
<tr>
<td>Apply Security</td>
<td>Allow security to be set on the dimension members; must be selected before assigning access rights to dimension members. Otherwise, dimensions have no security and users can access members without restriction.</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Select a data storage option. The default is Never Share.</td>
</tr>
</tbody>
</table>

To add or change user-defined dimensions:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. Click + or select an existing dimension and click .
3. Specify any of the properties listed above.
4. Click Save.
5. Click OK.

Click Refresh to revert to the previous values and keep the page open.

• Setting Dimension Properties
• Setting Dimension Density and Order
• Setting the Evaluation Order

Setting Dimension Properties

Dimension properties must conform to guidelines listed in Naming Restrictions.
Table 18-6  Dimensions Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension</td>
<td>Enter a name that is unique across all dimensions.</td>
</tr>
<tr>
<td>Description</td>
<td><strong>Optional</strong>: Enter a description.</td>
</tr>
<tr>
<td>Alias Table and Alias</td>
<td><strong>Optional</strong>: Select an alias table. Enter an alternate name for the dimension. See Administering Alias Tables.</td>
</tr>
<tr>
<td>Valid for Cubes</td>
<td>Select cubes for which the dimension is valid. Clearing this option makes all members of the dimension invalid for the deselected cube.</td>
</tr>
<tr>
<td>Two Pass Calculation</td>
<td>Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with Dynamic Calc or Dynamic Calc and Store properties.</td>
</tr>
<tr>
<td>Apply Security</td>
<td>Allow security to be set on the dimension members; must be selected before assigning access rights to dimension members. Otherwise, dimensions have no security and users can access members without restriction.</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Select a data storage option. The default is <strong>Never Share</strong>.</td>
</tr>
<tr>
<td>Display Option</td>
<td>Set application default display options for the Member Selection dialog box. Select <strong>Member Name</strong> or <strong>Alias</strong> to display members or aliases. <strong>Member Name:Alias</strong> displays members on the left and aliases on the right. <strong>Alias:Member Name</strong> displays aliases on the left and members on the right.</td>
</tr>
</tbody>
</table>

Setting Dimension Density and Order

The **Performance Settings** tab enables you to set dimensions as sparse or dense and set their order of precedence.

To manage performance settings:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Dimensions**.
2. Select the **Performance Settings** tab.
3. For each dimension, set its **Density** as **Dense** or **Sparse**.
   - See About Sparse and Dense Dimensions.
4. Set the order of precedence by selecting a dimension and clicking ⬆ or ⬇ next to the **Position** column heading.
   - See About Reordering Dimensions.
Setting the Evaluation Order

The **Evaluation Order** tab enables you to specify which data type prevails when a data intersection has conflicting data types. For example, if Account members are set to the Currency data type, and Product members are set to the Smart List data type, you can set whether the Currency or Smart List data type prevails at an intersection.

To set evaluation order:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Dimensions**.
2. Select **Evaluation Order**, and then select the cube.
3. From **Available Dimensions**, select dimensions and move them to **Selected Dimensions**:
   - moves selected dimensions
   - moves all dimensions
   - removes selected dimensions
   - removes all dimensions
   You need select only dimensions whose members have specific data types (that is, their data type isn't "Unspecified"). The data type "Unspecified" doesn't conflict with another data type.
4. If you select multiple dimensions, set the order of precedence by clicking ↑ or ↓.
5. Click **Save**.

Working with Members

You can assign access rights to members, rearrange the dimension member hierarchy, and share members of the Entity, Account, and user-defined custom dimensions.

- **About Dynamic Members**
- **Adding or Editing Members**
- **Deleting Members**
- **Deleting Parent Members**
- **Viewing Member Properties from Forms**
- **Working with Shared Members**
- **Creating Shared Members**

About Dynamic Members

Dynamic members are members that users can create when working with business rules. They're sometimes referred to as "on the fly members". An administrator enables end users to create dynamic members under a parent member and must refresh the database to create the required placeholders in Essbase. In business rules with runtime prompts, users can then create members by entering desired member names.
in the runtime prompts. Subsequent database refreshes will rename used dynamic children to the names specified by the end users and recreate the required placeholders in Essbase. For information about working with business rules and dynamic members, see *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*.

If a parent member is enabled for adding dynamic children (as described in this section), users can create new members by entering their name in the runtime prompt.

To enable a parent member for adding dynamic children:

1. From the Home page, click **Navigator**, and then under **Create and Manage**, click **Dimensions**.

2. Edit the parent member and select the option **Enable for Dynamic Children** (see **Adding or Editing Members**).
   - **Optional**: Set the member property **Number of Possible Dynamic Children** (the default is 10). This setting determines the number of placeholders that are created for dynamically adding or loading members under the parent. If all placeholders are used, subsequently added children are added as normal members and can't be used until the database is refreshed.
   - **Optional**: Set the member property **Access Granted to Member Creator** (the default is Inherit).

3. Refresh the database to create the placeholders for dynamic members in the databases for the cubes in which the members are used.

4. In Calculation Manager:
   a. Create a business rule with a runtime prompt (of the Variable type Member). Under the **Dynamic Member Parent** column, use the **Member Selector** to select the parent member that you enabled for dynamic children in the application.
   b. Select the option **Create dynamic members**.
   c. Deploy the business rule.

For information about working with business rules, see the *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud*. 
**Note:**

- In Calculation Manager, selecting both the **Create dynamic members** and the **Delete dynamic members** options enables temporary dynamic members to be created for calculations, and those temporary dynamic members will be deleted after the business rule completes.

- Selecting the **Delete dynamic members** option alone presents the end user with the **Member Selector** in the runtime prompt so he can delete any member that he created dynamically under the parent (if he has write access to that member). This enables the end user to have full control over cleaning up and managing the members under the parent. The key is proper design to fit your requirements, setting the appropriate access rights to dynamic children with the **Access Granted to Member Creator** member property.

If all the above conditions are met, when users run the business rule with the runtime prompt, they enter the name of the dynamic member, and then click **Launch**. If the business rule runs successfully, the member is created in the dimension hierarchy under the dynamic member’s parent.

Child members that you import under parent members enabled for dynamic children are added as dynamic child members if there are dynamic member placeholders in Essbase. After the placeholders are full, any remaining children are added as normal members, and can't be used until the database is refreshed.

**Note:**

If you load a parent member that is enabled for dynamic children and its child members during the same import, the child members are loaded as normal members. This is because the database must be refreshed to create the placeholders in Essbase.

**Adding or Editing Members**

Members must conform to guidelines listed in **Naming Restrictions**. Shared members must be consistent with **Working with Shared Members**.

**Table 18-7  Member Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name that is unique across all dimension members.</td>
</tr>
<tr>
<td>Description</td>
<td><strong>Optional:</strong> Enter a description.</td>
</tr>
<tr>
<td>Alias Table</td>
<td><strong>Optional:</strong> Select the alias table to store the alias name. Enter an alternate name for the member in <strong>Alias</strong>. See <strong>Administering Alias Tables</strong>.</td>
</tr>
</tbody>
</table>
Table 18-7  (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>For Account members only: Account Type</td>
<td>Select Expense, Revenue, Asset, Liability, Equity, or Saved Assumption. For descriptions, see Account Types.</td>
</tr>
<tr>
<td>For Account members only: Variance Reporting</td>
<td>If the account type is Saved Assumption, select Expense or Non-Expense. Designate the saved assumption as a revenue, asset, liability, or equity account.</td>
</tr>
<tr>
<td>For Account members only: Time Balance</td>
<td>Select Flow, First, Balance, Average, Fill, Weighted Average - Actual, Actual, or Weighted Average - Actual, 365. For descriptions, see Time Balance Property.</td>
</tr>
<tr>
<td>For Account members only: Skip</td>
<td>If the account type is Asset, Equity, or Liability, select None, Missing, Zeros, or Missing and Zeros. For descriptions, see Setting Account Calculations for Zeros and Missing Values.</td>
</tr>
<tr>
<td>For Account members only: Exchange Rate Type</td>
<td>Select Average, Ending, or Historical. For descriptions, see Data Type and Exchange Rate Type.</td>
</tr>
<tr>
<td>For Account members only: Data Type</td>
<td>Select Currency, Non-currency, Percentage, Date, or Text. For descriptions, see Data Type and Exchange Rate Type.</td>
</tr>
<tr>
<td>For Account members only: Distribution</td>
<td>Sets the weekly distribution. Available for leaf Account members if the option was selected when creating the application and the base time period is 12 months.</td>
</tr>
</tbody>
</table>
Table 18-7  (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hierarchy Type</td>
<td>Hierarchy Type is available for dimensions bound to an aggregate storage cube. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be stored.</td>
</tr>
</tbody>
</table>

Note:

For members with a stored hierarchy type, the only valid cube aggregation options are Addition or Ignore. In a stored hierarchy, the first member must be set to Addition. For members with a dynamic hierarchy type, all cube aggregation options are valid. Stored hierarchy members that are not children of Label Only members must have Addition set as the consolidation operator. Children of Label Only members can be set to Ignore.

Data Storage

Select a data storage property. The default is Never Share for new custom dimension members (except root members).

Two Pass Calculation

Recalculate values of members based on values of parent members or other members. Available for Account and Entity members with Dynamic Calc or Dynamic Calc and Store properties.

For Entity members only: Base Currency

For Standard multiple currency applications only, select the Entity member’s base currency.
Table 18-7  (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plan Type</td>
<td>Select the plan types (or cubes) for which the member is valid.</td>
</tr>
</tbody>
</table>

**Note:**
A member can belong to both aggregate storage and block storage cubes.

Select an aggregation option for each selected cube. You can select a source cube only if multiple cubes are valid for the member. Only cubes and aggregation options for which the member’s parent is valid are available. If the parent isn't valid for a cube or aggregation option, neither is the child member. Deselecting a cube for an account or entity parent member deselects it for all descendents of that parent. For members with a stored hierarchy type, the only valid aggregation options are Addition or Ignore. See Aggregation Options.

**Caution:**
Deselecting a cube for dimension members after data is entered into an application may result in loss of data when an application is refreshed. For account members, data is lost if the deselected cube is the source cube.

For Entity members only: **Base Currency**
Members of a custom dimension and a Period dimension can set usage by cube, similar to the Account and Entity dimensions.

For Standard multiple currency applications only, select the base currency for the Entity member.
### Table 18-7  (Cont.) Member Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>For Account members only: <strong>Source Cube</strong></td>
<td>Select the source cube for the member. A shared member is a pointer to the base member and isn't stored; this is disabled for shared members. The source cube of a shared Account member matches the source cube of the base member, even though Source Plan field is unavailable because it doesn't apply to shared members.</td>
</tr>
<tr>
<td><strong>Smart Lists</strong></td>
<td>Optional: Select a Smart List to associate with the member.</td>
</tr>
<tr>
<td><strong>Enable for Dynamic Children</strong></td>
<td>Enables users to create children for this member by entering a member name in the runtime prompt for a business rule that has been configured with a dynamic parent member (see About Dynamic Members).</td>
</tr>
<tr>
<td><strong>Number of Possible Dynamic Children</strong></td>
<td>This option is available if <strong>Enable for Dynamic Children</strong> is selected. Enter the maximum number of dynamically-added members that users can create. The default is 10.</td>
</tr>
</tbody>
</table>
| **Access Granted to Member Creator** | This option is available if **Enable for Dynamic Children** is selected. Determines the access that member creators have to dynamic members that they create with a runtime prompt:  
  - **Inherit**—The member creator will inherit the closest parent's access to the newly-created member.
  - **None**—The member creator will not be assigned any access to the newly-created member. (An administrator can later assign the member creator access to the members.)
  - **Read**—The member creator will be assigned Read access to the newly-created member.
  - **Write**—The member creator will be assigned Write access to the newly-created member. |

**Note:**  
If an administrator changes these settings, they affect only future dynamic members; they don’t retroactively affect dynamic members.
To add or edit members:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension.
3. Perform one action:
   - To add a child member, select the parent level of the dimension hierarchy to which to add a member and click 📝.
   - To add a sibling, select the level of the dimension hierarchy to which to add a sibling and click 📝.
   - To edit a member, select that member from the dimension hierarchy and click 📝.

**Note:**
To add an All Years parent member that includes all members of the Years dimension, select the Years dimension and then click 📝. The All Years parent member enables users to view the accumulated data across multiple years, for example, a project's total cost up to its end date. The All Years member doesn't include the No Year member, if one is defined for the application.

4. On **Member Properties**, set or change member properties described in **Table 1**. If you don't see the new member on the page, click **Next**.
5. Click **Save** to save information to the relational database and see changes in the dimension hierarchy.
6. Refresh the database so edited members are visible to planners entering data.
7. After creating a dimension member, you typically complete these tasks:
   - Assign access. See **Assigning Access to Dimension Members**.
   - Specify attributes.

**Deleting Members**

Each data value is identified by a set of dimension member values and a cube. Deleting dimension members or deselecting the cube results in data loss when refreshing an application. Deleting entity members deletes all approval units (including data) associated with them.

Before deleting members, understand where in the application they are used (in which forms, approval units, exchange rates, and so on) by using Show Usage. See **Determining Where Members Are Used in an Application**.

You must delete the entity member throughout the application before deleting it from Dimensions. For example, if the entity member is used in a form, you must delete it from the form before deleting it from Dimensions.
When deleting a large subtree of entities, you can improve performance if you first exclude approval units for the subtree (by excluding the root member) for all scenarios and versions. See Approvals Process.

To delete members:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension whose member you want to delete.
3. From the dimension hierarchy, select the entity member to delete.
4. Click **X**. Deleting a base member also deletes its shared members.
5. Click **Yes**.
6. Update and validate business rules and reports.

### Deleting Parent Members

Data values are identified by a set of dimension member values and a cube. Deleting dimension members or deselecting the cube results in data loss when refreshing the application.

To delete a parent member and all its descendants from the dimension hierarchy:

1. From the Home page, click **Navigator** and then under **Create and Manage**, click **Dimensions**.
2. Select the dimension whose member and descendants you want to delete.
3. Select the member whose branch to delete.
4. Click **X**.
5. Click **Yes**.

### Viewing Member Properties from Forms

To view member properties from forms:

1. In the form, select a row or column member and right-click.
2. Select **Show properties in outline**.
   The Dimensions page displays the member highlighted in the hierarchy.
3. **Optional**: Select **Edit** to view the member's properties, then click **Cancel**.

### Working with Shared Members

Sharing members allow alternate rollup structures within an application. A base member must exist before you can create a shared member. You can create multiple shared members for the base member. A base member must display before its shared members in position from top to bottom.
Shared members are available for Entity, Account, and user-defined custom dimensions. Shared member values can be ignored to avoid double-counting values when you roll up the outline.

Shared members share some property definitions with base members, such as member name, alias name, base currency, and cubes for which members are valid. Shared members must have unique parent members and different rollup aggregation settings. Custom attributes, custom attribute values, and member formulas are not allowed for shared members. Renaming base members renames all shared members.

Shared members can't be moved to another parent member. You must delete shared members and recreate them under different parent members. The base member need not be level zero. You can enter data in shared members, and values are stored with base members.

Shared members are displayed similarly to base members in the dimension hierarchy for member selection in Oracle Smart View for Office.

Creating Shared Members

You create shared members the same way as other members, with these differences:

- The base member can't be the parent of the shared member.
- You can't add a shared member as a sibling to the base member.
- You must give the shared member the same name as its base member. It can have a different description.
- You must select Shared as the Data Storage for the shared member.

Working with Attributes

Use attributes to group members using the same criterion. You can assign attributes to sparse dimensions only. You can't assign attributes to label-only members. Attribute dimensions don't have aggregation properties because parents are dynamically calculated.

The Account dimension is usually defined as dense, so you can't assign attributes to it unless it's changed to sparse for all cubes. If you change a dimension from sparse to dense, all attributes and attribute values for that dimension are automatically deleted.

Attributes can have data types of text, date, Boolean, and numeric, as described in Understanding Attribute Data Types. Attribute names must conform to guidelines listed in Naming Restrictions. When attributes are defined, you can use the Member Selection dialog box to select attribute functions, such as Equal and GreaterOrEqual.

To create and change attributes, attribute values, and aliases:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. Select a sparse dimension for which to define an attribute, attribute value, or alias.
3. Click Note:
   Only sparse dimensions can contain attributes.

4. Select options.
   - To create attributes, click Note: Type an attribute name, and select a data type: Text, Date, Boolean, or Numeric. See Understanding Attribute Data Types. You can't modify the data type after the attribute is created.
     Cube options are available for Entity dimension attributes only. You can't change this setting after the attribute is created.
   - To modify attributes, click , and update the attribute name.
   - To set aliases for attributes, select an attribute and an attribute value, click . Select an alias table, type an alias name, and click Save.

5. Click Close.
   When you click Close, the hierarchy is validated and an error displays if issues are detected. For example, date attribute values must be entered in the correct format, and numeric and date attribute dimensions must have at least one attribute value defined.

6. Update and validate business rules and reports.
   - Understanding Attribute Data Types
   - Deleting Attributes

Note:
Instead of explicitly filtering by an attribute (like Red, for instance), you can create a user variable for the attribute dimension, and then use the user variable as the filter. Then you can enable the user variable as a dynamic user variable which would allow users to change the value of the filter at runtime. This is a useful technique that allows for dynamic filtering. See Managing User Variables.

Understanding Attribute Data Types

Attribute dimensions can have a data type of text, numeric, Boolean, or date that enables different functions for grouping, selecting, or calculating data. The attribute type applies only to level 0 members of the attribute dimension.
• Text attributes enable basic attribute member selection and attribute comparisons in calculations. When you perform such comparisons, characters are compared. For example, a package type Bottle is less than a package type Can because B precedes C in the alphabet.

• Numeric attribute dimensions use numeric values for the names of level 0 members. You can include the names (values) of numeric attribute dimension members in calculations. For example, you can use the number of ounces specified in an Ounces attribute to calculate profit per ounce for each product. You can also associate numeric attributes with ranges of base dimension values, for example, to analyze product sales by market population groupings.

• Boolean attribute dimensions in a database contain only two members. When a Boolean attribute dimension is added in Planning, two attribute values, True and False, are created for this attribute dimension by default. A base dimension, such as Account or Entity, can be associated with only one attribute dimension that has the Boolean data type.

• Date attributes can specify the date format as month-day-year or day-month-year, and sequence information accordingly. You can use date attributes in calculations, for example, comparing dates in a calculation that selects product sales since 12-22-1998. Users can set the date format by selecting an option in Attribute Dimension Date Format in Application Settings.

Deleting Attributes

When you delete an attribute, all attribute values associated with the attribute are also deleted. Attribute values are removed from members to which they had been assigned, and the attribute is removed from dimensions to which it was assigned.

To delete attributes:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. Select the sparse dimension for which to delete an attribute, and click Delete.
3. Select the attribute to delete.
4. Above the Attributes column, click Delete.
5. Click OK.
6. Update and validate business rules and reports.

Working with Attribute Values

Attribute values provide users with another way of selecting dimensions members when using forms. Data values for attribute values are dynamically calculated but not stored.

• Creating Attribute Values
• Assigning Attribute Values to Members
• Editing and Deleting Attribute Values
Creating Attribute Values

You can define attribute values for sparse dimensions, which are typically the Entity and user-defined custom dimensions. After you define an attribute value for a dimension, you can assign it to members of that dimension.

To create attribute values:

1. From the Home page, click Navigator and then under Create and Manage, click Dimensions.
2. Select the sparse dimension for which to create an attribute value.
3. Select the top level in the dimension hierarchy.
4. Click .
5. On the Manage Attributes and Values page, select the attribute for which to specify a value.
6. Above the Attribute Values column, click . If the options are available, you can click or .
7. Enter a name and choose a data type.
8. Click Save.

Assigning Attribute Values to Members

You can assign attribute values members of a dimension that are defined as sparse for all cubes. Attribute values must be assigned to the same-level sparse dimension members. Otherwise, errors display during refresh.

To assign attribute values to members:

1. From the Home page, click Navigator and then under Create and Manage, click Dimensions.
2. Select the sparse dimension for whose member you want to assign an attribute value.
3. In the Dimension hierarchy, select a member to which to assign an attribute value.
4. Click .
5. For members assigned attribute values, click to change a member’s attribute value.
6. Select attribute values to assign to the member.
7. Perform an action:
   - To assign the value to the selected member, click .
   - To remove a value from the selected member, select the value to remove and click .
• To remove all values from the selected member, click \( \text{\textup{\textbackslash}} \).

8. Click \textbf{Save}.


table

<table>
<thead>
<tr>
<th>Note:</th>
</tr>
</thead>
<tbody>
<tr>
<td>When you delete an attribute value, it's removed from custom dimension members to which it's assigned.</td>
</tr>
</tbody>
</table>

table

To edit or delete attribute values:

1. From the Home page, click \textbf{Navigator} and then under \textbf{Create and Manage}, click \textbf{Dimensions}.
2. Select the sparse dimension containing the attribute value you want to modify or delete.
3. Select the top level in the dimension hierarchy.
4. Click \( \text{\textup{\textbackslash}} \).
5. For \textbf{Attributes}, select the attribute containing the value to modify or delete, and then select the attribute value.
6. Above \textbf{Attribute Values}, click \( \text{\textup{\textbackslash}} \) or \( \text{\textup{\textbackslash}} \).
7. If editing, enter a name. If deleting, confirm the deletion.
8. Click \textbf{Save}.
9. Update and validate business rules and reports.

Customizing Calendars

Use the Period dimension to work with the yearly calendar rollup structure. When creating the application, the administrator specifies the base time periods that span the application database. Use the Years dimension to add years to the calendar.

Subtopics:

- Defining How Calendars Roll Up
- Creating and Editing Summary Time Periods
- Deleting Summary Time Periods
- Working with the Years Dimension
- Adding Years to the Calendar
- Editing Year Information
- Renaming Time Periods
- Assigning Aliases to Summary Time Periods
Defining How Calendars Roll Up

Table 18-8  Calendar Roll Up

<table>
<thead>
<tr>
<th>Base Time Period</th>
<th>Roll Up</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 Months</td>
<td>Four quarters are created per year. Months roll up into parent quarters and quarters roll up into years.</td>
</tr>
<tr>
<td>Quarters</td>
<td>Quarters roll up into years.</td>
</tr>
<tr>
<td>Custom</td>
<td>No default rollup structures. A flat list of the custom base time periods displays.</td>
</tr>
</tbody>
</table>

After the application calendar is created, you can't change the base time period or reduce the number of years in the calendar. Administrators can change the names, descriptions, aliases, and ranges of the summary time periods in the hierarchy.

Creating and Editing Summary Time Periods

You can change such aspects as name, description, alias, starting period, and ending period. However, you can't change the order of base time periods or skip base time periods. The range can't extend beyond the current fiscal year.

You must work from the top of the hierarchy to the bottom when creating summary time periods. (Otherwise, the application views the rollup structure as asymmetrical and you can't continue.) The summary time period displays in the hierarchy as a parent of the selected item. To enforce a balanced hierarchy, all base members must be the same number of levels from the root.

To create or edit summary time periods:

1. From the Home page, click Navigator, and then under Create and Manage, click Dimensions.
2. For Dimension, select Period.
3. Perform one action:
   • To add a time period, select the time period in the dimension hierarchy to which you want to add a child or sibling, and then click or .
   • To edit a time period, select the time period and click .
4. For Name, enter or change the name for the summary time period.
5. Optional: For Description, enter a description.
6. Optional: For Alias Table and Alias, select an alias table to use. Enter an alias name.
   The default table is used if you don't select one.
7. For Start Period, select the starting period.
The range can't extend beyond the current fiscal year. For summary time periods, **Start Period** displays the first child, or all children except the first child of the sibling above it.

8. For **End Period**, select the ending period.
   For summary time periods, **End Period** displays the last child, or all children from the Start Period through the next sibling’s children, except the last child.

9. Click **Save**.

Deleting Summary Time Periods

When you remove a summary time period from the hierarchy, its children are moved into another summary time period:

- If you delete the first summary time period, children are moved into the next sibling of the summary time period.
- If you delete the last summary time period, children are moved into the previous sibling of the summary time period.
- If you delete a summary time period from the middle of a hierarchy, children are moved into the previous sibling of the summary time period.

To delete summary time periods:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Dimensions**.
2. For **Dimension**, select **Period**.
3. In the dimension hierarchy, select the summary time period to delete.
   You can't delete base time periods.
4. Click ✗.
5. Click **OK**.

Working with the Years Dimension

Use the Years dimension to work with calendar years.

**Table 18-9  Years Tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>See Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Add years before the Start year or after the End year of the calendar.</td>
<td>Adding Members in the Simplified Dimension Editor</td>
</tr>
<tr>
<td>- Add an All Years parent member that includes all members of the Years dimension (except No Year, if that member exists).</td>
<td></td>
</tr>
<tr>
<td>Add or update the description and alias for a year.</td>
<td>Editing Members in the Simplified Dimension Editor</td>
</tr>
</tbody>
</table>
Adding Years to the Calendar

You can add years to the beginning or end of the calendar, but you can't reduce the number of calendar years without creating a database.

To add years to the calendar:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. For Dimension, select Years.
3. Click .
4. In Add Years, enter the number of years to add to the calendar.
5. Click OK.

The confirmation dialog asks you to confirm the following:

- To add years after the End year, click Yes.
- To add years before the Start year, click No.

Note:

To add an All Years parent member that includes all members of the Years dimension, click . The All Years parent member enables users to view the accumulated data across multiple years, for example, a project's total cost up to its end date. (This parent member doesn't include the No Year member, if one exists.)

Editing Year Information

You can add or update the description and alias for a member of the Years dimension.

To edit a member of the Years dimension:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. For Dimension, select Years.
3. Select the year you want to edit, and then click .
4. Update the information for that year, and then click Save.
Renaming Time Periods

You can rename root-level, base time periods, and user-defined summary time periods.

To rename time periods:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. For Dimension, select Period.
3. Select the time period you want to rename, and then click .
4. Rename the time period, and then click Save.

Assigning Aliases to Summary Time Periods

You can assign and change aliases to base time periods and summary time periods.

To assign or change the alias:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. For Dimension, select Period.
3. Select the summary time period.
4. Click .
5. For Alias Table, select the alias table to use.
6. For Alias, enter an alias name.
7. Click Save.

Editing the BegBalance Member

You can edit the BegBalance member of the Period dimension. As the first time period in the application, the BegBalance member is useful for entering beginning data when you start a new application, fiscal year, or calendar year. You can rename and describe BegBalance and give it an alias.

To edit the BegBalance member:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. For Dimension, select Period.
3. Select the first member, BegBalance.
4. Click .
5. For Edit Period:
   • Enter a name.
• Enter a description.
• Select an alias table to use for the BegBalance member, and enter an alias.

6. Click **Save**.

## Setting Up Currencies

You can plan, forecast, and analyze financial information in one or more currencies. You can create, edit, and delete currencies. Administrators control:

• Which currencies an application uses, including for reporting
• How currencies display in reports and forms
• How currencies convert to other currencies
• For Standard multiple currency applications, whether a triangulation currency converts currencies
• When currency conversions occur

## Enabling Multiple Currencies

In a Standard multiple currency application only, you can enable multiple currencies per entity on forms. See **Defining the Layout**.

When selecting business rules for forms, you can select the Calculate Currencies business rule to convert values among the available currencies. See **Adding and Removing Business Rules in Forms**.

• **Working with Currencies in a Standard Multicurrency Application**
• **About the Calculate Currencies Business Rule**
• **Exchange Rate Types**
• **Scaling**
• **Number Formatting**
• **Reporting Currencies for Standard Multicurrency Applications**
• **Checking How Currencies are Used**
• **Creating Currencies**
• **Editing Currencies**
• **Deleting Currencies**

## Base Currency

For a Standard multiple currency application, specify each entity member’s base currency. The default base currency for entity members is the currency specified when creating the application. For example, if U.S. Dollars is the default currency, you may specify Yen as the base currency for the Japan entity and U.S. Dollars for the United States entity. When using forms having values for the Japan entity, if the display currency is set to U.S. Dollars, values are converted to U.S. Dollars using the rates in the exchange rate table (assuming Yen is the local currency and U.S. Dollars is the reporting currency).
Note:

If the Simplified multiple currency option was selected during application creation, you don't need to specify a base currency for entity members. See About Simplified Multicurrency.

Working with Currencies in a Standard Multicurrency Application

Note:

This topic assumes the Standard multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

If multiple currencies are enabled, users can see values converted from the local currency to a reporting currency and can override a cell's base currency.
**Note:**

- When the local currency is selected on forms, the default stored and displayed currency for cells is the entity’s base currency (which you specify). Users can enter data values only into local currency members. If the local currency member is selected, all currencies specified for the application are available as input types.

- You can set dimension properties for each currency in the **Edit Currency** dialog box. In preferences, users can select different display options, and can select Currency Setting to apply the properties set by the administrator.

- Currencies can be converted only to reporting currencies. Users can’t enter data into cells displayed in reporting currencies. The application’s main currency is by default a reporting currency. You can change which currencies are reporting currencies.

- Currencies defined for the application are valid currencies for data entry. Valid currencies for data entry are displayed in a list that users access by clicking the Currency link during data entry.

- To get meaningful results, roll up values in one common reporting currency. If members of a subtotal have mixed currencies, the currency type is blank and the currency symbol doesn’t display. For example, adding 10 US dollars and 10 Japanese yen to a value of 20 makes no sense.

- An application with 500 time periods can successfully run currency conversion calculation scripts only if the time periods have default names, TP 1 through 500. Otherwise, the conversion calculation script you try to create exceeds the 64K limit.

- User-defined currency conversion calculation scripts created when a database is created or refreshed may be available in Oracle Smart View for Office, depending on user access. When using user-defined currency conversion calculation scripts, Oracle recommends changing the order so the currency conversion calculation script is first, before Calculate Form.

- Currency codes associated with input values are stored as numeric values. These codes are calculated in dimension formulas, calculation scripts, and business rules. The calculated values of these currency codes may translate to currency codes that are incorrect or invalid. Where there are children with mixed currencies, review calculated results on the upper levels.

- If a parent has multiple children, of whom only one child has an overridden currency, the parent inherits the overridden currency code (which isn’t displayed on forms).

- In certain cases, parent entities display #MISSING when trying to convert to a selected currency. Ensure that a currency rate is entered for each combination of local currencies and selected currencies on forms or reports. Currency combinations must exist for all mixed-currency children entities and parent members.
Input of multiple currencies to one entity isn't supported in Smart View. If worksheets include mixed currency types, users could inadvertently enter values in the wrong currency.

About the Calculate Currencies Business Rule

Note:
This topic assumes the Standard multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

The Calculate Currencies business rule is based on the dimensions and members on the form. It converts data from the local currency to the reporting currency specified on the form, applying the exchange rate conversions. It:

• Doesn't calculate subtotals. To subtotal values, run the Calculate Form business rule (or a customized business rule that includes aggregation) after converting currencies.
• Ignores #MISSING values.
• Can be turned on or off by associating or disassociating it with forms during form design.
• Is set by default to not run when saving data.

Exchange Rate Types

Exchange Rate Types for Standard Multicurrency Applications

For Standard multiple currency applications, these are the exchange rate types that are associated with currencies: Historical, Average, and Ending.

The exchange rate type for each account is specified in the Member Property dialog box. For average and ending rate types, enter values for all time periods. For historical rate types, enter one rate value that is used for all time periods, including the Beginning Balance period. For the Beginning Balance period, enter one rate value used for that time period for average and ending rate types.

A Standard multiple currency application supports currency conversion by triangulation through a triangulation currency.

Exchange Rate Types for Simplified Multicurrency Applications

For Simplified multiple currency applications, these are the exchange rate types that are associated with currencies: FX Rates-Average and FX Rates-Ending. See About Simplified Multicurrency.

Scaling

You can specify scaling data values when displayed in certain currencies. For example, you can set the scaling for Yen to Thousands, then enter 10,000 as a value for the
Japan entity on a form with the Local member selected for the Currency dimension. When you select Yen as the currency member for the form, the scaling is applied and 10 displays as the value for Japan.

**Number Formatting**

You can determine the initial display of numerical values for non-currency and currency data types in forms:

- **Thousands separator:**
  - None—1000
  - Comma—1,000
  - Dot—1.000
  - Space—1 000

- **Decimal separator:**
  - Dot—1000.00
  - Comma—1000,00

- **Negative number sign:**
  - Prefixed minus—-1000
  - Suffixed minus—1000-
  - Parentheses—(1000)

- **Negative number color:**
  - Black
  - Red

**Reporting Currencies for Standard Multicurrency Applications**

> **Note:**

This topic assumes the Standard multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

A reporting currency is the currency in which your company prepares financial statements. The application supports currency conversion from local currencies to one or more reporting currencies. Converted reporting currency values are stored and read-only for all users. An application’s default currency is the default reporting currency. You can disable a currency as a reporting currency.

**Checking How Currencies are Used**

You can view how an application uses currency: whether a currency is the default, is used for triangulation currency or by an entity (for Standard multicurrency applications), or has a conversion or exchange relationship with other currencies.

To see how currencies are used:
1. From the Home page, click **Application**, and then click **Overview**.
2. Click the **Dimensions** tab, and then select the Currency dimension.
3. Select the currency for which you want information.
4. Click **Actions**, and then click **Show Usage**.

**Creating Currencies**

Select from a predefined list or create your own. You can specify:

- The three-letter code
- The symbol
- A description of up to 256 characters
- The scaling factor to use when values are displayed
- For Standard multicurrency applications, the triangulation currency to use for currency conversion
- The alias table to use to display aliases
- Number formatting, including thousands separator, decimal separator, negative sign, and color
- Whether it's a reporting currency

To create currencies:

1. From the Home page, click **Application**, and then click **Overview**.
2. Click the **Dimensions** tab, and then select the Currency dimension.
3. Create a currency:
   - To add a predefined currency, click **Actions**, then click **Add Standard Currency**, select from a list of standard currencies, and then click **OK**.
   - To create a currency, click **Actions**, then click **Add Currency** and specify properties:
     - For **Member Name**, enter an abbreviation or identifier of up to three characters.
     - Optional: For **Description**, enter a name, such as Japanese yen.

**Note:**

To view all property columns in the grid, including the **Description** column, right-click any heading on the grid and scroll down until you see the **Default mode** check box. Clear the **Default mode** check box to view all the property columns in the grid.

- For **Symbol**, enter a symbol, or select a symbol from the **Predefined Symbol** column.
- **Optional:** For **Scale**, select how to enter and display the currency. For example, 3 yen represents 3000 yen if scaling is set to thousands.
– **Optional**: For **Triangulation Currency** (for Standard multicurrency applications), select the currency to use as the common third currency for conversion.
– **Optional**: For **Alias Table**, select the alias table to use.
– **Optional**: For **Alias**, enter a name for the currency alias.

4. **Optional**: Select **Reporting Currency** (see Working with Currencies in a Standard Multicurrency Application).

5. **Optional**: For **Thousands Separator**, select how to display the thousands separator (it must differ from the decimal separator).

6. **Optional**: For **Decimal Separator**, select how to display numbers with decimal values (it must differ from the thousands separator).

7. **Optional**: For **Negative Sign**, select how to display negative numbers:
   - Prefixed minus— -1000
   - Suffixed minus—1000-
   - Parentheses—(1000)

8. **Optional**: For **Negative Color**, select the display color.

9. **Optional**: Select the type of **Data Storage**.

10. **Optional**: Select **Two Pass Calculation**.

11. **Optional**: Select the **Data Type**.

12. **Optional**: Select a **Smart List**.

13. Click **Save**.

**Editing Currencies**

To edit currencies:

1. Click the **Navigator** icon ☞, and then under **Create and Manage**, click **Dimensions**.

2. From **Dimension**, select **Currency**.

3. Select the currency to edit.

4. Click ✍.

5. Click the **Currency** tab.

6. Modify properties:
   - To select from the predefined symbols, select one from the **Select from Predefined Symbols** drop-down list.
   - To change the currency's symbol, for **Symbol**, enter or select the symbol.
   - For **Scale**, set how to enter and display the currency.
   - For set currency precision (the number of digits to the right of the decimal place), select a number from 1 to 10 from the **Precision** drop-down list. **None** is the default.
• To specify the currency as a reporting currency, select **Reporting Currency**. See *Working with Currencies in a Standard Multicurrency Application*.
• For **Thousands Separator**, select how to display the thousands separator (it must differ from the decimal separator).
• For **Decimal Separator**, select how to display numbers with decimal values (it must differ from the thousands separator).
• For **Negative Sign**, select how to display negative numbers:
  - **Prefixed Minus**— -1000
  - **Suffixed Minus**—1000-
  - **Parentheses**—(1000)
  - **Use Default Setting**—Apply the display setting for the currency (see *Creating Currencies*).
• For **Negative Color**, select the display color.

7. Click **Save**.

**Deleting Currencies**

You can't delete the default currency.

To delete currencies:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Dimensions**.
2. From **Dimension**, select **Currency**.
3. Select the currency to delete.
4. For multiple currency applications, click to determine if the currency is the default currency. If a Standard multicurrency application is used, determine if the currency is a triangulation currency, or if the currency is associated with an entity. You can't delete a currency that meets these criteria.

**Note:**

For Standard multiple currency applications, if you delete a currency defined in the exchange rate table, it's deleted from the table.

5. Click **Close**.

6. Click , and then click **OK**.
7. Update and validate business rules and reports.

Specifying Exchange Rates

Note:
This topic assumes the multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

Use exchange rates to convert values from one currency to another. You can:

- Enable budget preparers in various countries to create plans in other currencies
- Show summary report data in a currency
- Summarize values from multiple currencies into one currency

For example, you might specify yen as the base currency for the Japan entity and US dollars for the United States entity. When you display a form having values for the Japan entity and the form’s display currency is set to US dollars, the exchange rates for the yen is used to convert the values for Japan to US dollars. If the display currency is set to yen, the exchange rates for US dollars converts values for the United States entity to yen.

To specify exchange rates, you must set up multiple currencies when creating an application.

- About Exchange Rate Tables for Standard Multicurrency Applications
- Hsp_Rates Dimension for Standard Multicurrency Applications
- Triangulation
- Calculation Method

About Exchange Rate Tables for Standard Multicurrency Applications

Note:
This topic assumes the Standard multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

Each application has a default currency specified when the application is created. When you specify exchange rate tables, only the default currency and triangulation currencies are available as destination currencies. You can enter exchange rates from source currencies to default or triangulation currencies.

You can create multiple exchange rate tables. Each table is typically associated with multiple scenarios, but each scenario can be associated with only one exchange rate table. When creating scenarios, select the exchange rate table for converting currencies.
Enter conversion values between the default currency and currencies defined in the Exchange Rates page. Exchange rate tables span all application time periods, so you can apply exchange rates to all scenarios. When creating or modifying exchange rate tables, you must refresh the application to store them in the cubes. See Refreshing Application Databases.

Hsp_Rates Dimension for Standard Multicurrency Applications

Note:
This topic assumes the Standard multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

A Standard multiple currency application includes the Hsp_Rates dimension for storing exchange rates. It includes these members and others that store currency rates:

- Hsp_InputValue—Stores data values
- Hsp_InputCurrency—Stores currency types for data values

When generating reports or loading data, refer to the Hsp_InputValue member. When loading data, you must load data against the local currency. You need not refer to the Hsp_InputCurrency member.

By default, the Hsp_Rates dimension is set to Sparse. You can change this (see Setting Dimension Density and Order).

Triangulation

Note:
This topic assumes the multiple currency option was selected during application creation. If the Simplified multiple currency option was selected, see About Simplified Multicurrency.

A multiple currency application supports currency conversion by triangulation through an interim currency called the triangulation currency. If you modify a currency’s triangulation currency, you must re-enter exchange rates for the triangulation currency property and refresh the application to transfer and store the exchange rates. You can't select the application’s default currency as a triangulation currency.

Calculation Method

When you input exchange rates for converting between currencies, you can select Multiply or Divide as the calculation method. For example, if you select 1.5 as the rate for converting British Pounds to US dollars, and select multiply as the calculation method, 1 British Pound is converted to 1.5 US dollars.
Setting up Dynamic Time Series Members

You can use Dynamic Time Series members to create reports that show period-to-date data, such as quarter-to-date expenses. Dynamic Time Series members are created automatically during application creation, and can be used with members of the Period dimension. To set up Dynamic Time Series, you enable a predefined Dynamic Time Series member and associate it with a generation number (and, optionally, an alias table and alias name). For example, to calculate quarter-to-date values, you can enable the Q-T-D (quarter-to-date) member and associate it with generation number 2. You can then use the Q-T-D Dynamic Time Series member to calculate monthly values up to the current month in the quarter.

Note:

Dynamic Time Series isn't supported for the Period dimension in an aggregate storage application.

The Dynamic Time Series members provide up to eight levels of period-to-date reporting. Your data and database outline determine which members you can use. For example, if the database contains hourly, daily, weekly, monthly, quarterly, and yearly data, you can report D-T-D (day-to-date), W-T-D (week-to-date), M-T-D (month-to-date), Q-T-D, and Y-T-D information. If the database contains monthly data for the past 5 years, you can report Y-T-D and H-T-D (history-to-date) information, up to a specific year. If the database tracks data for seasonal time periods, you can report P-T-D or S-T-D (season-to-date) information.

Oracle recommends that you avoid assigning time balance properties (such as First and Average) to members set for dynamic calculations if you plan to use the members in Dynamic Time Series calculations. Doing so may retrieve incorrect values for parent members in your accounts dimension.

To set up Dynamic Time Series members:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. Select the Period dimension, and then click .
3. Select Enabled for the DTS series to use:
   - Y-T-D: Year-to-date
   - H-T-D: History-to-date
   - S-T-D: Season-to-date
   - Q-T-D: Quarter-to-date
   - P-T-D: Period-to-date
   - M-T-D: Month-to-date
   - W-T-D: Week-to-date
   - D-T-D: Day-to-date
4. Select a generation.
   The number of generations displayed depends on the number of generations in
   the time dimension. You can't associate Dynamic Time Series members with the
   highest generation (the dimension root).

5. **Optional**: Add an alias name. In the **Edit Alias Name** column, click 
   ![Edit Alias](edit_aliase.png), enter an
   alias name, and then click **OK**.

6. Click **Save**.

### Additional Supported Application Features

Additional features are supported for application dimensions. You can add children
and siblings to Scenario, Version, and Period dimensions, and you can use the cut,
paste, expand, and collapse features to work with their dimension hierarchies (see
[Working with Dimension Hierarchies](#)). You can also use shared members for these di‐
dimensions, and can set two pass calculations at the root level for all dimensions. For
example, you can:

**Table 18-10  Features**

<table>
<thead>
<tr>
<th>Feature</th>
<th>More Information</th>
</tr>
</thead>
</table>
| In Scenario and Version dimensions, create hi‐
erarchies and use shared members. If you as‐
sign children to bottom-up versions, these ver‐
sions display as read-only parents on forms. | See [Setting Up Scenarios](#) and [Specifying Ver‐
sions](#). |
| In the Period dimension, create alternate hier‐
erarchies and use shared descendants. Data Storage for all time periods can be set to any valid Data Storage value. The Consolidation operator for all time periods, including BegBa‐
lance, can be set to any valid consolidation op‐
erator. For example, it can be set to + instead of − (ignore). | See [Working with the Years Dimension](#) and [Editing the BegBalance Member](#). |
| Turn on two pass calculation at the root level, for example, for Account. | See [Adding or Editing Members](#). |

**Caution:**

Two pass calcu‐
lation is ignored
on any non-Ac‐
count member
not set to Dy‐
namic Calc.
When using this
setting, consider
the impact on
currency conver‐
sion scripts.

For attributes, create hierarchies and assign aliases. | See [Working with Attributes](#). |
Considerations for Alternate Hierarchies in Period Dimensions

If you create an alternate hierarchy in the Period dimension, the alternate hierarchy must follow the YearTotal member in the outline.

Working with UDAs

You can use user-defined attributes (UDAs), descriptive words or phrases, within calc scripts, member formulas, and reports. UDAs return lists of members associated with the UDA. For example:

• You can use the HSP_UDF UDA to prevent a formula from being overwritten when the application is refreshed. You must log on to each database associated with the Planning application and create the member formula with a UDA. The syntax for this UDA is: (UDAs: HSP_UDF).

• If you use the @XREF function to look up a data value in another database to calculate a value from the current database, you can add the HSP_NOLINK UDA to members to prevent the @XREF function from being created on all cubes that are not the source cube selected for that member.

• For a Product dimension with several product members, you can create a UDA called New Products and assign this UDA to the new products in the Product dimension hierarchy. Then you can base certain calculations on the designation New Products.

• For the budget review process, rather than creating a data validation rule for each owner in a product line (some product lines can have hundreds of owners), you can create a UDA containing the user names that apply to members using the approval unit hierarchy. Then in the data validation rule, you can enter a lookup function which will return the user names stored in the UDA for the current member. For example, create a UDA for each user in the promotional path and assign a prefix to the UDA name (for example, ProdMgr:Kim).

UDAs are specific to dimensions. For example, creating a UDA for an Account member makes it available for non-shared Account members. Deleting it removes it for all Account members. To make UDAs available for multiple dimensions, create the same UDA for multiple dimensions. For example, create a UDA named New for Account and Entity dimensions to make the UDA named New available for Account and Entity members.

To work with UDAs:

1. Click the Navigator icon, and then under Create and Manage, click Dimensions.
2. Select the dimension for whose members to associate the UDA.

3. From the dimension hierarchy, select a member and click .

4. Select UDA.

5. Select a task:
   - To create a UDA, click , enter a name, and then click Save.
   - To modify a UDA, click , change the name, and then click Save.
   - To delete a UDA, select the UDA, click , and then confirm deletion.

   **Note:**
   If creating a UDA for approvals, assign a prefix to the beginning of the name (for example, ProdMgr:Name). A prefix indicates that the UDA contains a user name and enables the data validation rule to look up the user. Use the same prefix for all approvals UDAs.

   - To clear UDA selections, click .

6. To add or remove UDAs for the member, use the arrows to move UDAs to and from the Selected UDA panel.

7. Click Save.

**Working with Member Formulas**

You can define member formulas to combine operators, calculation functions, dimension and member names, and numeric constants to perform calculations on members. Member formulas can also include:

- Operator type, function, value, member name, UDA, and so on allowed in formulas.
- Predefined formula expressions, including Smart List values, that expand into a formula or value upon database refresh.

To define member formulas:

1. Click the Navigator icon , and then under Create and Manage, click Dimensions.
2. Select the dimension for whose member to add or change a formula.
3. Select the member and click Edit.
4. Select the Member Formula tab.

5. Select options for the following fields:
   - **Cube**

<table>
<thead>
<tr>
<th><strong>Note:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>A formula entered for the default cube will be applied to all cubes unless it's overridden by a different formula entered for a specific cube.</td>
</tr>
</tbody>
</table>

   - **Data Storage**—Select a data storage option. The default is Store.

<table>
<thead>
<tr>
<th><strong>Note:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>The cube specific data storage field will not display the Shared or Label Only options. This is because a member can't be set to Shared or Label Only in one cube and not another.</td>
</tr>
</tbody>
</table>

   - **Solve Order**—For aggregate storage cubes only, solve order specifies the order in which formulas are evaluated. Enter a whole number between 0 and 100000 (or use arrows to increase or decrease the number). The formulas for members that have a specified solve order are calculated in order from the lowest solve order to the highest. The default is 0.

6. In the text box, define formulas for the member.

7. **Optional:** To check the validity of the member formula, click Validate.

8. Click Save.

   Before you click Save, clicking Reset restores the previous member formula information.

### Viewing Details of Formula Validation

To view details of the member formula validation:

1. On Member Formula, click Validate.
2. Click Save.

### Working with Formula Expressions

In the text box on the Member Formula tab, you can include predefined formula expressions in member formulas, and test them with the Validate button. You can also load them.

You can update the dimension outline without updating the business rules and calc scripts that depend on the outline. Calculations become more independent of specifics in the outline. You can use Smart Lists as objects in calculations. Performance isn't decreased when you use formula expressions because they are run only when you refresh the database.

To use a formula expression in a member formula:
1. Click the Navigator icon, and then under Create and Manage, click Dimensions.

2. Select the dimension with which you want to work.

3. Select the member and click Edit.

4. Select Member Formula.

5. Select options for the following fields:

   - **Cube**
     
     **Note:**
     
     A formula entered for the default cube will be applied to all cubes unless it's overridden by a different formula entered for a specific cube.

   - **Data Storage**—Select a data storage option. The default is Store.
     
     **Note:**
     
     The cube specific data storage field will not display the Shared or Label Only options. This is because a member can't be set to Shared or Label Only in one cube and not another.

   - **Solve Order**—For aggregate storage cubes only, solve order specifies the order in which formulas are evaluated. Enter a whole number between 0 and 100000 (or use arrows to increase or decrease the number). The formulas for members that have a specified solve order are calculated in order from the lowest solve order to the highest. The default is 0.

6. In the text box, define formulas for the member.

   You can include Planning formula expressions and Essbase native formulas in the member formula.

   The application provides predefined formula expressions that you can use in member formulas. You can't edit or create your own formula expressions.

7. **Optional:** To check the validity of the member formula, click Validate.

8. **Optional:** If there are errors in the member formula, click Show Details to view a description.

9. **Optional:** Click Reset to restore the previous member formula if you don't want to save the changes you made to the member formula.

10. Click Save.

**Syntax**

Member formula expressions support functions and variables. Follow these syntax rules for functions and variables when you create formula expressions:

- Enclose variables or properties with square brackets, [ ]. If you omit square brackets, the variable is treated as a native variable.
• Enclose member names with quotation marks.
• Characters in variables are case-insensitive, but can't have extra spaces or characters such as underscore ( _ ).
• You can include subcalls to other functions within an expression.
• Don't enter text where a number is required.
• The order of the outline is important in a member formula. For example, don't reference a value that has not been calculated yet.

Including Smart List Values as Variables

You can include a Smart List as a variable in a formula expression, such as the formula expression, "Status"= [Status.Departed].

"Status" is the member name, Status is the Smart List name, and Departed is a Smart List entry. If the Smart List ID for Departed is 2, Status.Departed is replaced with a 2 in the member formula (the application treats Smart Lists as numbers). If the Smart List ID for Departed is 2, 2 is put in the calculation and 2 is stored in the database.

Write Smart Lists in this format: [SmartListName.SmartListEntry]

Formula Expressions

Formula expressions can include these predefined variables and functions.

Table 18-11 Variables in Formula Expressions

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenInputValueBlock</td>
<td>Generates an IF statement if the application is a multicurrency application, or an empty string if it's one currency application. Used with ClosedInputValueBlock.</td>
</tr>
<tr>
<td>CloseInputValueBlock</td>
<td>Generates an End IF statement if the application is a multicurrency application, or an empty string if it's one currency application. Used with OpenInputValueBlock.</td>
</tr>
<tr>
<td>NumberOfPeriodsInYear</td>
<td>Returns the number of time periods in the year</td>
</tr>
<tr>
<td>NumberOfYears</td>
<td>Returns the number of years in the application</td>
</tr>
</tbody>
</table>

Table 18-12 Functions in Formula Expressions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension(dimTag)</td>
<td>Returns the name of a predefined dimension. The dimtags are:</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_PERIOD</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_YEAR</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_ACCOUNT</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_ENTITY</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_SCENARIO</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_VERSION</td>
</tr>
<tr>
<td></td>
<td>• DIM_NAME_CURRENCY</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Period(periodName)</td>
<td>Returns the specified period. The periodName options are:</td>
</tr>
<tr>
<td></td>
<td>• FIRST_QTR_PERIOD</td>
</tr>
<tr>
<td></td>
<td>• SECOND_QTR_PERIOD</td>
</tr>
<tr>
<td></td>
<td>• THIRD_QTR_PERIOD</td>
</tr>
<tr>
<td></td>
<td>• FOURTH_QTR_PERIOD</td>
</tr>
<tr>
<td></td>
<td>• FIRST_PERIOD</td>
</tr>
<tr>
<td></td>
<td>• LAST_PERIOD</td>
</tr>
<tr>
<td>CrossRef(accountName)</td>
<td>Generates a cross-reference by adding the default prefix of &quot;No&quot; to each dimension name (except Currency, Period and Year), followed by the specified account.</td>
</tr>
<tr>
<td></td>
<td>For example, in an application with the following dimensions: Account, Period, HSP_View, Year, Scenario, Version, Entity, and Product</td>
</tr>
<tr>
<td></td>
<td>CrossRef(&quot;5800&quot;) returns: &quot;BegBalance&quot;-&gt;&quot;No HSP_View&quot;-&gt;&quot;No Scenario&quot;-&gt;&quot;No Version&quot;-&gt;&quot;No Entity&quot;-&gt;&quot;No Product&quot;-&gt;&quot;5800&quot;;</td>
</tr>
<tr>
<td>CrossRef(accountName, prefix)</td>
<td>Generates a cross-reference by adding the specified prefix to each dimension name (except Currency, Period and Year), followed by the specified account. The prefix should be in double quotes.</td>
</tr>
<tr>
<td></td>
<td>For example, in an application with the following dimensions: Account, Period, HSP_View, Year, Scenario, Version, Entity, and Product</td>
</tr>
<tr>
<td></td>
<td>CrossRef(&quot;5800&quot;, &quot;NoX&quot;) returns: &quot;BegBalance&quot;-&gt;&quot;NoX HSP_View&quot;-&gt;&quot;NoX Scenario&quot;-&gt;&quot;NoX Version&quot;-&gt;&quot;NoX Entity&quot;-&gt;&quot;NoX Product&quot;-&gt;&quot;5800&quot;;</td>
</tr>
<tr>
<td>CrossRef(accountName, prefix, true)</td>
<td>Generates a cross-reference by adding the specified prefix to each dimension name, including Year (except Currency and Period), followed by the specified account.</td>
</tr>
<tr>
<td></td>
<td>For example, CrossRef(&quot;5800&quot;, &quot;NoX&quot;, true) returns:</td>
</tr>
<tr>
<td></td>
<td>&quot;BegBalance&quot;-&gt;&quot;NoX HSP_View&quot;-&gt;&quot;NoX Year&quot;-&gt;&quot;NoX Scenario&quot;-&gt;&quot;NoX Version&quot;-&gt;&quot;NoX Entity&quot;-&gt;&quot;NoX Product&quot;-&gt;&quot;5800&quot;;</td>
</tr>
<tr>
<td>getCalendarTPIndex()</td>
<td>Generates a member formula that returns an index for the time period; the index is based on the calendar year.</td>
</tr>
<tr>
<td>getFiscalTPIndex()</td>
<td>Generates a member formula that returns an index for the time period; the index is based on the fiscal year.</td>
</tr>
<tr>
<td>CYTD(memberName)</td>
<td>Generates a calendar year-to-date formula for the member</td>
</tr>
</tbody>
</table>
Table 18-12  (Cont.) Functions in Formula Expressions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CYTD(memberName, calTplIndexName, fiscalTplIndexName)</td>
<td>Generates a calendar year-to-date formula for the member, and the time period index based on the calendar year and fiscal year. Use when members are renamed. The default member names are “Cal TP-Index” and “Fiscal TP-Index.”</td>
</tr>
</tbody>
</table>

Understanding Common Errors

Follow the rules of syntax carefully. If formula expression syntax contains errors, error messages are returned after you validate the member formula. To get information about error messages, review the formula details on the Member Formula tab. The most common error message is “Failed to execute.” This occurs when you use parameters in the expression incorrectly. These actions cause “Failed to execute” error messages:

- Entering the wrong number of parameters in the formula expression
- Misspelling member names, functions, or variable names
- Not surrounding member names with quotation marks
- Including numbers where text is required

Administering Forms

Create forms for entering data based on your planning needs.

Related Topics

- About Forms
- Understanding Implied Sharing in Forms
- Creating Simple Forms
- Designing Specific Types of Forms
- Working with Forms and Form Components
- Managing Forms and Folders

About Forms

Forms are grids for entering data. You can create simple forms to meet your needs and then you can use the simple forms and other artifacts to design dashboards to summarize the data.

See Designing Dashboards.
Form Components

Point of View

Select point of view (POV) dimensions such as year, scenario, and version to define the context for pages, rows, and columns. For example, if Scenario is Budget, all data displayed and entered in pages, rows, and columns is written to the Budget scenario dimension member. Each POV dimension is set to one member, which planners can't change.

To simplify a form’s POV, or better tailor it to the needs and roles of planners, you specify only relevant members or define user variables. See Defining Simple Form Page and Point of View and Managing User Variables.

Page Axis

Use the page axis to specify member combinations that may span dimensions so that users can work with data in smaller, logical views. Each page axis item can have members selected from different dimensions. Users see only members they can access.

You can specify multiple page drop-down lists, and select members using relationship functions or attributes. Switch between member sets by selecting them from the page axis.

You display member names or aliases on the page axis. You can specify the number of members in a page dimension that enables a search drop-down list on the data entry page, useful if dimensions contain many members. See Defining Simple Form Page and Point of View.

Rows and Columns

Rows and columns define the grid into which planners enter data. For example, you can assign Unit Sales to the row axis and January to the column axis. When planners access forms, they can enter data into the cell where the Unit Sales row intersects with the January column.

By default, forms have one set of rows and columns. You can add rows and columns to create asymmetrical combinations of members. See Creating Asymmetric Rows and Columns.

Attribute Dimensions

In addition to using attribute dimensions as filters, you can define attributes as unique dimensions within the application so that they can be placed on an axis other than
their associated base dimension. This enables you to perform cross-dimensional roll-ups across attribute members.

Attribute dimensions are optional on form grids and are listed separately on the Layout tab of the Form Designer. To use an attribute dimension in a form, drag the attribute dimension to the point of view, page, row, or column.

**Caution:**

Don't drag the attribute on any axis when using it as a filter as you'll cause a cross-dimensional reference which will have a performance impact. When using an attribute dimension as a filter, there is a form option that shows the associated attribute (just like there is an option to show alias). See Defining the Layout.

---

**Form Design Considerations**

**Forms and Cubes**

When you create a form, you associate it with a cube, which determines the form's valid members. For example, if you assign a form to the Revenue cube, you can add only accounts that are valid for the Revenue cube. Entered data is saved to the selected cube's database.
Note:

- You can't change a form's cube after assigning it.
- You can edit form accounts only if their source cube matches the form's cube.
- If you add an account to a form associated with a cube other than the account's source cube, the account is read-only on that form.

Forms and Permissions

Assign permissions to a form to determine which planners can modify its design (for example, layout and instructions) and input data. Users can edit forms only if they have permission to one secured dimension's member. For example, if planners have read-only permission to the Europe entity, the rows and columns that include the Europe entity are read-only. Planners can change data only for members to which they have write permission.

Forms and Currencies

For a single-currency application, all entities use the currency selected when the application was created. For a Standard multicurrency application, the selected Currency member on forms determines the currency in which values display. If the Currency member is Local, no currency conversion occurs, and planners can enter data in their native currency. If a currency member other than Local is selected, values are converted to the currency selected for that row or column, and the form is read-only. You can enter data in rows or columns that have Currency or Local as the selected member. See Designing Forms for Multiple Currencies.

Forms and Versions

For bottom-up versions, rows and columns with level 0 members allow data entry. Rows or columns set to a parent member are read-only. The point of view must also be set to the level 0 member to allow data entry on a bottom-up version. Target versions allow data entry in parent and children members.

Filtering Form Members by Attributes

You can select members by using attributes. For example, on the Entity dimension you can select members by a specific Region such as South. The resulting grid will only contain members that have the South attribute (for example, TX, NM, and so on). Values can be entered and saved into rows and columns filtered by attributes.

Forms and Shared Members

Because you can't select shared members individually, select them using a relationship function. For example, select an alternate functional rollup to include all members under that rollup. Planners can enter values in rows or columns that display shared members, and data is saved to the base members in the database.
Forms and Calculations

To optimize calculations, select row members using relationships (such as Descendants or Children) instead of selecting individual children. For example, calculating individual parent-level totals could take several passes, so use a relationship instead.

Understanding Implied Sharing in Forms

Some members are shared even if you don’t explicitly set them as shared. These members are implied shared members.

When an implied share relationship is created, each implied member assumes the other member’s value. The application assumes shared member relationships in these situations:

- A parent has only one child
- A parent has only one child that consolidates to the parent
- The data type of the parent is Label Only. The parent inherits the value of the first child, regardless of the child’s aggregation settings.

In a form that contains members with an implied sharing relationship, when a value is added for the parent, the child assumes the same value after the form is saved. Likewise, if a value is added for the child, the parent usually assumes the same value after a form is saved.

For example, when a calculation script or load rule populates an implied share member, the other implied share member assumes the value of the member populated by the calculation script or load rule. The last value calculated or imported takes precedence. The result is the same whether you refer to the parent or the child as a variable in a calculation script.

Creating Simple Forms

Table 18-13  Simple Form Creation Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Want to Know More?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define the layout, including:</td>
<td>See Defining the Layout.</td>
</tr>
<tr>
<td>• Adding rows and columns</td>
<td></td>
</tr>
<tr>
<td>• Assigning dimensions to columns and rows</td>
<td></td>
</tr>
<tr>
<td>• Assigning attribute dimensions to the point of view, page, columns, and rows</td>
<td></td>
</tr>
<tr>
<td>• Selecting dimension members for planners to work with</td>
<td></td>
</tr>
<tr>
<td>• Setting grid properties</td>
<td></td>
</tr>
<tr>
<td>• Setting dimension properties</td>
<td></td>
</tr>
<tr>
<td>• Adding formula rows and columns</td>
<td></td>
</tr>
<tr>
<td>• Setting display properties</td>
<td></td>
</tr>
<tr>
<td>• Setting printing options</td>
<td></td>
</tr>
<tr>
<td>• Adding and updating validation rules</td>
<td></td>
</tr>
<tr>
<td>Define page axis and point of view</td>
<td>See Defining Simple Form Page and Point of View.</td>
</tr>
</tbody>
</table>
### Table 18-13  (Cont.) Simple Form Creation Checklist

<table>
<thead>
<tr>
<th>Task</th>
<th>Want to Know More?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select members</td>
<td>See Using the Member Selector.</td>
</tr>
<tr>
<td>Set form precision, context menus associa-</td>
<td>See Setting Form Precision and Other Options.</td>
</tr>
<tr>
<td>tions, and whether to enable dynamic user</td>
<td></td>
</tr>
<tr>
<td>variables</td>
<td></td>
</tr>
<tr>
<td>Select business rules and set properties</td>
<td>See About Rules.</td>
</tr>
<tr>
<td>Define access permissions</td>
<td>See Setting Up Access Permissions.</td>
</tr>
<tr>
<td>Design formula rows and columns</td>
<td>See Designing Forms with Formula Rows and Columns.</td>
</tr>
<tr>
<td>Design data validation rules</td>
<td>See Designing Forms with Data Validation.</td>
</tr>
</tbody>
</table>

To create simple forms:

1. Click the Navigator icon, then under Create and Manage, click Forms.

2. Click .

3. On the Properties tab, enter a name of up to 80 characters, and an optional description of up to 255 characters.

4. Select the Cube associated with the form. See Forms and Cubes.

5. **Optional:** Enter instructions describing for planners how to work with the form.

6. Click Next, and see Defining the Layout.

### Defining the Layout

When you create forms, they initially only contain one row and column, and all dimensions, except for the optional attribute dimensions, are in Point of View. Add rows and columns as needed. When setting row and column layout:

- Assign at least one dimension to the row and column axis.
- You can't select the same dimension for multiple axes. To use dimensions on multiple axes, set user variables in the point of view.
- You can select a dimension from any axis and drag it to another

---

**Note:**

Attribute dimensions are optional and are displayed in their own area on the Layout tab. Drag the attribute dimension to the point of view, page, row, or column to use it in the form.
Caution:

When working with attributes as filters, don’t drag the attribute on any axis as you’ll cause a cross-dimensional reference which will have a performance impact. When using an attribute dimension as a filter, there is a form option that shows the associated attribute (just like there is an option to show alias).

To define the layout:

1. Open the form, and click **Layout**. See [Selecting and Opening Forms and Folders](#).

2. Click 🔄 to select the dimensions to use on **Rows**, **Columns**, or within a row or column. Optionally, you can also select the attribute dimensions to use on **Point of View**, **Page**, **Rows**, and **Columns**.

3. Select each dimension’s members.

4. Optional: To change the order of dimensions in rows or columns, click 🔄, and then **Move Dimension Up** or **Move Dimension Down**.

5. Select a row header (such as 1 or 2) to specify row properties, or a column header (such as A or B) to set column properties described in **Segment Properties**.

   To enable planners to transfer global assumptions from a test to a production environment for a simple form, select **Global Assumptions Form**, and then update the form to store global assumptions. See [Designing Forms with Global Assumptions](#).

6. Optional: Add formula rows and columns. See [Adding Formula Rows and Columns](#).

Subtopics:

- Segment Properties
- Setting Form Grid Properties
- Setting Dimension Properties
- Setting Display Properties
- Setting Printing Options
- Including Data Validation Rules in Forms
- Enabling Drilling on Shared Members

### Segment Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply to all rows</td>
<td>Clear to specify different properties for individual rows.</td>
</tr>
</tbody>
</table>
Table 18-14  (Cont.) Segment Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply to all columns</td>
<td>Clear to specify different properties for individual columns.</td>
</tr>
<tr>
<td>Hide</td>
<td>Conceal a column or row</td>
</tr>
<tr>
<td>Read-only</td>
<td>Create a read-only row or column, enabling planners to compare old, read-only data with new, editable data.</td>
</tr>
<tr>
<td>Show separator</td>
<td>Create a bold border before the segment to visually distinguish it</td>
</tr>
<tr>
<td>Suppress hierarchy</td>
<td>Suppress indentation.</td>
</tr>
<tr>
<td>Suppress missing data</td>
<td>Hides empty rows or columns. Clear to display rows or columns with &quot;#MISSING&quot; in cells when data is missing.</td>
</tr>
<tr>
<td>Suppress invalid Scenario/Time Periods</td>
<td>This option ties the column display to the start and end period for the scenario. If selected, it suppresses the display of time periods outside of the scenario time period range.</td>
</tr>
<tr>
<td>Column width</td>
<td>• Default—Use the column width defined at the grid level (under Grid Properties)</td>
</tr>
<tr>
<td></td>
<td>• Small—50px</td>
</tr>
<tr>
<td></td>
<td>• Medium—100px</td>
</tr>
<tr>
<td></td>
<td>• Large—300px</td>
</tr>
<tr>
<td></td>
<td>• Size-to-Fit—Column will expand to fit data</td>
</tr>
<tr>
<td></td>
<td>• Custom—Specify the number of pixels</td>
</tr>
<tr>
<td>Row height</td>
<td>• Default—Use the row height defined at the grid level (under Grid Properties)</td>
</tr>
<tr>
<td></td>
<td>• Medium—Use standard height</td>
</tr>
<tr>
<td></td>
<td>• Size-to-Fit—Row will expand to fit data</td>
</tr>
<tr>
<td></td>
<td>• Custom—Specify the number of pixels</td>
</tr>
<tr>
<td>Enable drop-down for dimensions</td>
<td>Allows you to set up drop-down member selectors on row dimensions in Oracle Smart View for Office grids and Planning web forms. When these drop-down selectors are enabled, form users can directly select a member from the drop-down list on the row dimension in the form. For example, suppose there are two dimensions placed on the row axes, Entities and Line item; the member selector drop-down may be enabled for the Entity dimension members, for the Line item dimension members, or for both. The drop-down member selector also allows users to add data to member rows that may have otherwise been suppressed.</td>
</tr>
</tbody>
</table>

Setting Form Grid Properties

Grid properties define how rows and columns display.

To set grid properties:
1. See Selecting and Opening Forms and Folders.
2. In Grid Properties, set row and column properties as follows:

| Table 18-15  Form Grid Properties |
|-----------------|-----------------------------------|
| **Option**      | **Description**                   |
| Suppress missing blocks | On multiple rows: Greatly improves the efficiency of Suppress missing data. This option can degrade performance if no or few rows are suppressed. Test forms before and after using this setting to determine if performance is improved. Note that if you use this option: • Some suppressed blocks may ignore Dynamic Calc members • Row members may not indent |
| Suppress missing data | Hide rows or columns without data. Clear to display "#MISSING" in cells when data is missing. |
| Suppress invalid data | Hide rows or columns with invalid data. Cells with invalid data are read-only. |
| Default row height | • **Size-to-Fit**—Row will expand to fit data • **Custom**—Specify the number of pixels |
| Default column width | Specify the width in pixels: • **Small**—50px • **Medium**—100px • **Large**—300px • **Size-to-Fit**—Column will expand to fit data • **Custom**—Specify the number of pixels |
| Suppress invalid Scenario/Time Periods | This option ties the form grid display to the start and end period for the scenario. If selected, it suppresses the display of time periods outside of the scenario time period range. |
| Global Assumptions Form | To transfer global assumptions from a test to a production environment for a simple form, select Global Assumptions Form, and update it to store assumptions. See Designing Forms with Global Assumptions. |
Table 18-15 (Cont.) Form Grid Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Autosave</td>
<td>This option lets planners use successive undo actions with Ctrl+Z. It also automatically saves their changes when they navigate between cells.</td>
</tr>
</tbody>
</table>

**Note:**

For optimal performance, use only dense dimensions on rows and columns.

See About Autosave.

Run Form Rules on Autosave

Only available if Enable Autosave is selected. If Run Form Rules on Autosave is selected, dynamically calculated cells that depend on modified and saved values (for example, a row with a formula that calculates a percentage of an aggregated parent value) are updated and display on a green background.

Suppress Missing also Suppresses Zero

When this option is selected along with the Suppress missing data option for forms, all rows or columns containing both #Missing and zeros are suppressed.

When selected, this setting overrides the runtime suppression selections made for suppressing missing data or zeroes in Oracle Smart View for Office.

3. Click Save to continue, or Finish to save and close the form.

Setting Dimension Properties

You can specify dimension display properties such as if to display the member names or aliases, and whether to permit planners to view member formulas. These properties apply to row, column, page, and point of view dimensions.

To set dimension properties:

1. Open the form, and then click Layout.

   See Selecting and Opening Forms and Folders.

2. Click in a point of view, page, row, or column dimension, then select Dimension Properties, and apply properties such as:

Table 18-16 Dimension Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply to all row</td>
<td>column</td>
</tr>
</tbody>
</table>
Table 18-16  (Cont.) Dimension Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply to all POV dimensions</td>
<td>Apply properties to all point of view dimensions</td>
</tr>
<tr>
<td>Start expanded</td>
<td>For row or column dimensions: Expand and display the dimension member list.</td>
</tr>
<tr>
<td>Enable custom attributes</td>
<td>For row or column dimensions: Use custom attributes.</td>
</tr>
<tr>
<td>Drill on shared members</td>
<td>For row or column dimensions: Enable drilling on shared members when the shared member is on a parent member for the main hierarchy.</td>
</tr>
</tbody>
</table>

3. Click **Save** to continue, or **Finish** to save and close the form.

Setting Display Properties

Define display options on the **Layout** tab.

**Note:**

- Assign the Account dimension to a row axis.
- Account, Entity, Versions, and Scenario dimensions can't be assigned to the column axis.
- The Entity dimension can be assigned to any axis.
- Assign Version and Scenario dimensions to the page or Point of View axis.

To set display properties:

1. Open the form, and then click **Layout**.
   
   See **Selecting and Opening Forms and Folders**.

2. Select **Display Properties**, and then options such as:

Table 18-17  Display Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hide form</td>
<td>Conceal forms that are part of a composite form, or accessed from menus or task lists.</td>
</tr>
<tr>
<td>Display missing values as blank</td>
<td>Display cells without data as empty cells. Clear to display &quot;#MISSING&quot;. See Displaying #MISSING with Smart Lists.</td>
</tr>
</tbody>
</table>
Table 18-17  (Cont.) Display Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable account annotations</td>
<td>Enable planners to add comments to accounts to which they have write access to the Account, Entity, Scenario, and Version members. Account-level annotations can vary by different combinations of Scenario, Version, and Entity dimensions. Comments can be plain text or URL links.</td>
</tr>
<tr>
<td>Allow multiple currencies per entity</td>
<td>Enable entities to support multiple currencies, regardless of the base currency.</td>
</tr>
<tr>
<td>Enable Mass Allocate</td>
<td>Planners must have the Mass Allocate role to use this option. See Working with Planning.</td>
</tr>
<tr>
<td>Enable Grid Spread</td>
<td>See Working with Planning.</td>
</tr>
<tr>
<td>Enable cell-level document</td>
<td>Enable planners to add, edit, and view documents in form cells, depending on access permissions. See Working with Planning.</td>
</tr>
<tr>
<td>Message for forms with no data</td>
<td>Enter custom text to display when invalid data exist. Leave blank to display There are no valid rows of data for this form.</td>
</tr>
<tr>
<td>Hide Save Confirmation Message</td>
<td>Prevents the form save confirmation message from being displayed to users.</td>
</tr>
</tbody>
</table>

3. Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Setting Printing Options

To set printing options:

1. Open the form, and then click **Layout**. See Selecting and Opening Forms and Folders.
2. Select **Printing Options**, and then settings such as:

Table 18-18  Printing Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Include supporting detail | Print supporting detail as extra rows in PDF files as follows:  
  Normal Order—In the same order as on the Supporting Detail page  
  Reverse Order—Before the member associated with it. Supporting detail for children displays above parents, and the order of siblings is preserved |
| Show comments        | Display cell text notes                                                                                                                                                                                  |
| Format data          | Apply number format settings                                                                                                                                                                             |
Table 18-18  (Cont.) Printing Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply precision</td>
<td>Print data using a specific number of decimal places</td>
</tr>
<tr>
<td>Show currency codes</td>
<td>If the form supports multiple currencies, print currency codes. Currency codes display based on their use in members on the form.</td>
</tr>
</tbody>
</table>

3. Click **Save** to continue, or **Finish** to save and close the form.

Including Data Validation Rules in Forms

You can add and update validation rules for grids, columns, rows, or cells, on the Layout tab. When processed, rules may change the color of cells, display validation messages to planners during data entry, and change the promotional path for approval units.

**Note:**

Before adding data validation rules, consider the function it will perform. For information and best practices about using validation rules, see **Managing Data Validation**.

To include data validation rules in forms:

1. Open the form, and then click **Layout**.
   
   See **Selecting and Opening Forms and Folders**.

2. Select **Validation Rules**, and then select validation rule options:

**Note:**

The right-click menu options that display are context-sensitive, and depend on if rules and menu options were already added.

Table 18-19  Validation Rules Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add/Edit Validation Rules</td>
<td>Create or modify existing rules in the Data Validation Rule Builder dialog box.</td>
</tr>
<tr>
<td>Copy</td>
<td>Paste Validation Rules</td>
</tr>
<tr>
<td>Validate only for users with access to this form</td>
<td>If the current user doesn't have access to the form, don't run form validations when validating the approval unit.</td>
</tr>
</tbody>
</table>
Table 18-19  (Cont.) Validation Rules Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validate only for pages with existing</td>
<td>Run validations only on page combinations that have potential blocks. The only exceptions being page combination having any Dynamic Calc, Dynamic Calc and Store, Label only, or Store with one child member settings. In this case, the page is loaded.</td>
</tr>
<tr>
<td>blocks</td>
<td></td>
</tr>
<tr>
<td>Validate only for cells and pages the user</td>
<td>Run validations as the current user, using their security, and, not an administrator.</td>
</tr>
<tr>
<td>has access to</td>
<td></td>
</tr>
</tbody>
</table>


4. In the form, click **Next** to continue building the form, and then validate and save it.

Enabling Drilling on Shared Members

Form designers often use shared members to create alternate rollup structures within an application. To increase usability for these alternative hierarchies, the application provides the ability to drill on a shared member for a given dimension. The option to drill is a dimension property which provides the flexibility to focus the drill behavior to a particular dimension. The drilling option is only available for dimensions placed on the row or column axis during form design. Page or POV dimensions don’t have drill capability.

For example, using the following hierarchy representing the Entity dimension:
A form with iDescendants(Southwest) defined on the row would return the following members:

- TX
- FL
- NM
- South
- CA
- AZ
- CO
- West

To enable drilling on shared members in a form:

1. Open the form, and then click **Layout**.
   
   See [Selecting and Opening Forms and Folders](#).

2. Click in a row or column dimension to expand **Dimension Properties**.

3. Under **Dimension Properties**, select **Drill on shared members**.

4. Click **Save**.
About Precision Settings

In Other Options, you can control data precision by applying minimum and maximum values for different account types. For example, you can truncate and round the decimal portion of longer numbers. Precision settings affect just value display, not their stored values. For example, if Minimum Precision is 2, and the value 100 from Q1 is spread to Jan, Feb, and Mar, the month cells display 33.33 when they are not selected. When selected, they display their actual values (for example, 33.33333333333333). Because the number of decimal places for storing values is finite, when the values for Jan, Feb, and Mar aggregate to Q1, 33.33333333333333 is multiplied by 3, and Q1’s value is 99.99999999999998.

Setting Form Precision and Other Options

To set precision and other options:

1. See About Precision Settings.
2. Open the form, and then click Other Options.
   See Selecting and Opening Forms and Folders.
3. In Precision, specify the number of decimal positions displayed in a cell for Currency Values, Non-Currency Values, and Percentage Values.
   Specify Minimum values to add zeros to numbers with few decimal places. Specify Maximum values to truncate and round the decimal portion of large numbers.
   For example:

   Table 18-20  Data Precision Examples

<table>
<thead>
<tr>
<th>Value</th>
<th>Minimum Precision</th>
<th>Maximum Precision</th>
<th>Displayed Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>0</td>
<td>Any</td>
<td>100</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>Any number greater than or equal to 3 or None</td>
<td>100.000</td>
</tr>
<tr>
<td>100.12345</td>
<td>Any number less than or equal to 5</td>
<td>None</td>
<td>100.12345</td>
</tr>
<tr>
<td>100.12345</td>
<td>7</td>
<td>None</td>
<td>100.1234500</td>
</tr>
<tr>
<td>100.12345</td>
<td>Any number less than or equal to 3</td>
<td>3</td>
<td>100.123</td>
</tr>
<tr>
<td>100.12345</td>
<td>0</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>100.12345</td>
<td>2</td>
<td>4</td>
<td>100.1235</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>4</td>
<td>100.00</td>
</tr>
</tbody>
</table>

   Note:

   Your selections override the precision set for the currency member. To use the currency member's precision setting instead, select Use Currency member precision setting.
4. Associate menus with the form by moving them from **Available Menus** to **Selected Menus**. For multiple menus, click a "move" options on the right to specify the display order.

5. Select **Enable dynamic user variables** to allow dynamic user variables in the form (see *Working with Planning*.)

**Creating Asymmetric Rows and Columns**

Asymmetric rows and columns contain different sets of members selected across the same dimensions. For example:

- **Row/Column A**—Scenario = Actual, Time Period = Q1
- **Row/Column B**—Scenario = Budget, Time Period = Q2, Q3, Q4

To create asymmetric rows or columns:

1. Open the form, and then click **Layout**. See *Selecting and Opening Forms and Folders*.
2. Click **** to select the dimension to modify.
3. Click ** ** to the right of the dimension name, and then modify the members selected for this dimension. See *Using the Member Selector*.
4. Click **Save** to continue, or click **Finish** to save and close.

**Adding Formula Rows and Columns**

You can define formulas for rows and columns. For example, you can create a formula column to calculate the percentage variance between monthly sales columns. The formula for a row or column applies to all row or column dimensions. To define or assign existing formulas to forms, select the row or column on the **Layout** tab, and use the options under **Segment Properties**. See *Creating Formulas*.

**Tip:**

Add a formula row between two other rows to create a blank row. Blank rows help you visually separate subtotals and totals in a form.

To add formula rows and columns:

1. Open the form, and then click **Layout**. See *Selecting and Opening Forms and Folders*.
2. On the **Layout** tab, right-click **Rows** or **Columns**.
3. Select **Add Formula Row** or **Add Formula Column**.
4. Click the new **Formula Label**, and then enter the formula name.
5. Click the row or column number and specify **Segment Properties**, such as **Display formula on form** so planners can view the formula when they click **.**
6. For each dimension in **Formula Data Type**, select how to display the formula result, such as a percentage or in date format.

7. Enter the formula, and then click 📊. See **Editing Formulas**.

8. Click **Validate** to find and fix any errors.

---

### Defining Simple Form Page and Point of View

You can select dimensions and members for the page axis and point of view (POV). POV dimensions and members must be valid for the form cube, and not assigned to a page, column, or row axis. The POV sets the members that define data intersections and the form’s context. If you use user variables, variable names displays in the POV. See **Managing User Variables**

To define page axis and point of view:

1. Open the form, and then click **Layout**.
   
   See **Selecting and Opening Forms and Folders**.

2. Click 📍 and then drag that dimension to **Page**.

3. Click 📊 for each page axis dimension and select members. See **Using the Member Selector**.

   **Note:**

   To shorten the number of members displayed in the member list so that the Search box can be easily seen, from the Home page, click **Application**, and then click **Settings**. For the **Number of Items on the Page Drop-down** option, enter a lower value.

4. Repeat Steps 2 and 3 as needed. Assign multiple dimensions to the page axis to enable planners to change dimensionality while entering data. They can use **Display Options** to use the most recent selection.

5. Specify **Dimension Properties**. See **Editing Dimension Properties in the Simplified Dimension Editor**.

6. Click 📍 and drag a dimension to **Point of View** to add them, or members, to the form’s point of view.

7. Specify **Dimension Properties**. See **Editing Dimension Properties in the Simplified Dimension Editor**.

8. Click **Save** to continue, or **Finish** to save and close.

---

### Designing Specific Types of Forms

**Related Topics**

- Designing Forms for Multiple Currencies
- Designing Forms with Formula Rows and Columns
- Designing Forms with Data Validation
Designing Forms for Multiple Currencies

To allow users to work with currencies other than entities’ base currencies, perform one task:

- Select members from at least two currencies to compare converted currencies in the same form.
- Assign the Currency dimension to the page axis and select reporting currencies as members to convert currencies for all the members in the form. Users can then select a currency member from the page axis and launch the Calculate Currencies business rule to view values in that currency.

Designing Forms with Formula Rows and Columns

Formula rows and columns contain formulas that perform mathematical calculations on grid members. For example, you might want to rank values in a particular column or calculate variance between two rows. A formula consists of grid references, arithmetic operators, and mathematical functions. To define or assign existing formulas to forms, select the appropriate row or column on the Layout tab and then make selections under Segment Properties.

See Adding Formula Rows and Columns. For information about creating formulas and using the mathematical functions, see Form Formula Functions.

Designing Forms with Data Validation

You can design forms that include predefined data validation rules that help implement business policies and practices. You can specify cell colors and data validation messages that are generated on the form if entered data violates a validation rule. Data validation rules are saved as part of the form. See Including Data Validation Rules in Forms and Managing Data Validation.

Designing Forms with Global Assumptions

Form designers typically create a specific-purpose form that stores global assumptions (also called “drivers” or “driver data”), such as a depreciation value, tax rate, or unit price. This is useful if you want to migrate the form’s driver data using Migration, for example, from a test environment to a production environment. Such forms are typically small, and are used as “lookup tables” that drive calculations.

To enable transferring global assumptions from a test to production environment, during form design for a simple form, in the Layout tab select Grid Properties and Global Assumptions Form. Then update the form to store driver data such as a tax rate. When migrating with Migration, this enables migrating the driver data contained in the tagged form. Note that global assumptions can’t be used with composite forms or forms with multiple segments.
Designing Forms for Rolling Forecasts

Related Topics

• About Rolling Forecasts
• Creating Rolling Forecasts
• Modifying Rolling Forecast Variables

About Rolling Forecasts

In a traditional forecast, the forecast cycle is always tied to the fiscal year end, and the months in the forecast period keep reducing as the months in the fiscal year progress.

Rolling forecasts differ from traditional forecasts in that they are continuous without regard to the annual fiscal year end period. The periods in a rolling forecast roll along based on the predefined window for the rolling forecast. The periods are generally defined on a monthly or quarterly basis. Monthly rolling forecasts are generally in 12-month, 18-month, or 24-month cycles. In a 12-month cycle, the 12-month period constantly shifts each month, and every month the forecast is for the next twelve months without regard to the actual fiscal year end.

For example, assume a company has a fiscal calendar for July through June. In the first month of year (Jul, FY11) the company’s planners fill in the forecast scenario for the periods Jul 11 – Jun 12. In the next month (Aug 11), the planners again fill in the forecast scenario with numbers for the next 12 months (Aug 11- Jul 12), even though the period of Jul 12 pertains to the next fiscal year of Jul FY12-Jun FY13.

Following are some examples of rolling forecasts:

Figure 18-1  12-Month Rolling Forecast

<table>
<thead>
<tr>
<th>Year and Period in Columns With No Additional Segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual</td>
</tr>
<tr>
<td>Jul</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
<tr>
<td>50</td>
</tr>
</tbody>
</table>

Figure 18-2  Quarterly Rolling Forecast

<table>
<thead>
<tr>
<th>FY12 Q2 Review</th>
<th>FY12 Q3 Review</th>
<th>FY 12 Q4 Review</th>
<th>FY13 Q1 Review</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>A</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>A</td>
<td>A</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>A</td>
<td>A</td>
<td>A</td>
<td>F</td>
</tr>
</tbody>
</table>
Creating Rolling Forecasts

Note:

Only administrators can create and work with rolling forecasts. This includes the ability to see the rolling forecast option when designing a form, the ability to move a rolling forecast from the user interface, and the ability to delete or modify substitution variables.

To design a form for a rolling forecast:

1. Create a new form.
   See Selecting and Opening Forms and Folders.
2. In the Layout tab, drop the Years and Period dimensions in the column axis.
3. Right-click the column segment header and select Rolling Forecast Setup.
   The Rolling Forecast Setup menu option is only available when the Years and Period are in the same grid axis (row or column).
4. In the Rolling Forecast Setup dialog box, enter the following information:

Table 18-21  Rolling Forecast Setup Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefix</td>
<td>Allows rolling forecast substitution variables to be distinct from other substitution variables; for example, 4QRF designates that the forecast is a 4-quarter rolling forecast.</td>
</tr>
</tbody>
</table>
### Table 18-21 (Cont.) Rolling Forecast Setup Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuse existing substitution variables</td>
<td>Select if you wish to specify a prefix that you have previously used.</td>
</tr>
<tr>
<td>Start Year</td>
<td>The year in which the rolling forecast starts; for example, FY11.</td>
</tr>
<tr>
<td></td>
<td>Either enter the start year or click <img src="477x612" alt="Image" /> to open the Member Selection dialog box.</td>
</tr>
<tr>
<td></td>
<td>If you entered a prefix that matches the prefix of an existing rolling forecast substitution variable and selected Reuse existing substitution variables, the Start Year is automatically filled in with the start year of the existing substitution variable.</td>
</tr>
<tr>
<td>Start Period</td>
<td>The period in which the rolling forecast starts; for example, Q1.</td>
</tr>
<tr>
<td></td>
<td>Either enter the start period or click <img src="484x483" alt="Image" /> to open the Member Selection dialog box.</td>
</tr>
<tr>
<td></td>
<td>If you entered a prefix that matches the prefix of an existing rolling forecast substitution variable and selected Reuse existing substitution variables, the Start Period is automatically filled in with the start period of the existing substitution variable.</td>
</tr>
<tr>
<td>Number of Periods</td>
<td>Number of year/period combinations that will be generated as separate segments.</td>
</tr>
</tbody>
</table>

5. Click **Generate**.

The defined substitution variables are created, and additional column segments are created in the form that contain the substitution variable combinations for the rolling forecast.

---

**Note:**

- Substitution variables are based on the periods selected for the Start Year and Start Period. See Selecting Substitution Variables as Members.
- When designing a form, if the Rolling Forecast Setup dialog box is brought up from a row or column with the Year and Level 0 time period selected (for example, FY12/Jan), the Start Year and Start Period are automatically filled in. Values are not automatically filled in if members in the column are selected using functions, variables, or non-level-0 members.
- To reuse the rolling forecast variables in a different form, right-click the column header in the new form to invoke the Member Selector.
Modifying Rolling Forecast Variables

You can revise the values for rolling forecast substitution variables directly in the form.

To modify rolling forecast variables in a form:

1. From the Home page, click Data.
2. Open the rolling forecast form.
3. Select a column header in the form, click Actions, and then select Set Rolling Forecast Variables.
4. In the Set Rolling Forecast Variables dialog box, enter or edit values for the Years and Period dimensions.
   - You can shift the values up or down by changing the selection next to Shift Values By. When you change the selection next to Shift Values By, the values for the Years and Period dimensions are automatically repopulated to show the resulting year and period values after the shift.
5. Click Apply.
   - The new values flow through to all the forms where these substitution variables are used, and those forms will reflect the changes.

Working with Forms and Form Components

Related Topics
- Selecting and Opening Forms and Folders
- Previewing Forms
- Printing Form Definitions
- Searching for Forms
- Editing Forms
- Moving, Deleting, and Renaming Forms
- How Cell Formatting in Smart View Persists in Planning

Selecting and Opening Forms and Folders

Use these procedures to select and open form folders and the forms they contain.

To select and open forms or form folders:

1. Click the Navigator icon, and then under Create and Manage, click Forms.
2. Perform one of the following steps:
   - To open a folder, select a form folder beneath Folders.
   - To open a form, select a form from the list displayed beneath Library when the appropriate form folder is open.
3. After you select a form folder, use the buttons next to Folders to create, rename, and assign access to the folder. After displaying a form, use the buttons above
**Form** to create, edit, move, delete, and assign access to forms. The icon next to the form name indicates the type of form:

- Simple form
- Composite form
- Master composite form
- Ad hoc grid

For information about setting up ad hoc grids, see *Working with Planning*.

### Previewing Forms

While you're designing forms, you can preview the dimensions that are assigned to the Point of View, columns, rows, and page axes. Previewing displays member attributes, alias, and data associated with forms, although new data can't be entered.

Previewing completes regular form design validation checks, and checks for proper evaluation of any data validation rules included in the form. Data validation rules must be properly completed before the form can be saved. In addition, data validation rules are saved as part of the form. If you don't save changes to a form, any data validation rule changes made after the form was last saved are lost.

To preview a form's design:

1. With a form open, click **Preview**.
   - The form opens in edit mode in a new tab.
2. Resolve any issues reported during the design validation checks, including any issues with data validation rules.
3. Save the form to ensure that updates are saved, including any changes to data validation rules.

### Printing Form Definitions

Administrators can print form definition reports that include information on dimension members, business rules, access permissions, and other form components.

To create and print form definition reports:

1. Click the **Navigator** icon, and then under **Monitor and Explore**, click **System Reports**.
2. Select the **Forms** tab.
3. Select the form definitions to print by moving them from **Available Forms** to **Selected Forms**.
4. **Optional**: Select **Include Member Selection List** to include column and row members on the report.
5. **Optional**: Select **Include Business Rules** to include associated business rules.
6. Click **Create Report**.
Adobe Acrobat generates a consolidated report, including:

- Cube
- Description
- Column dimension and members and additional column definitions
- Row dimension and members and additional row definitions
- Page and Point of View dimensions
- Form access permissions
- Associated business rules

7. To print the report, select **File**, and then **Print** on the Adobe toolbar.

Searching for Forms

To search for forms:

1. Click the **Navigator** icon and then under **Create and Manage**, click **Forms**.
2. For **Search**, enter part or all the form name.
   Ignoring capitalization, Search finds the next match.
3. Click ⬇️ to search forward (down) or ⬆️ to search backwards (up).

Editing Forms

You can edit the layout, members, and properties of both simple and composite forms. For example, you can add formula rows or columns to a simple form, or add forms to a composite form.

**Note:**

You can edit existing composite forms, but you can't create new ones. Oracle recommends that you shift your usage to dashboards instead of composite forms. See **Designing Dashboards**.

- **Editing Simple Forms**
- **Editing Composite Forms**

Editing Simple Forms

To edit simple forms:

1. Select the form, then click ✍️ (see **Selecting and Opening Forms and Folders**).
2. Select:
   a. **Properties** to edit the form name, description and instructions. See **Creating Simple Forms**.
b. **Layout** to edit form layout. See *Defining the Layout*.

c. **Other Options** to edit form precision and to change which context menus are associated with the form. See *Setting Form Precision and Other Options*.

d. **Business Rules** to change which business rules are associated with the form, or modify business rule properties. See *Administering Rules*.

3. Click **Finish** to save your work and close the form.

### Editing Composite Forms

You can edit existing composite forms, but you can’t create new ones. Oracle recommends that you shift your usage to dashboards instead of composite forms. See *Designing Dashboards*.

To edit existing composite forms:

1. Take one of these actions:
   - Select the form, click the **Show Usage** icon, select the simple form, and then click **Edit**.
   - Select the form, and then click **Edit** (see *Selecting and Opening Forms and Folders*).

2. Select:
   - **Properties** to edit the composite form name, description or instructions.

    **Note:**
    When editing a composite form, if this message displays, "Modifications have been made to one or more included forms; if you want to save changes to common dimensions, save the composite form," determine what changes were made to the common dimensions of the included simple forms before saving changes to the composite form.

3. Click **Finish** to save your work and close the form.

### Setting Composite Form Layout

The application provides tools that allow you to set the composite form layout that is best for your application. Each area in the composite form is called a section. Initially,
you specify whether to divide the composite form layout into two side-by-side sections, or two sections that are stacked one above the other. There is also a custom layout option.

**Note:**

You can edit existing composite forms, including setting composite form layout, but you can't create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See Designing Dashboards.

To set composite form layout in existing composite forms:

1. Open the composite form, and then click **Layout**.
   
   See Selecting and Opening Forms and Folders.

2. In **Select Layout**, select an option:

   • to create your own composite form layout.
     
     **Note:** The custom layout option is selected by default.

   • to split the composite form into two sections, one on top of the other, divided by a horizontal line.

   • to split the composite form into two side-by-side sections divided by a vertical line.

   The selected layout is displayed.

3. Add, rearrange, or delete simple forms as desired.

   See Adding Simple Forms to a Composite Form Layout, Rearranging Forms in the Layout, and Deleting Simple Forms from a Composite Form.

4. **Optional**: Click in the upper right side of a section to select the following additional layout options for that section:

   • **Split Horizontally** to split the section into two sections, one above the other.

   • **Split Vertically** to split the section into two side-by-side sections.
Adding Simple Forms to a Composite Form Layout

You can edit existing composite forms, including adding simple forms to a composite form layout, but you can’t create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See Designing Dashboards.

To add a simple form to a section in an existing composite form, do one of the following:

- Drag a form from the Forms in <Form Folder> pane to the desired section.
- Click in the desired section, select , and select Add Form. In the Form Selector dialog box, select a form and click OK.
- Expand Section Properties and click . In the Form Selector dialog box, select a form and click OK.

When you’re adding simple forms to a composite form, note the following:

- Composite forms can contain simple forms and ad hoc forms.
- During runtime, the simple forms selected for the composite form display from left to right, and then from top to bottom within each composite form section.
- If you select Group as Tabs, the form displays in the order selected.
- You can drag simple forms between sections of a composite form.
Rearranging Forms in the Layout

To rearrange the simple forms in the composite form layout, expand Section Properties, select a form, and click an arrow key. You can:

- Move the form to the top
- Move the form up
- Move the form down
- Move the form to the bottom

Note:

You can edit existing composite forms, including rearranging forms in the layout, but you can't create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See Designing Dashboards.

Editing Simple Forms from within a Composite Form

While editing a composite form, you can edit a single form from the Layout tab. This option isn't available for ad hoc forms. Access permissions apply as described in Forms and Permissions.

Note:

You can edit existing composite forms, including editing simple forms within composite forms, but you can't create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See Designing Dashboards.

To edit a simple form from a composite form:

1. Within the composite form, click the Layout tab.
2. Right-click a simple form, and then select Form Designer.
3. Edit the simple form as described in Editing Forms.

Deleting Simple Forms from a Composite Form

You can edit existing composite forms, including deleting simple forms from a composite form, but you can't create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See Designing Dashboards.

To delete a simple form from an existing composite form, do one of the following:

- Right-click the form and select Delete.
- Select the form in Section Properties, and click .
- Uncheck the form in the Form Selector dialog box and click OK.
Setting Composite Form Section Properties

Each section in a composite form is associated with properties set during composite form creation.

**Note:**
You can edit existing composite forms, including setting section properties, but you can't create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See Designing Dashboards.

To set properties in existing composite forms:

1. Open the composite form, and then click Layout.
   See Selecting and Opening Forms and Folders.
2. Expand Section Properties.
3. Click in a composite form section and set the properties as desired.

### Table 18-22  Composite Form Section Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Forms** | Displays the simple forms in the section. The following options are available for each form selected:  
- Display forms as tabs  
- Add form  
- Remove form  
- Edit form label  
- Move to top  
- Move up  
- Move down  
- Move to bottom |
| **Name** | Section name to be displayed at the top of the section in Preview mode and at runtime. Select 📖 to select a text style and color for the section name. |
| **Height** | Select:  
- Automatic to have the application set the height.  
- % (percentage sign) to set section height to a percentage of the composite form height. |
### Table 18-22  (Cont.) Composite Form Section Properties

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Width</strong></td>
<td>Select:</td>
</tr>
<tr>
<td></td>
<td>• Automatic to have the application set the width.</td>
</tr>
<tr>
<td></td>
<td>• % (percentage sign) to set section width to a percentage of the composite form width.</td>
</tr>
<tr>
<td><strong>Forms per Row</strong></td>
<td>Select:</td>
</tr>
<tr>
<td></td>
<td>• Automatic to have the application set the number.</td>
</tr>
<tr>
<td></td>
<td>• Select a number from 1 to 20. The default is one form per row. If <strong>Forms per Column</strong> is set to a value other than Automatic, <strong>Forms per Row</strong> is set to Automatic.</td>
</tr>
<tr>
<td><strong>Forms per Column</strong></td>
<td>Select:</td>
</tr>
<tr>
<td></td>
<td>• Automatic to have the application set the number.</td>
</tr>
<tr>
<td></td>
<td>• Select a number from 1 to 20. The default is one form per column. If <strong>Forms per Row</strong> is set to a value other than Automatic, <strong>Forms per Column</strong> is set to Automatic.</td>
</tr>
</tbody>
</table>

**Note:** If you have grouped the forms as tabs, this option isn't available.

**Set scope for all common dimensions as global** Sets all the common dimensions across all the sections in the composite form to global and displays a list of the global dimensions in Page and Point of View in the Global Dimensions properties.

---

**Setting Composite Form Point of View and Page Dimensions**

The composite form point of view and page dimensions specify where within a composite form each Point of View and Page dimension name displays.
When you select a section in a composite form, the right panel displays:

- **Global Layout Dimensions**, which list the Point of View and Page dimensions that display in the composite form heading.
  
  Only dimensions that are common to all simple forms in all sections of the composite form and that contain the same members can be designated as Global.

- **Common Dimensions**, which list the Point of View and Page dimensions common to all the simple forms included in the selected composite form section.

You can specify where common dimensions display in composite forms. Common dimension display choices are:

- **Local** displays the dimension name in the simple form heading.

- **Section** displays the section name in the section heading.
  
  Only dimensions that are common to all simple forms in a section and that contain the same members can be displayed in the section heading.

- **Global** displays the dimension name in the composite form heading. For more information on global POV bars, see About Global and Local POVs.

**Creating Master Composite Forms**

You can design composite forms that have one master form and multiple simple forms. When you do so, the selection of members in the master form automatically filters to the members in the simple forms, and the simple forms show only the details that are relevant to the members highlighted in the master form.

For example, assume that a user is looking at a new computer line item in a form and wants to see the cash flow impact from this line item. In this scenario, you could design a composite form that includes the following forms:

- A master form called “New Computers” that contains the following dimensions and members:
  
  - Entity: MA
  - Scenario: Plan
A simple form called "Cash Flow Impact"

In the master composite form, the user highlights the row Computers/Base SP1.

**Figure 18-5  Master Composite Form: "New Computers"**

The simple form, "Cash Flow Impact" is filtered to show only the data that is relevant for the members highlighted in the master composite form, "New Computers": Computers, Base SP1, Plan, Working, and MA.

**Figure 18-6  Simple Form: "Cash Flow Impact"**

To designate a form as a master composite form:

1. Open the composite form, and then click **Layout**.
   
   See **Selecting and Opening Forms and Folders**.

2. Right-click the form, and the select **Tag as Master Composite Form**.

   Indicates that the form is a master composite form.

**Note:**

The master composite form applies to the entire composite form. So, for a composite form, there can be only one master form across all its sections.
To filter the data in a simple form (or forms) that is relevant to the data in a master composite form, right-click the master composite form and select **Apply Context**.

**Embedding Charts in Composite Forms**

Administrators can edit composite forms to display the data in sections as charts. Planners can also drill down to the next level by clicking the underlined links or chart areas.

**Note:**

You can edit existing composite forms, including embedding charts in composite forms, but you can’t create new composite forms. Oracle recommends that you shift your usage to dashboards instead of composite forms. See **Designing Dashboards**.

**Design suggestions:**

- Display the top section as a chart and the bottom section as a grid, so that planners can see the effect of data they enter in the bottom grid (when saved) as a chart on the top.
- Include the same ad hoc grid twice, one to display as a grid and the other to display as a chart. Users can then perform ad hoc operations (such as **Zoom In**, **Pivot To**, and **Keep Only**) on the grid and view the changes in the chart.
- Create dashboards. For example:
To embed charts in existing composite forms:

1. Open the composite form, and then click **Layout**.
   See Selecting and Opening Forms and Folders.

2. Click a composite form section, and then right-click on a form.

3. Select **Display as Chart**.
   **Display as Chart** toggles with **Display as Grid**, allowing you to switch between them.

4. On **Chart Properties**, select a chart type:
   - **Bar**—The length of each bar proportionally represents a value over an independent variable (for example, time)
   - **Horizontal Bar**—Like the regular bar chart, but turned on its side so that the dependent variable is displayed on the horizontal axis
   - **Line**—Displays data points (for example, sales of various product lines) over time, connected by lines
   - **Area**—Like the Line chart, but the area between the axis and the line is emphasized with color
   - **Pie**—Each slice of the pie chart proportionally represents a class of data in relation to the whole
   - **Scatter**—Each point represents the distribution of data for two variables
   Read the onscreen text for advice on selecting a chart type.

5. Click **OK**.

6. **Optional**: To set where the chart displays the values that the chart represents (called the Legend), click **Options**, then click **Legend**, select one of the following, and then click **OK**:
   - **Right**—To display the legend to the right of the chart (the default).
• **Bottom**—To display the legend at the bottom of the chart.
• **Left**—To display the legend to the left of the chart.
• **Top**—To display the legend at the top of the chart.

7. **Optional**: To set where the chart labels (that is, the member names or aliases) are displayed, on **Options**, click **Label**, select one of the following, and then click **OK**.
   - **Outside Max**—To display the label above bar charts or, for non-bar charts, display the label above the data point for positive values and below the data point for negative values. **Outside Max** is the default.
   - **Center**—To display the label centered on bar charts, or for non-bar charts, display the label above the data point for positive values and below the data point for negative values.
   - **Inside Max**—To display the label on the bar, near the top, or for non-bar charts, display the label below the data point for positive numbers and above the data point for negative numbers.
   - **Inside Min**—To display the label inside on the bar, near the bottom, or for non-bar charts, display the label above the data point for positive values and below the data point for negative values.
   - **Max Edge**—To display the label on the bar, or for non-bar charts, display the label at the data point.

### Moving, Deleting, and Renaming Forms

To move, delete, and rename forms:

1. Select the form.  
   See **Selecting and Opening Forms and Folders**.
2. Select a task:
   - To move a form, click **Move**, and then select the destination folder.
   - **Note:** You can move multiple forms simultaneously if they are in the same folder.
   - To delete a form, click **Delete**.
   - To rename a form, click **Rename**, and then enter the new name.
3. Click **OK**.

### How Cell Formatting in Smart View Persists in Planning

To select the formatting for a form:

1. In the form, right-click, and then select **Apply**.
2. Select:
   - **Cell Styles**—To use the application’s formatting
• **Custom Styles**—To use the formatting saved in Oracle Smart View for Office

For more information:
• On which Microsoft Excel formatting features are supported in Smart View and Planning, see "About Smart View Formatting in Planning Forms" in the *Working with Planning*.
• On saving Excel formatting, see the *Oracle Smart View for Office User's Guide*.

Managing Forms and Folders

Use the Form Management and the Business Rule Folders pages to manage folders and forms.

**Table 18-23   Tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create folders</td>
<td>Creating Folders</td>
</tr>
<tr>
<td>Move, delete, or rename folders</td>
<td>Working with Folders</td>
</tr>
<tr>
<td>Create forms</td>
<td>Creating Simple Forms</td>
</tr>
<tr>
<td>Assign permissions to forms and folders</td>
<td>About Assigning Permissions to Artifacts, Rules, and Folders</td>
</tr>
<tr>
<td>Move and delete forms</td>
<td>Moving, Deleting, and Renaming Forms</td>
</tr>
</tbody>
</table>

To view all forms or business rules in a folder, click the folder’s name in the left-hand folders area. To select all the forms, select the check box at the top of the forms list.

Creating Folders

Use folders to hierarchically organize forms and Calculation Manager business rule security. You can move folders within the hierarchy, and give folders the same name if they are on different hierarchical levels. You can’t:

• Delete folders unless they are empty
• Select multiple folders
• Rename, move, or delete the top-level folder called, respectively, Library and **CalcMgrRules**

To create folders:

1. Perform an action:
   - For form folders: Click the **Navigator** icon and then under **Create and Manage**, click **Forms**.
   - For Calculation Manager business rule folders: Click the **Navigator** icon , and then under **Create and Manage**, click **Rules Security**.

2. Select the folder under which to create the folder.
3. Above the folders list, click **Create**.
4. Enter the folder name.
5. Click **OK**.

### Working with Folders

To move, delete, or rename folders:

1. For form folders: Click the **Navigator** icon 
   and then under **Create and Manage**, click **Forms**.

   For business rule folders: Click the **Navigator** icon 
   and then under **Create and Manage**, click **Rules Security**.

2. Select the folder to move, delete, or rename.

3. Perform a task:
   - To move, click **Move**. Select the destination folder to which to move the selected folder.
   - **Note:** When you move folders, all nested folders, forms, and Calculation Manager business rules within them are also moved.
   - To delete, click **Delete**.
   - To rename, click **Rename**, and then enter the new name.

4. Click **OK**.

### Tip:

Did you know that folder names can be localized based on the user's browser locale? To localize a folder name (for example, from English to Japanese), you can customize the folder name on the **Artifact Labels** page in the **Tools** cluster. For more information, see Specifying Artifact Labels.

### Administering Rules

**Related Topics**

- About Rules
- Adding and Removing Business Rules in Forms
- Setting Business Rule Properties
- Viewing Rules Usage
- About Runtime Prompts
- Understanding Runtime Prompts
- About Runtime Prompts and Approvals Security
- Designing Secure Runtime Prompts
Using Groovy Rules
Oracle supports the creation of business rules written in the Groovy scripting language.

About Groovy Business Rules
Java API Reference for Groovy Rules
Groovy Business Rule Examples
Groovy Business Rule Tutorial Videos
Groovy Rule Business Scenarios
This section provides examples of business scenarios where you can use Groovy business rules.

Moving Modified Data Using Groovy Rules and Smart Push
Calculating Modified Data Using Groovy Rules

About Rules

With appropriate access, users can launch business rules from the application. Rules can also prompt users for input when rules are launched.

To launch business rules from within data forms or independently from the Launch menu, the rules must be deployed from Calculation Manager. For optimum performance, business rules that run in forms should be designed to execute within three minutes. For rules with a longer execution time, you can schedule batch processing or run the rules during non-peak hours.

You can diagnose performance issues with business rules and pinpoint and address any issues using activity reports. Activity reports identify which business rules are taking the longest to execute. You can then open the rules in Calculation Manager, review the log, and optimize the steps within the rule to improve performance.

For information on:
- Creating, updating, and deploying business rules, see Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud
- Using runtime prompts, see About Runtime Prompts
- Generating a report that details where rules are used, see Viewing Rules Usage
- Viewing an activity report to diagnose rules performance issues, see Viewing Activity Reports
- Optimizing business rules, see Optimizing Business Rules in Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud

Adding and Removing Business Rules in Forms

You can associate one or more business rules with a form, by cube. Planners can launch associated business rules from the form to calculate and allocate values. You can set whether each business rule associated with a form automatically launches when the form is opened or saved. You can also change the order in which the rules are launched or remove rules from a form.
Note:

Rules must be deployed to Planning from Calculation Manager before you can add rules to Planning forms. See About Rules.

Likewise, if a rule is deleted in Calculation Manager, the rules must be redeployed to Planning in order for them to no longer be used in Planning. Optionally, you can manually remove a rule from a form using the procedure in this topic.

To add, update, or remove business rules in forms:

1. Take an action:
   - To update the current form, click Actions, and then click Business Rules.
   - To open a form for editing so you can associate rules, go to the Home page and click the Navigator icon and then under Create and Manage, click Forms. Select the form, click , and then click Business Rules.

2. From the Cube drop-down menu, select the cube.

3. For composite forms, select Only execute rules that are set to 'Run After Save' on contained forms where data is changed to launch only the rules where Run After Save is selected for each simple form included in the composite form. If this option is not selected, none of these rules will run after the form is saved.

4. From the Business Rules list, select the business rules to associate with the form, and add them to Selected Business Rules. To remove associated rules, select the rules under Selected Business Rules and remove them. See Using the Member Selector.

   By default, the Calculate Form and Calculate Currencies business rules are selected. Calculate Form is automatically created for forms to calculate subtotals. Calculate Currencies is created for forms that include multiple currencies in a row, column, or page, to enable converting values among the available currencies. You can remove Calculate Currencies if you use customized calculation scripts to calculate currency conversions. You can remove Calculate Form to prevent planners from calculating data in forms.

5. To change the order of selected business rules (the order in which rules display and launch), select a business rule in Selected Business Rules and click the up or down arrow to move it up or down in the list. The rule listed first displays and launches first; the rule at the bottom of the list displays and launches last.

Caution:

The order in which business rules launch is important and may affect data. For example, it's important to convert currencies first, before subtotaling.

7. Click **Save** to save your work and continue creating or editing the form, or click **Finish** to save your work and close the form.

**Note:**

When selecting business rules for composite forms, you can select which included forms’ business rules run in composite forms. Business rules from included forms don't run unless you select them for the composite form itself. For example, to run all the business rules associated with an included form named “Total Expense Impact”, select “Business rules for Total Expense Impact”.

---

**Setting Business Rule Properties**

You can specify whether business rules associated with forms launch automatically when users load (open) or save the form. If business rules have runtime prompts, you can set whether the default members in the runtime prompt match the members selected in the page and Point of View axes.

To set business rule properties:

1. **Take an action:**
   - To update the current form, click **Actions**, and then click **Business Rules**.
   - To open a form for editing, go to the Home page and click the **Navigator** icon and then under **Create and Manage**, click **Forms**. Select the form, click , and then click **Business Rules**.

2. **There are four phases in a form where a rule can be assigned and each may have information relevant to that phase. Select from the following:**
   - **Run Before Load**
   - **Run After Load**
   - **Run Before Save**
   - **Run After Save**
### Note:

- **Run After Load** and **Run Before Save** are enabled only for Groovy rules. You can't select these options for calc scripts or graphical rules. For more information about Groovy rules, see Using Groovy Rules.

- Some actions such as saving the grid, changing the page, or launching an action menu item will also reload the page after the operation completes. This ensures the grid contains the newest data. When this happens, the before and after load actions will run as they do in normal page loads.

- While hidden runtime prompts are not supported for calc script rules on the before and after load options, they are supported for Groovy rules.

- Groovy rules are not supported in composite forms.

- Regarding Groovy rules:
  - If you purchased EPM Cloud before June 4, 2019, you can use Groovy rules only for applications of type “Enterprise” (available with Enterprise PBCS or PBCS Plus One licenses), Oracle Strategic Workforce Planning Cloud, or Oracle Sales Planning Cloud.
  - If you purchased EPM Cloud Standard Edition after June 4, 2019, you can use the Groovy rules included with the modules.
  - If you purchased EPM Cloud Enterprise Edition after June 4, 2019, you can use the Groovy rules included with the modules. Additionally, you can edit and create Groovy rules in Custom, Module, and Free Form business processes.

3. **Optional:** If a business rule has runtime prompts, select **Use Members on Form** to match the default member selection on the runtime prompt window to the current members in the page and Point of View axes of the open form.

   To learn how this option interacts with other settings and conditions, see Understanding Runtime Prompts.

4. **Optional:** To hide the runtime prompt value from the user, select **Hide Prompt**, which automatically selects **Use Members on Form**.

   After saving the form, the next time you return to this page, **Use Members on Form** displays as selected.

   You can hide runtime prompts if:

   - All runtime prompt member values are filled in (appropriate dimension members can be read from form’s Page/Point of View)
   - No dimensions are repeated in the runtime prompt

5. Click **OK**.
Viewing Rules Usage

Business rules can be used in the following artifacts:

- Forms
- Rulesets
- Menus
- Task lists

Service Administrators can generate a report, called the Rules Usage Report, that details where rules are used. Filters enable you to refine the details by rule name, rule type, and cube.

**Note:**
The only rules listed in the Rules Usage Report are those rules that are deployed in Planning.

To view rules usage:

1. Click the Navigator icon, and then under Monitor and Explore, click System Reports.
2. Click the Rules Details tab.
3. If filtering, select filter criteria, and then click Apply Filter:
   - Rule Type—Choose one or more options: All, Rules, and Rulesets
   - Cube—Choose from a list of available cubes.
   - Name Filter—Enter a rule or ruleset name. Optionally, you can use wildcard characters.

   The Rules list displays the rules that are deployed in Planning and that match the filter criteria.

4. For File Format, select one of the following formats:
   - XLSX (default)
   - PDF
   - HTML
   - XML
Note:

Choosing XML format enables you to configure the Rules Usage Report with the help of Oracle BI Publisher Add-in for MS Office. You can have greater control over which columns to display in your report, the order in which the columns are displayed, whether to apply sorting, and so on.

5. Click Create Report.

The Rules Usage Reports provides the following information:

- Rule Name
- Rule Type
- Cube
- Primary Association (Type and Name)
- Secondary Association (Type and Name)

Primary association refers to the rule that is associated with the artifact directly; for example, a rule linked to a form or to a task. Secondary association refers to the rule that is associated with an artifact indirectly; for example, a rule that is linked to a menu item, and the menu item is linked to a form, or a rule that is linked to a task for a given task list.

Rules in the generated report are displayed in alphabetical order as follows:

- Rule name
- Primary association type
- Primary association name
### Example 18-1  Sample Rules Usage Report Showing Primary Associations (PDF Format)

#### Rules Usage Report

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>Rule Type</th>
<th>Cube</th>
<th>Primary Association</th>
<th>Secondary Association</th>
</tr>
</thead>
<tbody>
<tr>
<td>Act_Agg_Ek</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alloc - Clear 5-7</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agg</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td>Form</td>
<td>_XX_Agg</td>
</tr>
<tr>
<td>Agg - Roll up E and O from drop down. Customer in row, all others at none</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td>Form</td>
<td>1 - Corp Profit</td>
</tr>
</tbody>
</table>

### Example 18-2  Sample Rules Usage Report Showing Primary and Secondary Associations (PDF Format)

#### Rules Usage Report

<table>
<thead>
<tr>
<th>Rule Name</th>
<th>Rule Type</th>
<th>Cube</th>
<th>Primary Association</th>
<th>Secondary Association</th>
</tr>
</thead>
<tbody>
<tr>
<td>psp_Agg_budget</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>psp_Agg_Fact_All_Dims</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td>Task</td>
<td>Aggregation</td>
</tr>
<tr>
<td>psp_Agg_FC @AER_All_Dims</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>psp_Agg_budget_All_Dims</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>psp_Rowfocused_Agg</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td>Form</td>
<td>1.0 psp_Agg</td>
</tr>
<tr>
<td>psp_Agg_Actuals_All_Dims</td>
<td>Rules</td>
<td>PSPPlan1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
About Runtime Prompts

When launched, business rules can prompt users for such variables as members, text, dates, or numbers. Prompts should be specific and tell users what type of data is expected. For example:

- Select a month.
- Enter the expected number of customer visits per quarter.
- What percentage change in earnings do you expect next month?

If, in Calculation Manager, the option **Create dynamic members** is selected for the business rule and the parent member is enabled for adding dynamic children, then users can create new members by entering their name in the runtime prompt.

When launching business rules with runtime prompts, the application validates the value entered, but not the business rule. To set the default member selection in a runtime prompt, see Setting Business Rule Properties. To understand how other settings and conditions affect runtime prompts, see Understanding Runtime Prompts.

Understanding Runtime Prompts

The display and values of runtime prompts are affected by such aspects as:

- Whether the **Use as Override Value** property is set at the rule or the ruleset level at design-time
- Whether there are valid members on the form’s Page/Point of View and whether the **Use Members on Form** and **Hide Prompt** options on the **Business Rule Properties** tab are selected (see Setting Business Rule Properties)
- Whether the **Runtime Prompt** option is set during form design or set when designing the runtime prompt (see Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud)
- Whether the **Use Last Value** property is set when designing the business rule
- Whether, in Calculation Manager, the option **Create dynamic members** is selected for the business rule and the parent member is enabled for adding dynamic children. If yes, then users can create new members by entering their name in the runtime prompt.

Principles:

1. If the **Use as Override Value** property is set at the rule or the ruleset level at design-time, the value overridden at the rule level or the ruleset level will take precedence over the values of members in the Page/Point of View and the last saved value. This occurs regardless of where the rule is launched (from the form or from the **Rules** link on the **Navigator** menu), and regardless of whether the runtime prompt is hidden during design. The **Override Value** can be set as a user variable, in which case the rule will be launched with the current value of the variable.

2. When launched from a form, the values of members in the Page/Point of View take precedence over the last saved value if the **Use Members on Form** option is selected, regardless of whether the runtime prompt is hidden during design. The business rule is run without displaying the hidden runtime prompt to users, and the runtime prompt values are taken from the Page/Point of View members.
This isn't the case when business rules associated with composite forms are launched on save or from the left-hand pane or when business rules are launched from the Rules link on the Navigator menu. In these cases, the Use Members on Form setting is ignored, hidden runtime prompts get design-time values, and the last saved value takes precedence.

3. If the Use Last Value option is selected for the runtime prompt at design time, and if any of these conditions exist:
   - Use Members on Form isn't selected
   - A runtime prompt is launched from the Rules link on the Navigator menu
   - Values can't be pre-filled from the context

   Then the precedence of runtime prompt values is determined by:
   
   a. The last saved value takes precedence.
   
   b. If a ruleset is launched, the value overridden at the ruleset level at design-time is used.
   
   c. If a business rule is launched, the value overridden at the rule-level at design-time is used. If it's not overridden at the rule-level, the runtime prompt value at design-time is used.

   Runtime prompts that are hidden at design time never use the last saved value. In these cases, the Use Last Value setting is ignored.

4. The Use Members on Form and Hide Prompt options apply only to Member and Cross Dimension runtime prompt types (Cross Dimension runtime prompt types are available only for business rules created with Calculation Manager).

   The value set at the rule or the ruleset level at design-time when the Use as Override Value property is set participates in the Hide Prompt behavior.

5. For Cross Dimension runtime prompts: the runtime prompt isn't hidden unless all the prompts in the runtime prompt can be pre-filled from the Override Value or Page/Point of View. The runtime prompt is displayed with some values pre-filled from the Override Value or Page/Point of View and others filled according to Principles 1, 2 and 3.

   This table describes the result on runtime prompts of these settings and conditions:

<table>
<thead>
<tr>
<th>Availability of Override Value and member on the Page/Point of View</th>
<th>Use Members on Form option is selected</th>
<th>Hide Runtime Prompt property is set during runtime prompt design</th>
<th>Hide Prompt option is selected for the form</th>
<th>Result on Runtime Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use as Override Value is set and Override Value is available or the member is available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes or No Setting is ignored</td>
<td>The business rule runs without displaying the runtime prompt to users. Instead, the runtime prompt value is taken from the Override Value or Page/Point of View member.</td>
</tr>
</tbody>
</table>
Table 18-24  (Cont.) How Member Availability and Other Settings Affect Runtime Prompts

<table>
<thead>
<tr>
<th>Availability of Override Value and member on the Page/Point of View</th>
<th>Use Members on Form option is selected</th>
<th>Hide Runtime Prompt property is set during runtime prompt design</th>
<th>Hide Prompt option is selected for the form</th>
<th>Result on Runtime Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use as Override Value is set and Override Value is available or the member is available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>If all runtime prompts can be pre-filled from the Override Value or Page/Point of View context and are valid and within limits, the runtime prompts are not displayed. However, if even one runtime prompt value can't be pre-filled from the Override Value or Page/Point of View context, then all runtime prompts display, with values pre-filled wherever possible. All others follow Principles 1 and 3.</td>
</tr>
<tr>
<td>Use as Override Value isn't set and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>The runtime prompt is displayed to users, with values pre-filled according to Principle 3.</td>
</tr>
<tr>
<td>Use as Override Value isn't set and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes or No Setting is ignored</td>
<td>The business rule displays the runtime prompt to users, with values pre-filled according to Principle 3. For example, the form context can't be passed because the dimension of the runtime prompt is on rows or columns, so the Hide Prompt setting is ignored and the runtime prompt displayed.</td>
</tr>
</tbody>
</table>

The runtime prompt is displayed to users, with values pre-filled according to Principle 3.
Table 18-24 (Cont.) How Member Availability and Other Settings Affect Runtime Prompts

<table>
<thead>
<tr>
<th>Availability of Override Value and member on the Page/Point of View</th>
<th>Use Members on Form option is selected</th>
<th>Hide Runtime Prompt property is set during runtime prompt design</th>
<th>Hide Prompt option is selected for the form</th>
<th>Result on Runtime Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use as Override Value is set and Override Value is available, and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>If all runtime prompts can be pre-filled from the Override Value and are valid and within limits, the runtime prompts are not displayed. However, if even one runtime prompt value can't be pre-filled from the Override Value, then all runtime prompts display, with values pre-filled wherever possible. All others follow Principles 1 and 3.</td>
</tr>
<tr>
<td>Use as Override Value isn't set and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>The runtime prompt is displayed to users, with values pre-filled according to Principle 3.</td>
</tr>
<tr>
<td>Use as Override Value is set and Override Value is available, and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>The runtime prompt is displayed to users, with values pre-filled according to Principles 1 and 3.</td>
</tr>
<tr>
<td>Use as Override Value is set and Override Value is available or the member is available on the Page/Point of View to use as the runtime prompt value.</td>
<td>No</td>
<td>Yes</td>
<td>Not available</td>
<td>The business rule runs without displaying the runtime prompt to users. Instead, the design-time values are used.</td>
</tr>
<tr>
<td>Use as Override Value is set and Override Value is available or the member is available on the Page/Point of View to use as the runtime prompt value.</td>
<td>No</td>
<td>No</td>
<td>Not available</td>
<td>The runtime prompt is displayed to users, with values pre-filled according to Principle 3.</td>
</tr>
</tbody>
</table>
Table 18-24    (Cont.) How Member Availability and Other Settings Affect Runtime Prompts

<table>
<thead>
<tr>
<th>Availability of Override Value and member on the Page/Point of View</th>
<th>Use Members on Form option is selected</th>
<th>Hide Runtime Prompt property is set during runtime prompt design</th>
<th>Hide Prompt option is selected for the form</th>
<th>Result on Runtime Prompt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use as Override Value isn't set and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>No</td>
<td>Yes</td>
<td>Not available</td>
<td>The business rule runs without displaying the runtime prompt to users. Instead, the design-time values are used.</td>
</tr>
<tr>
<td>Use as Override Value isn't set and the member isn't available on the Page/Point of View to use as the runtime prompt value.</td>
<td>No</td>
<td>No</td>
<td>Not available</td>
<td>The runtime prompt is displayed to users, with values pre-filled according to Principle 3.</td>
</tr>
</tbody>
</table>

When hidden runtime prompt values are ambiguous, note:

- If the form context can't be passed in (because the dimension is on the row or column, for example), hidden runtime prompts are displayed.

- With hidden Cross Dimension runtime prompt types, if all prompts can't be passed in from the Override Value or context, the runtime prompt displays with values pre-filled from **Override Value** or context values and design time values. For example, if the Cross Dimension has runtime prompts for Period, Entity, and Scenario, and Entity is defined on the row and Scenario has an **Override Value** set, then the runtime prompt displays with the Override Scenario, then design time value for Entity, then Page Period.

- If the **Override Value** is present, or context can be passed in for the runtime prompt value but it's out of limits, then the runtime prompt is displayed with the context value pre-filled.

- If there is more than one runtime prompt of type Member or type Cross Dimension combined, then the runtime prompts are displayed with the **Override Value** or context values pre-filled. For example, if there is a member type runtime prompt for the Entity dimension and a Cross Dimension type runtime prompt with one prompt for the Entity dimension, then both runtime prompts are displayed. This rule doesn't apply to Calculation Manager rulesets.

- When launched from the **Rules** link on the **Navigator** menu, runtime prompts are hidden and the design-time value (overridden at the rule or ruleset level) is used to launch the business rule. If the provided design-time value is out of limits, then the runtime prompt is displayed with the design-time value pre-filled.

- Runtime variables that are hidden during design never use the last saved value. The **Use Last Value** property is ignored, and the values are not saved to the database.
About Runtime Prompts and Approvals Security

Administrators can design runtime prompts to honor Approvals security for members. Doing so prevents planners from changing data in approval units to which they don’t have access, according to Approvals rules. For example, the administrator may not want planners to change data after they have promoted the related approval unit. In Calculation Manager, administrators can set runtime Security for a member or members:

- **Approvals**—The application allows a user to change member data if both these conditions are true:
  - The user has write access to the members (as assigned in the application).
  - If the members belong to an approval unit, the user owns the approval unit.
  If both conditions are not met, the user can’t change the members’ data.
- **Write**—Users launching the business rule who have write access to the members (as assigned in the application) can change its data. The members’ Approvals status is ignored.
- **Read**—Users launching the business rule have read access to the members (as assigned in the application). Approvals status is ignored.
- **Use Default**—Security is applied to the runtime prompt only if member access isn’t set to None (that is, either read or write).

See [Designing Secure Runtime Prompts](#).

Designing Secure Runtime Prompts

The application supports runtime security on Scenario, Version, Entity, and secondary dimension intersections by relying on the order in which runtime prompts are designed. For the business rule to apply Approvals security, the Calculation Manager designer must place runtime prompts for Scenario and Version with Write or Approvals security before the Entity runtime prompt.

Runtime prompts for Version and Scenario members are filtered by write access when Security in Calculation Manager is set to either Approvals or Write. Runtime prompts for Entities with the Security set to Approvals are filtered according to the last Scenario/Version that is displayed before the Entity runtime prompt. If either the Scenario or Version runtime prompt doesn’t exist, Entities are filtered by write access.

Runtime prompts for other dimensions with Security set to Approvals are considered to be secondary dimensions and are filtered according to the last Scenario/Version/Entity that displays before that runtime prompt.

So, for runtime prompts using Approvals security defined in this order:

Scenario1: Version2: To_Entity1, Scenario2: Version1, To_Entity2, To_Product

Runtime prompts are filtered as follows:

- To_Entity1 is filtered by the combination—Scenario1: Version2
- To_Entity2 is filtered by the combination—Scenario2: Version1
- To_Product is filtered by the combination—Scenario2, Version1, To_Entity2, To_Product
For example:

Fix {FY11, Jan, {EntitySalesByCountry}, {MyScenario2}, {MyVersion2}}

Fix {{MyProduct}, {MyCountry}}

{ToAccount} = {FromAccount} * 2;

ENDFIX

Endfix

---

**Note:**

If the business rule designer omits an approval unit dimension from the list of runtime prompts (for example, the runtime prompt doesn't include a Scenario or Entity), then Approvals security isn't applied and the hierarchy is filtered by write access.

---

See [About Runtime Prompts and Approvals Security](#).

**Using Groovy Rules**

Oracle supports the creation of business rules written in the Groovy scripting language.
About Groovy Business Rules

Note:

- If you purchased EPM Cloud before June 4, 2019, you can use Groovy rules only for applications of type “Enterprise” (available with Enterprise PBCS or PBCS Plus One licenses), Oracle Strategic Workforce Planning Cloud, or Oracle Sales Planning Cloud.
- If you purchased EPM Cloud Standard Edition after June 4, 2019, you can use the Groovy rules included with the modules.
- If you purchased EPM Cloud Enterprise Edition after June 4, 2019, you can use the Groovy rules included with the modules. Additionally, you can edit and create Groovy rules in Custom, Module, and Free Form business processes.
- Groovy rules are not supported in composite forms.

Groovy business rules allow you to design sophisticated rules that solve use cases that normal business rules can't solve; for example, rules to prevent users from saving data on forms if the data value is above a predefined threshold.

You create Groovy rules in Calculation Manager and execute them from any place that you can execute a calc script rule in a Planning application; for example, on the Rules page, within the context of a form, in the job scheduler, in dashboards, in task lists, and so on.

Groovy rules are also supported in rulesets. You can have a combination of calc script rules and Groovy rules within a ruleset.

You can execute jobs of type rules, rulesets, and templates synchronously from a Groovy rule.

Oracle supports two types of Groovy rules:

- Rules that can dynamically generate calc scripts at runtime based on context other than the runtime prompts and return the calc script which is then executed against Essbase.
  
  For example, you could create a rule to calculate expenses for projects only for the duration (start and end dates) of the project.

  Another example is a trend-based calculation that restricts the calculation to the accounts available on the form. You could use this calculation for various forms in Revenue, Expense, Balance Sheet, and Cash Flow. This allows for optimization and reuse.

- Pure Groovy rules that can, for example, perform data validations and cancel the operation if the data entered violates company policies.
Java API Reference for Groovy Rules

For Enterprise applications, a Java API Reference is available to use as you create Groovy rules. The Java API Reference includes examples that demonstrate the syntax and powers of the EPM Groovy object model.

To view the Java API Reference, see the Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules on the cloud help center. You can also access this reference from the Oracle Enterprise Planning and Budgeting Cloud Academy. To access the Academy, sign in, and then click Academy.

Groovy Business Rule Examples

To see example Groovy scripts:


2. Do one of the following:
   - Under Example Groovy Scripts on the main page, click the word "here" to view sample scripts:

     Example Groovy Scripts

     The example Groovy scripts provided here demonstrate the syntax and power of the EPM Groovy object model.

     [Overview] [Package] [Class] [Tree] [Index] [Help]

   - Under All Classes in the left pane, click a class to see the examples for that class.

     For example, to see Strategic Modeling examples, click the StrategicModel class in the left pane.

Groovy Business Rule Tutorial Videos

Watch this tutorial video to learn best practices in moving modified data using Groovy rules and Smart Push.

Moving Modified Data Using Groovy Rules and Smart Push

Watch this tutorial video to learn how to improve calculation performance on Planning forms by creating context-specific, dynamic business rules using the Groovy scripting language.

Calculating Modified Data Using Groovy Rules
Groovy Rule Business Scenarios

This section provides examples of business scenarios where you can use Groovy business rules.

These topics contain scenarios that show you how to use Groovy rules to perform focused data movement and smart calculations on planning data:

- Moving Modified Data Using Groovy Rules and Smart Push
- Calculating Modified Data Using Groovy Rules

Moving Modified Data Using Groovy Rules and Smart Push

In this topic, we’ll show you best practices on how you can use Groovy rules and Smart Push to move modified data into a reporting cube.

On a data form, planners can modify employee salary and assign a Reporting Manager. After the form data is saved, Smart Push is run to copy data to the reporting cube for analysis. Then, dimensions are also consolidated, if necessary, as the data is being copied.

If the form contains hundreds of rows of data, all of that data is checked against the configured data map, processed for dimension consolidation, if any is set, and then pushed to the reporting cube. Depending on the amount of data on your form, this process may take a while to complete.

You can improve the performance of data synchronization between your input and reporting cubes by designing a Groovy rule that identifies and isolates the data you modified, and pushes only the isolated data into your reporting cube.

For example, we configured a simple data map for Smart Push that maps dimension members from our input cube to our reporting cube.

Then we prepared forms that display data from our input and reporting cubes.
The form used for data entry has Smart Push enabled and uses the form's context to push data for all the members selected for the dimensions defined on the form.
For the Groovy rule, here’s the entire script.

```groovy
if (operation.grid.hasSmartPush("PushCompensationToRepCube")) {
    return;
}

Set<String> employees = []
operation.grid.dataCellIterator('Salary', 'Reporting Manager').each { DataCell cell ->
    if (cell.getEditName().equals("Employee")) {
        employees << cell.getName()
    }
}

if (employees.size() > 0) {
    String employeesStr = employees.join(",")
    operation.grid.getCellData('PushCompensationToRepCube').execute([ "Employee" : employeesStr ])
    println("The following employee compensation details were moved to reporting cube: ${employeesStr}")
} else {
    println("There were no modifications. No data was moved to the reporting cube.")
}
```

This section of the script uses the `dataCellIterator` method to identify edited cells and isolates them.
Then, this section takes the isolated data and runs the focused data movement using Smart Push and the data map configured and associated with the form. A message displays when modified data is moved to the reporting cube.

If there are no modifications, a message displays when there are no changes and data was not moved to the reporting cube.

After we created this Groovy rule, we deployed it the same way we deploy business rules.
Then, in form management, associated it with the form. The Groovy rule runs after the form data is saved.
To run the Groovy rule, open the form in **Input Cube**, make some changes, and then click **Save**.

Once saved, the updated data will display in the reporting cube.

**Calculating Modified Data Using Groovy Rules**

In this topic, we'll show you how to improve Planning calculation performance by using Groovy rules to calculate only the data that's been modified.

After you update data in a form, it needs to be calculated. When you use a business rule to calculate a standard calculation script, all the data for the form is calculated. With large forms or calculations that affect large areas of the database, you might see slower calculation performance.
You can manually calculate one row at a time by right-clicking on the row. This can provide faster calculation performance, but isn’t a good option when making changes to multiple rows of data.

Groovy rules can dynamically focus on the cells that you update.

You can use Groovy rules on forms in two ways:

- When you calculate data on save, the Groovy rule generates a calculation script with a FIX statement based on the cells you updated. Then the calculation is run on the server against the modified data instead of all records on the form. As a best practice, use this option with larger forms where you are updating a lot of data.

- When you calculate data before saving, the Groovy rule calculates modified data in memory and displays the results on the form before saving them to the database. As a best practice, use this option when you have a small number of calculations to perform, or when you want to perform complex procedural calculations that are difficult to build into the outline.

In this example, bonus calculations are calculated as a percent of salary.

Let’s look at calculating on save. Notice that this form is defined with a Groovy rule to calculate on save.
After updating some of the salary data and some of the phone numbers on the form, the phone numbers are saved but don’t affect the calculation.

When you save the form to run the associated Groovy rule, the calculated bonus data is displayed on the form.
Viewing the job details, here’s the calculation script that the Groovy rule created. Notice that the FIX statement includes only the modified employees.

Let’s open Calculation Manager to see the Groovy rule. This first section iterates over the grid in the form to find Salary cells that have been edited and collects the related employee names.
This section generates the calculation script; it finds the POV members from the form, then creates a FIX statement on the POV members and the edited employees.

Now let's take a look at calculating before save.

This is the same form, but now it's defined with a different Groovy rule to calculate before save.
After updating the salary data and some of the phone numbers again, save the form to run the associated Groovy rule. The calculated bonus data is displayed on the form, but when you look at the job details, you can see that some of the bonus data was updated, but this time no calculation script was generated.

Open the Groovy rule in Calculation Manager.

Similar to the first rule, the script iterates over the grid in the form to find Salary cells that have been edited and collects the related employee names. Then it calculates bonus data for just the edited employees. Notice that no Essbase calculation script is generated.
Groovy rules also work with Planning forms in Oracle Smart View for Office. Smart View gives you all the power of Microsoft Excel to make mass adjustments your data, and any Groovy rules for the form set to run before or after saving are executed when you submit data.

Assigning Access to Rules

To assign access to rules:

1. Assigning Access to Rules
2. Adding, Editing, and Removing Access to Rules

Assigning Access to Rules

To assign access to rules:
1. Click the Navigator icon, and then under Create and Manage, click Rules Security.

2. Under Business Rule Folders, select the folder containing the rules, and then select the rules.

3. Click .

4. Add, edit, or remove access.

   See Adding, Editing, and Removing Access to Rules.

Adding, Editing, and Removing Access to Rules

You can specify which users and groups can access the selected rule.

Note:

Rulesets inherit launch permissions from the rules included in the ruleset.

To assign, edit, and remove access permissions to rules:

1. Click the Navigator icon, and then under Create and Manage, click Rules Security.

2. Under Business Rule Folders, select the folder containing the rules, and then select the rules.

3. Click .

4. Perform a task:

   • To add access, click , and then select from the list of available users and groups.
     For Type of Access:
     – Select Launch to allow the selected users and groups to launch the selected rules.
     – Select No Launch to prevent the selected users and groups from launching the selected rules.

   • To edit access, click , and then select the applicable Type of Access.

   • To remove access, select the users or groups for which to remove access, and then click .

Administering Smart Lists

Smart Lists are custom drop-down lists that users access from form cells.

Related Topics

• Working with Smart Lists
• Setting Smart List Properties
• Defining Smart List Entries
• Previewing Smart Lists
• Displaying #MISSING with Smart Lists

Working with Smart Lists

Administrators use Smart Lists to create custom drop-down lists that users access from form cells. When clicking in cells whose members are associated with a Smart List (as a member property), users select items from drop-down lists instead of entering data. Users can't type in cells that contain Smart Lists. Smart Lists display in cells as down arrows that expand when users click into the cells.

Perform these tasks to create and administer Smart Lists:

• Define Smart Lists.
• Associate Smart Lists with members.
• Select dimensions for which Smart Lists are displayed.
• Optionally:
  – Use Smart List values in member formulas.
  – Set how #MISSING cells associated with Smart Lists display in forms.
  – Synchronize Smart Lists in reporting applications

Note:

For a Smart List to be viewable in forms, the dimension to which the Smart List type member belongs must be first in the evaluation order. See Managing Dimensions.

To create or work with Smart Lists:

1. Click the Navigator icon , and then under Create and Manage, click Smart Lists.

2. Perform one action:

   • To create a Smart List, click , and then set Smart List properties. See Setting Smart List Properties.

   • To edit a Smart List, select it, click , and then set Smart List properties. See Setting Smart List Properties.

   • To delete Smart Lists, select them, click , and then click OK. Deleting Smart Lists also deletes any associated mappings with dimension members and reporting applications.

Data cells can display only one Smart List. If multiple Smart Lists intersect at cells, set which one takes precedence.
• **Optional**: Click to synchronize Smart Lists between an application that uses Planning application administration and a reporting application. See Synchronizing Smart Lists in Reporting Applications.

Setting Smart List Properties

Define Smart List properties on the Properties tab.

![Note:](image)

For a Smart List to be viewable in forms, the dimension to which the Smart List type member belongs must be first in the evaluation order. See Managing Dimensions.

### Table 18-25  Smart List Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart List</td>
<td>Enter a unique name containing only alphanumeric and underscore characters (for example: Position) and no special characters or spaces. Smart List names can be referenced in formula expressions.</td>
</tr>
<tr>
<td>Label</td>
<td>Enter the text to display when the Smart List is selected. Spaces and special characters are allowed.</td>
</tr>
<tr>
<td>Display Order</td>
<td>How Smart Lists are sorted in the drop-down list: by ID, Name, or Label</td>
</tr>
<tr>
<td>#MISSING Drop-Down Label</td>
<td>Enter a label (for example, &quot;No Justification&quot;) to be displayed as an entry in the Smart List whose value is #MISSING. Note the following:</td>
</tr>
<tr>
<td></td>
<td>• It displays as the first selection in the Smart List drop-down, allowing #MISSING as a selection in the form.</td>
</tr>
<tr>
<td></td>
<td>• When the cell isn’t in focus, this label displays only if Drop-Down Setting is selected in the next option. Otherwise, #MISSING or a blank cell is displayed, depending on the Display Missing Values As Blank selection for the form.</td>
</tr>
<tr>
<td></td>
<td>• #MISSING labels determine only the display of cells with #MISSING data; #MISSING remains the stored value.</td>
</tr>
</tbody>
</table>
Table 18-25  (Cont.) Smart List Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#MISSING Form Label</td>
<td>Determines how #MISSING values are represented in cells associated with Smart Lists. Options:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Drop-Down Setting</strong>—Displays the label set in #MISSING Drop-Down Label.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Form Setting</strong>—Displays #MISSING or leaves cells blank, depending on the <strong>Display Missing Values As Blank</strong> selection for the form. This selection determines what is displayed in the cell when it's not the focus. When the cell is in focus, the Smart List item that is selected from the drop-down is displayed.</td>
</tr>
<tr>
<td>Automatically Generate ID</td>
<td>Generate a numeric ID for each Smart List entry. If you don't select this option, you can customize Smart List ID values.</td>
</tr>
<tr>
<td>Create from Members</td>
<td>Create a Smart List based on dimension hierarchies. Smart List values are dynamically updated when members are updated.</td>
</tr>
</tbody>
</table>

**Note:**

User security for the dimension is honored for Smart Lists created from the dimension's hierarchy.

Member Selection

Manually select the members to use in Smart Lists.

Use the **Entries** tab to define selections on Smart Lists. See Defining Smart List Entries.

Defining Smart List Entries

To define Smart List entries:

1. On the **Entries** tab, define drop-down list items:
   - **For first items only**: enter information into the first row.
   - To add an item, click + and enter the information.
   - To delete an item, select it and click -.
   - To edit an entry, change the information in its row:
Table 18-26  Smart List Entries

<table>
<thead>
<tr>
<th>Entry Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Unique number that sets the order for the displayed entry. The ID is customizable only if Automatically generate ID isn’t selected on the Properties tab.</td>
</tr>
<tr>
<td>Name</td>
<td>Unique alphanumeric name containing alphanumeric and underscore characters (for example: Customer_Feedback) and no special characters or spaces</td>
</tr>
<tr>
<td>Label</td>
<td>Displayed text for the Smart List entry on the drop-down list (for example: Customer Feedback).</td>
</tr>
</tbody>
</table>

Note: Items highlighted in red are duplicates.

2. Click Save.

Use the Preview tab to preview the defined Smart List. See Previewing Smart Lists.

Previewing Smart Lists

Preview the defined Smart List on the Preview tab. The tab shows the Smart List as displayed in a drop-down list or a table.

To preview a Smart List:

1. Click the Navigator icon and then under Create and Manage, click Smart Lists.
2. Select a Smart List and click .
3. Click Save.

Displaying #MISSING with Smart Lists

Administrators set values displayed in Smart Lists and data cells, including the display when no data is in the cell. Cells can display no value, #MISSING, or (for cells associated with Smart Lists) a specified value.

Use these options to control the display of #MISSING when cells are not in focus:
Table 18-27  Display Options for #MISSING

<table>
<thead>
<tr>
<th>Option</th>
<th>Guideline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blank</td>
<td>When designing forms, select <strong>Display Missing Values as Blank</strong>. When setting Smart List properties, select <strong>Form Setting</strong>.</td>
</tr>
<tr>
<td>#MISSING</td>
<td>When designing forms, don't select <strong>Display Missing Values as Blank</strong>. When setting Smart List properties, select <strong>Form Setting</strong>.</td>
</tr>
<tr>
<td>A custom label, such as &quot;No Change&quot;</td>
<td>When setting Smart List properties, enter the custom label in the <strong>#MISSING Drop-Down Label</strong> field (for example, No Change). Select <strong>Drop-Down Setting</strong>.</td>
</tr>
</tbody>
</table>

Administering Task Lists

Guide users through the planning process with task lists. Task lists guide users through the planning process by listing tasks, instructions, and end dates. Administrators and interactive users create and manage tasks and task lists.

- Working with Task Lists
- Adding Instructions to Task Lists
- Adding Tasks to Task Lists
- Editing Task Lists
- Assigning Access to Task Lists

Working with Task Lists

Task lists organize groups of tasks for users. You must create task lists before creating tasks.

To create and rename task lists:

1. Click the **Navigator** icon 📉, and then under **Create and Manage**, click **Task Lists**.
2. Perform a task:
   - To create a task list, click 📉.
   - To rename a task list, select the task list, then click 📉.
   - To move a task list, select the task list, then click 📈.
   - To remove a task list, select the task list, then click ✗.
3. Click OK.

To define the task list, see:
• Adding Instructions to Task Lists.
• Adding Tasks to Task Lists.

Adding Instructions to Task Lists

To add instructions to task lists:

1. Click the Navigator icon , and then under Create and Manage, click Task Lists.

2. For Task List, select the task list to modify, and then click .

3. Click the Instructions tab.

4. Enter instructions for the task list.

5. Click Save, and then click Close

Adding Tasks to Task Lists

You can set completion dates and alerts for tasks that users perform at runtime. You can also set up email messages, for example, to alert users that a task was not completed by its end date. Alert messages are sent after an "alert date" that you set, and are repeated until the end date is reached for a task.

To add tasks to task lists:

1. Click the Navigator icon , and then under Create and Manage, click Task Lists.

2. Perform a task:
   • Define a task list by clicking , and then entering a name.
   • Modify a task list to add a new task by selecting it, and then clicking .

3. Click .

4. For Task, enter a task name.

5. From Type, select the kind of task you're creating and what it should enable users to do. For example, if you're creating a task that enables administrators to modify all dimensions at runtime, select Dimension Editor.

   Specify the information required for the task type using this table:
### Table 18-28  Task Information

<table>
<thead>
<tr>
<th>Task Type</th>
<th>Action</th>
</tr>
</thead>
</table>
| **URL**     | Opens a specified URL  
   Enter a fully qualified URL to associate with  
   this task, such as http://www.company_name.com, and then go to step 6. |
| **Form**    | Opens a form  
   Select the form for users to complete, and  
   then go to step 6.  
   **Optional**: Select **Set Page Member Defaults** to select the member from each  
   dimension to display as the default when the task is first opened. After you select this  
   option, you can select the members for the page dimensions. The page member defaults apply  
   until a user updates the form and returns to the task in another session.  
   Where page member defaults are set, they override the most recently used settings in  
   each session. |
| **Dashboard** | Opens a dashboard  
   Select the dashboard for users to work with, and then go to step 6. |
| **Business Rule** | Launches a business rule  
   Perform these tasks:  
   - From **Cube**, select the cube associated with the business rule to execute.  
   - From **Business Rules**, select the business rule to execute.  
   - Go to step 6. |
| **Manage Approvals** | Starts the review process with a specified scenario and version  
   Specify the scenario and version in which the user can start the approvals process,  
   and then go to step 6. |
| **Descriptive** | Enables administrators to add a task with no task properties. |
| **Copy Version** | Copies the current form’s data, including supporting detail, annotation, cell text, and  
   cell documents, for the specified source and destination versions  
   Select values for **Scenario**, **Copy From**, and **Copy To**, and then go to step 6. |
Table 18-28  (Cont.) Task Information

<table>
<thead>
<tr>
<th>Task Type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Console</td>
<td>Opens the Job Console to enable users to view a list of their jobs by type (such as copy data and push data) and by completion status</td>
</tr>
<tr>
<td></td>
<td>Perform these tasks:</td>
</tr>
<tr>
<td></td>
<td>• In <strong>Job Type</strong> select the kind of job to be displayed in the console.</td>
</tr>
<tr>
<td></td>
<td>• In <strong>Status</strong> select the status of the task to be displayed in the console.</td>
</tr>
<tr>
<td></td>
<td>• Go to step 6.</td>
</tr>
<tr>
<td>Dimension Editor</td>
<td>Enables administrators to view and modify a dimension at runtime</td>
</tr>
<tr>
<td></td>
<td>Perform these tasks, and then go to step 6:</td>
</tr>
<tr>
<td></td>
<td>• To enable administrators to view and modify all dimensions, select <strong>All</strong> from <strong>Dimension</strong>.</td>
</tr>
<tr>
<td></td>
<td>• To enable administrators to view and modify a particular dimension, select it.</td>
</tr>
<tr>
<td>Refresh Application</td>
<td>Enables administrators to refresh the application at runtime to capture recent changes. After enabling, go to step 6.</td>
</tr>
<tr>
<td>Import and Export</td>
<td>Enables administrators to import and export data and metadata using flat files.</td>
</tr>
<tr>
<td></td>
<td>From <strong>Load Method</strong>, specify what import or export task the user must perform, such as export data to a file or import metadata from a file, and then go to step 6.</td>
</tr>
</tbody>
</table>

**Note:**

The Security Management, Process Management, and Form Management task types cannot be added to a task list or edited in the Simplified Interface. These task types are accessible only in desktop mode of the application.

The Dashboard task type cannot be added to a task list or edited while in desktop mode. This task type is accessible only in the Simplified Interface.

6. For **Duration**, enter how long you expect it will take to complete the task. For example, if you think the task will take two weeks, enter 2 in the first field and select week(s) in the second field.

7. To ensure that the task is started on a certain date and then completed by a certain date, perform these tasks:

   a. For **Start Date**, click 📅, and then use the calendar to specify the year, month, day, and time by which the task must begin.
b. For **End Date**, click ⌚, and then use the calendar to specify the year, month, day, and time by which the task should be completed.

To send email messages that prompt users to finish uncompleted tasks, select **Repeat Every**, and then specify how often to send these emails, such as once a day.

8. To send email messages that prompt users that the date by which they must complete the task is approaching, select **Alert**, and then:
   a. Click ⌚ to use the calendar to specify the date and time on which to start sending alert reminders.
   b. Click **Repeat Every** and specify the frequency for email reminders.

For example, if a task's end date is June 9 2014 at 5 PM, and you want to remind users once a day starting the week before the task is due, select June 2 2014 at 5 PM, enter 1, and then select Weeks.

9. To make task completion dependent on completing another primary (or parent) task, click ⬇️ next to **Dependency**, select the primary task, and then click **OK**.

---

**Note:**

You cannot make task completion dependent on completing a child task of another primary task. Dependencies across task list hierarchies are not supported when a task has children. For example, consider a case where Task 3 is dependent on Task 2 and Task 4 is, in turn, dependent on Task 3. When Task 3 is incomplete, Task 4 becomes incomplete automatically. However, when Task 2.1 is incomplete, its parent task (Task 2) changes to incomplete, but this is not recursive (Task 3 and Task 4 are not affected).

- Task1
- Task2
  - Task2.1
  - Task2.2
- Task3
- Task4

10. In the **Instructions** pane to the right, enter information that explains how to complete the task. Use the formatting buttons to customize the information. For example, you can modify the font, font size, and layout (as bullet points or numbered steps, for example).

11. Click **Save** and then click **OK**.

**Editing Task Lists**

**Related Topics**

- Editing Tasks
- Copying and Moving Tasks
• Moving and Reordering Task Lists
• Clearing Task Lists
• Deleting Tasks and Task Lists

Editing Tasks

Use the Edit Task dialog box to modify the type of task, its instructions, the end date and alert date, and email reminder messages.

To edit tasks:

1. Click the Navigator icon, and then under Create and Manage, click Task Lists.

2. Select a task list, and then select .

3. Select a task, and then select .

4. In Edit Task:
   • Modify the task name.
   • Select another task type.
   • Optional: Modify the values for Duration, Start Date, End Date, Alert, and Dependency options.

For descriptions of the various options, see Adding Tasks to Task Lists.

5. For Task - Instructions, modify instructions for completing the task.

6. Click Save.

Copying and Moving Tasks

To copy or move tasks:

1. Click the Navigator icon, and then under Create and Manage, click Task Lists.

2. Perform an action:
   To copy tasks:
   a. Select a task list, and then select .
   b. Select the task to copy, and then click Save As.
   c. Enter the name for the new task list, and then click OK.
   To move tasks:
   a. Select a task list, then select .
   b. Select a task, and then click .
   c. Select the task that will be a child of another task, and then select .
d. Select the task that will be the parent task, and then select \( \text{父} \).
e. Click **Save**.

To cut and paste tasks:

a. Select a task list, and then click \( \text{剪刀} \).
b. Select a task, and then click \( \text{剪刀} \).
c. To move the task to a new position, select the task to appear above it.
d. Click \( \text{移动} \).
e. Click **OK**.

### Moving and Reordering Task Lists

To move or reorder task lists:

1. Click the **Navigator** icon \( \text{文件夹} \), and then under **Create and Manage**, click **Task Lists**.
2. Perform a task:
   - To move a task list:
     a. Select a task list, and then click \( \text{剪刀} \).
     b. Select the destination folder.
     c. Click **OK**.
   - To reorder task lists:
     a. Select a task list.
     b. Click \( \text{上移} \) or \( \text{下移} \).

### Clearing Task Lists

After a task list is completed, you can clear the completion status, end dates, and alerts for all tasks within a selected task list. This allows tasks to be reused for a future planning period.

Alerts are enabled only when task lists contains end dates. Clearing task lists disables alerts by clearing the check boxes for end dates. It doesn't delete the dates.

To clear task lists:

1. Click the **Navigator** icon \( \text{文件夹} \), and then under **Create and Manage**, click **Task Lists**.
2. Select the task list you want to clear, and then click \( \text{剪刀} \).
3. In the **Clear** drop-down list, select an option:
   - **Completion Status**—Clears completion status only
Due Dates and Alerts—Clears any alerts that are enabled, based on the end date set for the task

Both—Clears completion status, end dates, and alerts

4. Click 
5. Click OK.

Deleting Tasks and Task Lists

To delete tasks and task lists:

1. Click the Navigator icon and then under Create and Manage, click Task Lists.
2. Perform a task:
   - To delete a task, select the task list with the task to delete, and then click .
     Select the tasks to delete, and then click .
   - To delete a task list, select the task list to delete, and then select .
3. Click OK.

Assigning Access to Task Lists

You can determine who can view and modify task lists. By default, administrators can manage and assign access permissions for task lists.

Note:

Being assigned to a task list means being able to access and complete tasks in the task list. It doesn't mean being able to assign tasks to someone else.

- Adding Access to Task Lists
- Changing and Removing Access to Task Lists

Adding Access to Task Lists

To assign access to task lists:

1. Click the Navigator icon , and then under Create and Manage, click Task Lists.
2. Select the task list to modify.
3. Click  
4. Click  
5. Select the users or groups to access the task list.
• Click **Users** to display all user names; click **Groups** to display all groups.

• If there are multiple pages of users and groups, type the page number to go to in **Page**, and click **Go**.

• Click **Start** or **End** to navigate to the first or last page.

• Click **Prev** or **Next** to move to the previous or next page.

6. For **Type of Access**, select how users or groups can use the task list:
   • **Assigned**—View and use
   • **Manage**—Modify
   • **Manage and Assign**—View, use, and modify
   • **None**—No access

7. Click **Add**.

8. Click **Close**.

### Changing and Removing Access to Task Lists

To change or remove access to task lists:

1. Click the **Navigator** icon, and then under **Create and Manage**, click **Task Lists**.

2. Select the task list to modify, then click ✎.

3. Perform a task:
   
   To change access to task lists:
   
   a. Select users or groups, and click ✒.
   
   b. For **Type of Access**, select:
      • **Assigned**—View and use
      • **Manage**—Modify
      • **Manage and Assign**—View, use, and modify
      • **None**—No access
   
   c. Click **Set**.

   To remove access to task lists:
   
   a. Select the user or group, and click ✗.
   
   b. Click **OK**.

4. Click **Close**.

### Clearing Cell Details

You can clear these cell details for a cube: account annotations, supporting detail, cell text, and cell-level documents. For information on creating and viewing account annotations, supporting detail, cell text, and cell-level documents, see *Working with Planning*. 
To clear cell details:

1. Click the **Navigator icon** and then under **Actions**, click **Clear Cell Details**.
2. For **Cube**, select a cube.
3. To select all dimensions in the cube, click **Add All Dimensions**. Select members for the displayed dimensions.
4. Specify the type of information to delete by selecting at least one option from **Clear Options**.
5. Select members for the data intersections:
   a. In **Dimension**, select at least one dimension with details to delete.
   b. For displayed dimensions, click **.** On the **Member Selection** page, make a selection that includes the details to delete.

**Note:**

For every dimension selected, you must select at least one member. If a dimension isn't selected, the application includes all its members when clearing cell details.

**Caution:**

To clear exactly the data you want, select at least one Account member and members from all other dimensions. If you don't select at least one Account member, no data is cleared, even if you select the **Yes, set value to #Missing** option. If you don't select members from the other dimensions, all data for the selected Account member is cleared across all other dimensions.

6. **Optional**: To select another dimension so you can select its members, click **.**
7. Click **Clear**.
   A Confirmation page displays your selections.
8. Click **Finish** to proceed, or **Back** to change your selections.

If Clear Cell Details is successful, data is deleted from the cube. If data isn't deleted successfully, a message is displayed. You can also check the log file.
9. **Optional**: To view the execution status of Clear Cell Details and review the information that was deleted, view the Job. See Managing Jobs.

10. To clear cell details for another cube, select another cube in step 2, and repeat the procedure.

---

**Copying Data**

You can copy plans from one dimensional intersection to another, including relational data and supporting detail. For example, you can copy *Budget, FY10, Final* to *Forecast, FY11, First Draft*.

### Note:

- Selected Copy Data settings are preserved for the current session only.
- Copied dimension members must be present in the selected cubes.
- Data must be copied into cells that can accept data. For example, you can't copy data into read-only or dynamic cells.
- You can copy account annotations, supporting detail, and cell text. You can't copy approval unit annotations.
- You can't use this feature with attributes, so don't select attributes to be copied.
- Because this is an administrative function, the application assumes you have full access to data you copy. You're not prevented from copying to approval units that are approved.
- This feature doesn't calculate data. To perform calculations, such as increasing the forecast by 5%, apply the business rule after copying data.
- For Copy Data to be successful, you must select at least one member for Scenario, Account, Entity, Period, and Version dimensions.

To copy data:

1. Click the **Navigator** icon, and then under **Actions**, click **Copy Data**.
2. In **Cube**, select a cube.
   - Dimensions are displayed for this cube. You can only copy from one cube at a time.
3. In **Copy Data Options**, select the type of information to copy.
4. Under **Static Dimensions**, select the members for the data intersections:
   - **a.** For **Dimension**, select a dimension from which to copy.
   - **b.** For **Members**, click to make a selection from which to copy. You can select multiple members. You must select at least one member for Scenario, Account, Entity, Period, and Version dimensions.
5. **Optional:** To add another static dimension to the list, click and select dimension members.

   **Note:**
   
   To remove a dimension, select **Select Dimension.** The dimension moves to the **Dimensions with Source and Destination** area.

6. In **Dimensions with Source and Destination,** enter dimensions into which to copy data.

   In **Source** and **Destination** click .

7. Click **Copy Data.**

   Data is copied from one intersection to the other. If data isn't copied successfully, a message is displayed. You can also check the log file.

8. To copy data for another cube, select another cube in step 2, and repeat the procedure.

**Administering the Application Monitor**

**Related Topics**
- About the Application Monitor
- Assumptions
- How the Application Monitor Works
- Using the Application Monitor Graphs
- Launching the Application Monitor
- Modifying Artifacts for Optimal Performance

**About the Application Monitor**

The application monitor enables administrators, at design-time, to identify and resolve design flaws before an application is placed in production (and on an ongoing basis, as the application evolves with new members and data is added to the application) and used by end users. Administrators can use the application monitor to evaluate the following:

- An entire application
- Types of artifacts such as forms and approval units
- Individual "cherry picked" artifacts

The application monitor identifies changes that administrators can make at design-time to resolve artifacts with design flaws, and provides easy access to the editors required to modify them. At runtime, internal processes perform checks that may deny system request if artifacts need to be modified. The application monitor evaluates your application and artifacts based on the user an administrator selects. This enables you to determine the potential problems that may occur at runtime given the individual user's variables and security access as part of the analysis.
Assumptions

Oracle can only provide general guidelines about how to design your application and artifacts to ensure reasonable performance. Oracle can't specify precise value ranges for artifacts and operations.

How the Application Monitor Works

The application monitor evaluates applications as follows:

- At design-time—Analyzes the application or specific artifacts in terms of all possible problems that could occur. To determine this worst possible situation, suppression options (such as those used with forms) are ignored. The application monitor provides information about how to fix the design flaws it encounters.
- At runtime—Internal governors perform checks that may prevent users from performing particular actions if the design flaws found are not resolved.

Use the application monitor to analyze the following:

- Block storage cubes
- Dimensions
- Simple forms
- Standalone business rules
- Approval unit hierarchies
- Reporting mappings
- Export data functionality

Using the Application Monitor Graphs

Artifacts that may cause performance issues and that should be modified are yellow, and those that will cause performance issues and must be redesigned are red. Use the pie graphs as follows:

- **Planning Artifacts Distribution**—Depicts how much of your application is comprised of various artifact types. Place your mouse over portions of the graph to view how many artifacts of each type were analyzed by the application monitor. For example, if you mouse over forms and the pop-up value is 55, then 55 forms were analyzed. Click each portion of the graph to view the compliance status of each artifact type. This displays another pie graph that represents how many artifacts of each type are in acceptable, warning, or error status.

Artifact Distribution Graph:
Planning Artifact(s) Distribution

- **Filtered For <Artifact>**—Click the graph portions that represent the artifact status, and to view and drill down on artifacts with a particular status. For example, to view information for only approval unit hierarchies in error status, click the red portion of the graph.

Filtered Graph:

Launching the Application Monitor

To launch the application monitor:

1. Click the **Navigator** icon 📚, and then under **Monitor and Explore**, click **Application Diagnostics**.

2. In **Select User**, click 🔄 to select the user whose settings you want to use to perform the analysis, and then click 🔄.

3. Perform a task:
4. Click or select Actions, and then Run Application Monitor.

5. If the View Details column doesn't display in the grid, select View, then Columns, and then View Details. This also enables you to display or conceal different types of artifact data.

6. To display information about how to fix artifacts, click in View Details. Then click artifacts to launch editors that enable you to modify them.

Modifying Artifacts for Optimal Performance

After reviewing the details provided by the application monitor that tell you how to modify your artifacts, edit them as follows:

• Simple forms—Clicking on the form artifact name launches the Form Designer in a new tab. Edit forms as described in Administering Forms.

• Business Rules—Clicking a business rule name launches the System View which displays all application artifacts that you can access. You can perform tasks such as:
  – Use the upper Rule Designer to view the individual components of the rule, such as conditions, commands, and scripts. Click each component to modify them in the lower portion of the page. You can also insert and remove components like formulas, system templates, and scripts by dragging and dropping them from the Rule Pallet and Existing Object panes to the left.
  – From Designer, select Edit Script to modify and format the script by performing tasks such as inserting functions, editing templates, inserting variables, inserting member ranges, use comments, and validate syntax.
  – Create, open, delete, and refresh artifacts, pages, and other documents.
  – Import and export business rules and other objects.
  – Determine how and where artifacts are used.
See *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud* and *Understanding Runtime Prompts*.

- Approval unit hierarchies—Clicking an approval unit hierarchy artifact name launches the **Approval Unit Hierarchy Designer** in a new tab. Edit approval units as described in *Creating Approval Unit Hierarchies*.

- Cubes—Clicking the cube artifact name launches the **Performance Settings** tab of the dimension editor in a new tab. Edit the dimensional outline or dimensional sparsity as described in *Editing Dimensions in the Simplified Dimension Editor*.

- Reporting Mappings—Clicking the reporting mapping artifact name launches the **Report Mapping Designer** in a new tab. Edit reporting mappings as described in *Defining Data Maps*.

## Managing Approvals

Manage the approvals process by creating approval units, setting the promotional path, tracking budgets, and reviewing status, process issues, and ownership.

### Related Topics

- **About the Approvals Process**
- **Defining the Approvals Process**
- **Starting and Supporting the Review Process**
- **Printing Approval Unit Annotations**
- **Creating Approval Status Reports**
  You can view detailed status of the approval process using approval status reports. All users can access this report, however you’ll see only the approval units that you have write access to.
- **Approval Unit Promotional Path**

### About the Approvals Process

You can track budgets and review status, process issues, and approval unit ownership using approval units. Budget cycle time is reduced:

- Approval path is independent of organizational structure
- Exceptions and problem areas are highlighted
- Audit information includes annotations and process status
- Reviews include annotations and comments

Watch this overview video to learn how to set up approvals.

[Overview Video](#)

### Related Links

- **About Approval Units**
- **Approvals Process**
- **Data Validation Rules**
About Approval Units

An approval unit is the basic unit for preparing, annotating, reviewing, and approving plan data. Approval units are combinations of scenario, version, and entity or part of an entity. You can have one scenario/version combination per approval unit. Scenarios and versions are the basis of the review cycle. Approval units submit planning data for a scenario and version. For example, an approval unit might consist of a version (Best Case), a scenario (Actual), and an entity (New York). Approval units can also include secondary dimensions within any entity, refining the granularity of an approval unit.

Approvals Process

The approval unit moves from one reviewer to another until the budget process is complete. The review process follows the promotional path you set up when you select the owner and reviewers for an approval unit, unless an event triggers a change in the promotional path. Events that affect the promotional path include:

- Exceeding or not reaching expense boundaries for budget items such as salaries, new hires or capital equipment
- The current owner returning the budget to the previous owner for additional information
- The current owner requesting help from an authorized user who isn't necessarily on the promotional path

The selected approvals template determines the first user to review the budget (see Setting Approval Unit Hierarchy Name, Scope, and Template). The first user completes the assigned tasks, then promotes (Bottom Up template) or submits (Distribute template) the budget, which alerts the next owner that the budget requires their attention. Other users may also be notified whenever the budget passes from one user to another.

Each reviewer must validate the approval unit before sending the budget to the next reviewer. The validation runs all data validation rules defined for the approval unit with which the reviewer is working, and reports any data errors or changes in the approval unit promotional path. See Modifying the Approval Unit Promotional Path.

Tip:

To display users’ full names (instead of their user IDs) in Approvals, select the option Display Users’ Full Names, as described in What Application and System Settings Can I Specify?.

Data Validation Rules

To implement business policies and practices, administrators can build data validation rules that are checked when conditions are met in forms. Rules can generate validation messages, enforce limits on submitted approval unit data, and designate a specific reviewer or owner to review data that meets some condition.
For example:

• Conditionalizing the approval unit promotional path
• Preventing the promotion of approval units that include invalid data

When you design rules that affect the approval unit promotional path, you need to understand the order in which these rules are evaluated and applied. For information about designing data validation rules and expected outcomes, see Managing Data Validation.

Task Lists

Task lists guide users through the planning process by listing tasks, instructions, and end dates. Administrators and interactive users create and manage tasks and task lists. See Administering Task Lists.

Setting Up Email for Approvals Notification

Owners and reviewers must enable approvals notifications if they want to get an email notice when there's a change in approvals status or approvals ownership. Approvals notifications are enabled on the Application Settings page.

Approvals notification emails contain the following information:

• URL to Check Status
• Application Name
• Approval unit
• Current Owner
• Previous Owner
• Last Action
• Current Status
• Time
• Annotation Title
• Annotation Text

To enable approvals notifications:

1. From the home page, click Application, and then click Settings.
2. Under Notifications, select Approvals.
3. Enter your email address.

See also, Managing Application and System Settings.

Defining the Approvals Process

Related Topics

• Budget Process
• Approval Operations and Data Validations
Budget Process

The application supports bottom-up, distributed, or free-form budgeting. Typically, high-level users start the approval units containing loaded data, and then delegate data entry into the lowest-level members to their direct reports, who distribute to their direct reports, and so on. Until a budget is distributed, users can’t access it.

To define the budget process, administrators define:

- Approval unit hierarchies
- Owners and reviewers of the approval unit hierarchies
- Validation rules for evaluating submitted data

When a user submits budget data, the data is subjected to validation rules. If the data passes the validations, the budget is promoted to the next owner, and the original user can’t edit the data unless ownership is granted again. The submission process locks the data from being edited by anyone except the current owner.

Approval Operations and Data Validations

During approvals operations that invoke data validations, user variables and context user variables on the forms that are replaced with the distinct union of the result sets of all possible inputs. User variables and context user variables on the POV will be treated as a page.

Note:

To view the results of the user variable substitution, navigate to the data validation report task list. You may need to purposely cause a validation to fail on the form and then run a validation operation for the approval unit for it to display on the report.

Starting and Supporting the Review Process

After administrators start the review process, the approval unit moves from one reviewer to another until the budget process is complete. The selected approvals template determines the first user to review the budget.

You can view and provide detailed status of the approval process using approval status reports. See Creating Approval Status Reports.

To start the review process:

1. Click the Navigator icon and then under Workflow, click Manage Approvals.
2. In Scenario, select a scenario.
4. Click Go to display the approval units associated with the selected scenario and version combination.
The approval units displayed are enabled for approvals. If no approval units are started with the selected scenario and version, this message is displayed: You have not assigned Approval Unit Hierarchy to the selected Scenario and Version combination.

5. In the upper-right corner of the page for View, select Tree View, and then in Plan Cycle, click Start to begin the review process.

Note:
If the approval unit hierarchy uses the Bottom Up template, selecting Start starts the approval unit and runs the Originate action. These actions set the user defined as the approval unit owner in the approval unit hierarchy as the current owner, and the approval unit status changes to Under Review.

6. Optional: Select Exclude to remove an approval unit from the planning process or from being tracked in the system.

Caution:
After you exclude an approval unit, all associated annotations and history are discarded. Approval unit status is returned to Not Started and the owner is set to No Owner. Data values are retained.

Printing Approval Unit Annotations
Administrators can check approval unit status by reporting on annotations for a set of scenarios, versions, and approval unit members. Reports can be based on process status. The application name, selected scenarios, versions, and entities, approval unit title, status, author, origin date, and annotations are displayed. Annotation text displays chronologically, with the most recent entry first.

To create and print reports for approval unit annotations:

1. Click the Navigator icon and then under Monitor and Explore, click System Reports.
2. Select Approval Unit, and then click Annotations.
3. Under Select Report Options, select the Scenario, Version, and Entity combination for which you want to generate a report. If Custom is selected, click to select the custom scenarios, versions, and entities.
4. Under Approvals Status, select status states.
5. Click Create Report.
6. Click Print on the Adobe Acrobat toolbar.
Creating Approval Status Reports

You can view detailed status of the approval process using approval status reports. All users can access this report, however you'll see only the approval units that you have write access to.

Filters allow you to generate more focused reports. You can filter by:

- Approval Status
- Scenario
- Version
- Approval Unit Hierarchy

Approvals status reports provide the following information:

- Approval Unit
- Parent
- Status and Sub-Status
- Previous, Current, and Next Owner
- Current Location
- Total Value
- Last Status Change Date

Sample Approval Status Report (PDF Format)

You can generate approval status reports in the following formats:
• XSLX (Excel)
• PDF
• HTML
• XML

To create approval status reports:

1. Click the Navigator icon ➔, and then under Monitor and Explore, click System Reports.
2. Click Approval Unit, and then click Approval Status.
3. Make your selections:
   • Under Select Approval Status, select which states you'd like to view in the report or select All.
   • Under Filters, select the Scenario and Version. Click ➔ to open the Member Selection dialog.

   **Note:**
   Filtering on Approval Unit Hierarchy is optional. Use this option if you want to filter on a subset of an approval unit hierarchy.

   • Under Select Format, choose a file format (XSLX, PDF, HTML, or XML).
4. Click Create Report and select where to save the report.

Approval Unit Promotional Path

An approval unit moves from person to person and department to department based on the owners and reviewers you assign to each approval unit and its parents in the approval unit hierarchy.

There are two ways to affect the promotional path of an approval unit hierarchy:

• Modify owners and reviewers for approval units and their descendants using approvals actions (for example, Promote or Reject)
• Use data validation rules to determine whether a change in the approval unit promotional path is necessary
• Modifying the Approval Unit Promotional Path
• Design Considerations for Approval Unit Promotional Path Data Validation Rules

Modifying the Approval Unit Promotional Path

When you select an owner and reviewers for an approval unit and its parents, you set the approval unit promotional path (see Approval Unit Promotional Path). Sometimes, however, budget calculation results change the person who reviews the budget next. For example, if salaries for a sales group are more than 10% over the previous budget year, approval from someone other than the next reviewer may be required. To automate redirecting the budget in these cases, add conditions and actions to data valida-
tion rules that test for these exceptions, and then modify the promotional path when necessary. You can also use data validation rules to stop an approval unit from passing to the next reviewer when data errors exist in the approval unit.

To modify the approval unit promotional path:

1. Create or select a data validation rule (see Managing Data Validation).

2. In the Data Validation Rule Builder, click to create promotional path conditions and specify the action taken when a budget calculation doesn't meet these conditions.

3. Select one action:
   • Update Promotional Path to create a promotional path condition that adds reviewers or owners to the promotional path if the condition you set is triggered.
   • Do Not Promote to prevent promoting an approval unit to the next reviewer. This action usually indicates invalid data in the budget.

4. In the Process Cell dialog box, beneath Approvals, click Add to add a promotional path condition.

Promotional path conditions are processed in the order they are listed. Use the and to move them up or down within the list.

5. Optional: To duplicate a promotional path condition:
   a. Select a promotional path condition, and then click Copy.
   b. Select the promotional path condition that will precede the copied one in the list, and then click Paste to add the copied promotional path condition.

6. Optional: Click Delete to remove a promotional path condition.

7. In Approval Unit Hierarchy and Approval Units, click to select the approval unit hierarchy and approval units the promotional path condition affects.

8. In Promotional Path Condition, specify the promotional path changes for the approval units selected in Promotional Path Condition. Select:
   a. In Position, specify where the promotional path changes for the approval units selected in Approval Units. Select:
      • Before to insert the alternate owner or reviewer before ownership reaches the approval units specified in Promotional Path Condition.
      • After to insert the alternate owner or reviewer after ownership reaches the approval units specified in Promotional Path Condition.
      • Before and After to insert the alternate owner or reviewer before and after ownership reaches the approval units specified in Promotional Path Condition.
   b. In the Approval Units column of Promotional Path Condition, click to select the approval units the action selected in Position affects.
9. In the Assign column, specify the role, user type, and the user or UDA for each alternate owner, reviewer, and user to be notified if the rule exception occurs.
   
a. In Role, select:
   - Owner to select the alternate owner
   - Reviewer to select the alternate reviewer
   - Notified Users to select the user to be notified

   b. In Type, select:
   - User Name to specify the alternate user
   - From UDA to select the dimension and to enter the prefix that was assigned to the UDA

c. In Users, specify:
   - For User Name, the name of the alternate user or choose one from the user selector
   - For From UDA, the dimension from the Select a Dimension drop down, and then specify the prefix that was used in the UDA to indicate that the UDA contains a user name (for example, ProdMgr:)

   **Note:**
   
The selected approval units must be ancestors of the approval units selected in the previous step. If you select approval units that are not ancestors, the promotional path condition isn't evaluated when the approval unit hierarchy is validated.

10. Optional: Provide messages for the users involved in the approval unit promotional path changes:

   a. Click in:
   - Sender Message to enter the email message text sent to the user promoting the approval unit when the promotional path changes.
   - In Reviewer Message to enter the email message text sent to the user receiving the approval unit for review due to the rule exception.

   b. Click OK to save the messages.
   
The text you enter is added to the messages sent when the approval unit changes status.

11. Click OK to save the promotional path conditions, and return to data validation rule creation.
Design Considerations for Approval Unit Promotional Path Data Validation Rules

When you design rules that affect the approval unit promotional path, you need to understand the order in which these rules are evaluated and applied. For information about designing data validation rules and expected outcomes, see Managing Data Validation.

Managing Approval Unit Hierarchies

Related Topics

- About Approval Unit Hierarchies
- Working with Approval Unit Hierarchies

About Approval Unit Hierarchies

An approval unit hierarchy contains approval units and entities that are part of the review process.

Parent/child relationships between approval unit hierarchy members affect the review process:

- When you promote or reject a parent, its children are promoted or rejected unless they are Approved. The owner for the parent becomes the owner of the children.
- When you approve a parent, its children are approved.
- After all children are promoted to the same owner, the parent is promoted to the owner.
- When the status of all children changes to one status, for example Signed Off, parent status changes to the same status.

You can't change the status of a parent if its children have different owners. If the children are promoted to, submitted to, or signed off by different users, the parent has no owner and only administrators can change its status.

Working with Approval Unit Hierarchies

Use approval unit hierarchies to adapt the budgeting process to all types of organizational requirements.

- Approval units are a combination of the entity and other dimensions. For example, if an application includes all of a company's products, the approval unit hierarchy for North America can include dimensions and members appropriate to products sold in North America. Similarly, the approval unit hierarchy for the European division can include dimensions and members for products sold in Europe. Within the same approvals hierarchy, Latin America entities can be enhanced using the Account dimension, creating approval units such as Entities by HR, Entities by Capital Expenditures, and Entities by Revenue.
- Use preset budgeting mode templates to create hierarchies that are bottom-up, distributed, or free form.
• Include dynamic links to dimensions based on generation numbers for the entity dimension and the secondary dimension that is used. For example, automatically add generations 0 to 3 in the entity or segment dimension to the approval unit hierarchy. If a change occurs in the dimension, the approval unit hierarchy can be easily updated.

• Import and export approval unit hierarchies.

• Create approval unit hierarchies that differ by scenario and version. For example, the Budget scenario can have a large approval unit hierarchy consisting of departments, accounts, and products, while the Forecast has a simpler process organization with fewer levels of approval.

Related Links

• Creating Approval Unit Hierarchies
• Selecting Approval Unit Hierarchy Scenario and Version
• Editing Approval Unit Hierarchies
• Deleting and Renaming Approval Unit Hierarchies
• Viewing Approval Unit Hierarchy Usage
• Synchronizing Approval Unit Hierarchies
• Exporting Approval Unit Hierarchies
• Importing Approval Unit Hierarchies

Creating Approval Unit Hierarchies

Entity is the primary dimension for each approval unit hierarchy. As members are added to the Entity dimension, the inclusion rules you create determine whether a new member is part of the approval process. An administrator can also add members to the hierarchy as exceptions to the approval unit hierarchy structure.

To create an approval unit hierarchy:

1. Click the Navigator icon, and then under Workflow, click Approval Unit.
2. Click .

Note:

The approvals dimension is set to Entity. There are no other choices.

3. Create the generic rule that defines which Entity members are included in the approval process (see Setting Approval Unit Hierarchy Name, Scope, and Template).
4. Select the primary and subhierarchy members to include in the approval process (see Selecting Approval Unit Hierarchy Members).
5. Assign owners and reviewers for each stage of the approval process and create the approval unit promotional path (see Assigning Approval Unit Owners and Reviewers).
6. Click Save.
Setting Approval Unit Hierarchy Name, Scope, and Template

To set up an approval unit hierarchy:

1. Click the Navigator icon, and then under Workflow, click Approval Unit.

2. Take an action:
   - To create a new hierarchy, click .
   - To edit an existing hierarchy, select an approval unit hierarchy, and then click .

3. Complete the Approvals Dimension details:
   - For Hierarchy Name, provide the approval unit hierarchy name.
   - Optional: For Description, provide a description.
   - In Enable Approvals, select:
     - All to add all approval units to the approval process.
     - None to include no approval units in the approval process by default.
       To add approval unit groups or individual approval units to the approval process, see Selecting Approval Unit Hierarchy Members.
     - Custom to define which approval units to include in the approval process.
       You can add individual approval units and approval units based on Parent member and generation criteria. See Selecting Approval Unit Hierarchy Members.
   - For Approvals Template, select:
     - Bottom Up (see Bottom-Up Budgeting)
     - Distribute (see Distributed Budgeting)
     - Free Form (see Free-Form Budgeting)
   - For Cube, select the cube from which aggregated approval unit values are derived.
     See Assigning Approval Unit Owners and Reviewers.

4. Take an action:
   - Click Next or select Primary and Subhierarchy Selection to select the approval unit hierarchy members (see Selecting Approval Unit Hierarchy Members).
   - Click Save and then OK to save changes and close the approval unit hierarchy.

Bottom-Up Budgeting

Data is input at the leaf member level (for example, children of Budget Group) and consolidated by rolling data up the organizational hierarchy. When the budget is started, data is populated for each scenario and user independently. The ownership follows the hierarchy of approval in bottom-up mode. Users can view or edit data based on ac-
cess permissions defined for the approval unit. The topmost Budget Group owner consolidates individually approved budgets into a final consolidated budget.

Distributed Budgeting

Budget data is entered at the leaf level of the organization, and ownership starts at the top level of the organization. Ownership is then distributed down the organization hierarchy. After ownership reaches the lower levels, budgets are submitted back to the top through the approval process. The top budget group owner reviews, approves, and loads the budgets for budgetary control, transaction control, and reporting.

Free-Form Budgeting

With free-form budgeting, data is input at the leaf member, and planners select the next owner from a drop-down list. The free-form budgeting mode allows planners to select the next owner from a drop-down list. Select this budget template if you're not using the approvals features described in Creating Approval Unit Hierarchies.

Selecting Approval Unit Hierarchy Members

To select approval unit hierarchy members:

1. Take an action:
   - From Approvals Dimension, click Next or select Primary and Subhierarchy Selection to continue defining an approval unit hierarchy.
   - Click the Navigator icon, and then under Workflow, click Approval Unit. Select an approval unit hierarchy, then click , and then select Primary and Subhierarchy Selection to edit members.

   **Note:**
   - At any time during approval unit hierarchy member selection, you can click Reset to Default Hierarchy to reset the approval unit hierarchy to its default membership defined in the Approvals Dimension page.
   - Shared members are not displayed in the approval unit hierarchy.

2. Define how to display the approval unit hierarchy:
   - **All Entities**—Displays all potential approval units
   - **Approval Units**—Displays only enabled approval units
   - **Search**—Searches approval units by Name, Alias, or Both.

   Enter any part or all of a name in Search to locate an entity, then click or in the approval unit hierarchy.
   - To move from page to page in a multipage approval unit hierarchy, enter a page number in Page and click Go, or click Start (first page), Prev (previous page), Next, or End (last page).
3. **Optional:** For approval units not included in the default settings for the approval process, check the box to the left of the approval unit name to include the approval unit in the approval process.

4. **Optional:** Right-click an approval unit name to define subhierarchy members for the approval process, and then select one option:
   - **Include Children** to include the children of the approval unit.
   - **Include Member** to include just the approval unit, but none of its descendants.
   - **Include All Descendants** to include all descendants of the approval unit.
   - **Include Generation** to include one or more approval unit generations. Specify the generations to include when prompted.
   - **Exclude Children** to exclude the children of the approval unit.
   - **Exclude Member** to exclude just the approval unit, but none of its descendants.
   - **Exclude All Descendants** to exclude all descendants of the approval unit.
   - **Exclude Generation** to exclude approval unit generations. Specify the generations to exclude when prompted.

5. **Optional:** Add a secondary dimension to an approval unit included in the approval process to provide finer granularity:
   a. Select a dimension from **Dimension**.
   b. In **Parent Member**, click ![icon] to display the member selection window, and then select one member as the parent member for that dimension.
   c. In **Relative Generation**, specify the parent-member generations to include. Adding a generation adds all members of that generation to the approval process.
   d. Check **Auto Include** to automatically include newly-added members to the approval unit hierarchy that meet the selected criteria.
   e. **Optional:** Click ![icon] under **Selected Members** for the approval unit to refine the members it includes. Clear the check box next to any member you want to remove.

6. **Take an action:**
   - Click **Next** or select **Assign Owners** to specify approval unit ownership (see Assigning Approval Unit Owners and Reviewers).
   - Click **Save** and then **OK** to save changes and close the approval unit hierarchy.

**Assigning Approval Unit Owners and Reviewers**

Approval unit ownership is inherited from the approval unit parents. Approval unit reviewers are also inherited. You can also explicitly specify approval unit owners and reviewers to assign owners and reviewers other than those approval units inherit.

To assign approval unit reviewers and owners:

1. **Take an action:**
• From **Primary and Subhierarchy Selection**, click Next or select **Assign Owners** to continue defining an approval unit hierarchy.

• Click the **Navigator** icon, and then under **Workflow**, click **Approval Unit**. Select an approval unit hierarchy, click , and then select **Assign Owners** to specify approval unit ownership.

2. Select an approval unit, then under **Owner**, click to select an owner.

   An approval unit can have only one owner. Either a user or a group can be the owner. Select the **Users** tab to assign an individual user as the owner. Select the **Groups** tab to assign a group as the owner. See **About Group-based Approvals**.

3. Under **Reviewer**, click and select approval unit reviewers.

   Reviewers can be individual users, a single group, or multiple groups. Select the **Users** tab to assign individual users as reviewers. Select the **Groups** tab to assign a single group or multiple groups as the reviewer. See **About Group-based Approvals**.

   **Note:**

   When using the Bottom Up or Distribute template, if the reviewers are individual users, select the reviewers in the order you want them to review the approval unit. The first reviewer in the list is the first user to work on the approval unit. When the first reviewer promotes or submits the approval unit, the second reviewer selected becomes the approval unit owner, and so on through the list of reviewers you create.

4. To enable viewing the aggregated value of approval units during approvals, under **Value Definition Members**, click to define the intersection of non-approval unit dimensions.
**Note:**

- For non-approval unit dimensions: To view aggregated approval unit values, on **Member Selection**, you must select one member for each non-approval unit dimension in the cube selected on the **Approvals Dimension** tab. If you don't select members for the non-approval unit dimensions, the aggregated approval unit value is blank.

- For approval unit dimensions: Value definition members specified for a parent member in the approval unit hierarchy are inherited by the descendants of that member. If value definition members are selected for both a parent and child member of an approval unit dimension, then the child's selection takes precedence. For example, you select a value definition member for both the parent US and its child, Western Region; the value definition member for Western Region takes precedence. At runtime, the Entity, Scenario, Version, and secondary dimension members specified in the approval unit hierarchy are selected.

- In Standard multicurrency applications: In **Preferences**, users set the **Reporting Currency** in which to display the aggregated approval unit. See *Working with Planning*. To ensure that totals are correct, remind users to launch the Currency Conversion business rule.

  Note that in Standard multicurrency applications, the reporting currency is assumed, so you don't select a value definition member for currency.

5. Under **Promotional Path**, click to display the approval unit promotional path, verify that it's correct, and then correct any errors.

6. Under **Notify These Users**, click to select the users to notify for each move of the approval unit from one user to another.

7. **Optional:** Repeat Steps 2 through 6 for other approval units to change their inherited owners or reviewers.

8. Click **Save** to save your work and continue, or click **OK** to save your work and close the approval unit hierarchy.

**About Group-based Approvals**

When assigning approval unit owners, you can assign individual users, or you can assign a group. When assigning approval unit reviewers, you can assign individual users, a group, or multiple groups.

See the following topics for more information:

- Assigning a Group as the Owner
- Assigning a Group or Groups as the Reviewer

**Assigning a Group as the Owner**

Only one user or one group can be assigned as an owner of an approval unit. Within a group, any user can become the owner, but only one user can be the owner at a time.
Only the user assigned as an owner can perform actions. Other group members can take ownership away from the current owner. If no one is assigned as the owner, then anyone in the group can perform actions on behalf of the group without having to first claim ownership.

Assigning a Group or Groups as the Reviewer

If you select individual users as reviewers, all users must approve, and the approvals must follow the order in which the users are entered. If you select a group (or groups) as a reviewer, any user within the group can be the reviewer and can promote to the next level.

The following are some examples of how you can enter multiple reviewers:

Example 18-3  Example 1

North America - Bill  
USA - Nick, Sandy, Kim  
CA - John

The approval path is John, Nick, Sandy, Kim, Bill.

Example 18-4  Example 2

North America - Bill  
USA - (Group A)  
CA - John

The approval path is John, any user in Group A, Bill.

Example 18-5  Example 3

North America - (Group B), Bill  
USA - Susan, (Group A)  
CA - John

The approval path is John, Susan, any user in Group A, any user in Group B, Bill.

Example 18-6  Example 4

North America - (Group B), Bill  
USA - Susan, (Group A)  
CA - John

**A validation rule indicates that if New Hires > 4, then before USA, set {Group HR} as reviewer.

If the condition is true for the CA approval unit, then the path is: John, any user in Group HR, Susan, any user in Group A, any user in Group B, Bill.

Selecting Approval Unit Hierarchy Scenario and Version

To select the approval unit hierarchy scenario and version:
1. Click the Navigator icon, and then under Workflow, click Manage Approvals.

2. In Scenario, select a scenario.


4. Click Go to display the approval unit hierarchy defined for the selected scenario and version.

5. In View, click one of the view options.

6. Optional: If you own approval units for this hierarchy, click Display My Approval Units to select them.

7. For each approval unit, the following is displayed:
   - Plan Cycle (Tree View only), which displays whether the approval unit is started and if it's included in the budget process
   - Approvals Status, for example, 1st Pass
   - Sub-Status
   - Current Owner
   - Location
   - In Path, click to view the potential promotional path.
   - Click Details, to display approval unit details and add or edit approval unit annotations.

Editing Approval Unit Hierarchies

To edit an approval unit hierarchy:

1. Click the Navigator icon, and then under Workflow, click Approval Unit.

2. Select the approval unit hierarchy with which you want to work.

3. Click .

4. Select the appropriate tab, depending on the changes you want to make (for example, select Assign Owners to edit ownership of the approval units in the hierarchy):
   - Select the Approvals Dimension tab to modify the approval unit hierarchy description or approvals default scope. See Setting Approval Unit Hierarchy Name, Scope, and Template.
   - Select the Primary and Subhierarchy Selection tab to select or remove members from approvals. See Selecting Approval Unit Hierarchy Members.
   - Select the Assign Owners tab to modify approval unit owners or reviewers. See Assigning Approval Unit Owners and Reviewers.
   - Select the Usage tab to view the application objects (forms or scenario and version combinations) that reference the approval unit hierarchy. See Viewing Approval Unit Hierarchy Usage.

5. Click Save when done.
To select an approval unit hierarchy scenario and version with which to work, see Selecting Approval Unit Hierarchy Scenario and Version.

Deleting and Renaming Approval Unit Hierarchies

You can delete an approval unit hierarchy if it's not referenced by other application objects such as forms or scenario and version combinations. The Usage tab displays objects that reference the selected approval unit hierarchy and links you to the objects so you can disassociate them from the hierarchy.

You can also change the name of an approval unit hierarchy. Renaming an approval unit hierarchy doesn't affect application objects that reference the hierarchy.

To delete or rename approval unit hierarchies:

1. Click the Navigator icon, and then under Workflow, click Approval Unit.
2. Take an action:
   - To delete, select the approval unit hierarchy to delete, and then click .
   - To rename, select the approval unit hierarchy to rename, and then click . Enter the new name for the approval unit hierarchy in the dialog box.
3. Click OK.

If you select an approval unit hierarchy that is referenced by other application objects, an error message is displayed. See Viewing Approval Unit Hierarchy Usage to view and edit the application objects that reference the approval unit hierarchy.

Viewing Approval Unit Hierarchy Usage

Approval unit hierarchies might have dependencies such as approval unit assignments or data validation rules which are defined in forms. If dependencies exist for an approval unit hierarchy, the hierarchy can't be deleted until the dependencies are removed. The Usage tab enables you to view approval unit hierarchy dependencies and links you to the forms or scenario and version combinations so you can remove the dependencies, if needed.

To view approval unit hierarchy usage:

1. Click the Navigator icon, and then under Workflow, click Approval Unit.
2. Select an approval unit hierarchy, then click , and then select the Usage tab to view approval unit hierarchy dependencies.
3. Select Forms to view associated data validation rules:
   - If no data validation rules are associated, a message is displayed.
   - If data validation rules are associated, they are listed by form. Click the rule hyperlink to display the form in edit mode in a new tab. You can then update or delete the data validation rule to disassociate it from the approval unit hierarchy. See Creating and Updating Data Validation Rules.
4. Select Approval Unit Assignment to view associated scenario and version combinations:
• If no scenario and version combinations are associated, a message is displayed.

• If scenario and version combinations are associated, they are listed by scenario. Click the version hyperlink to display the combination in a new tab. You can then remove the approval unit assignment to disassociate it from the approval unit hierarchy. See Assigning Approval Unit Hierarchy Scenario and Version Combinations.

5. If dependencies are removed during steps 3 or 4, click Refresh on the Usage tab to view the remaining dependencies.

6. If you’re deleting an approval unit hierarchy, repeat steps 3, 4, and 5 until all dependencies are removed.

Once all dependencies are removed from an approval unit hierarchy, you can delete the hierarchy. See Deleting and Renaming Approval Unit Hierarchies.

Synchronizing Approval Unit Hierarchies

When you add, delete, or modify dimension members that are used in approval unit hierarchies, the affected approval unit hierarchy must be synchronized with the changes. When you display the list of approval unit hierarchies, the entry for each approval unit hierarchy specifies whether recent changes are reflected in the approval unit hierarchy. Use this procedure to synchronize dimension member changes with the approval unit hierarchy.

Note:

When you add dimension members, they are added as approval units only if they meet the criteria in the inclusion rules for the approval unit hierarchy. For example, if the added entity is a fourth-generation entity, and the inclusion rules specify generations one through three as approval units, the entity isn't added as an approval unit. If the entity is a third-generation member, however, it's added as an approval unit the next time the approval unit hierarchy is edited and saved, or synchronized.

To synchronize changes to approval unit hierarchies:

1. Click the Navigator icon, and then under Workflow, click Approval Unit.
2. In the Synchronized column, approval unit hierarchies are labeled as follows:
   • Synchronized—Changes are synchronized with the approval unit hierarchy
   • Not Synchronized—Changes are not synchronized with the approval unit hierarchy
   • Locked By user—A user is editing or synchronizing the approval unit hierarchy
Note:

If a user begins editing or synchronizing an approval unit hierarchy after you display the approval unit hierarchy list, the approval unit hierarchy list doesn't display "Locked" for the approval unit hierarchy. If you try to synchronize this approval unit hierarchy, the synchronization doesn't occur, and an error message states that it's being edited.

3. Select an approval unit hierarchy listed as Not Synchronized, and then click Synchronize.

Changes are applied to the approval unit hierarchy, and the list of approval units is updated according to the inclusion rules defined for the approval unit hierarchy.

Note:

You can't synchronize changes to an approval unit hierarchy that another user is editing or synchronizing.

Exporting Approval Unit Hierarchies

When you export an approval unit hierarchy, you create a file that contains the approval unit hierarchy information. After you create this file, you can copy its contents to an existing approval unit hierarchy (see Importing Approval Unit Hierarchies).

To export approval unit hierarchies:

1. Click the Navigator icon, and then under Workflow, click Import and Export.
2. Select Export Approval Unit Hierarchy.
3. In Existing Approval Unit Hierarchy Name, select an approval unit hierarchy to export.
4. Click OK.
5. When the Save dialog box is displayed, save the export file to a location of your choice.
6. Click Export or Done. Export performs the action, and Done closes the dialog box.

Importing Approval Unit Hierarchies

An approval unit hierarchy can be populated with the contents of an approval unit hierarchy import file. The import file is the result of exporting an existing approval unit hierarchy. See Exporting Approval Unit Hierarchies.

Importing the approval unit hierarchy information doesn't create an approval unit hierarchy. The approval unit hierarchy populated from the export file must exist and have at least a name before the import. Loading approval unit hierarchies first deletes all members of the approval unit hierarchy, and then adds each member specified in the input file as a new member.
To import an approval unit hierarchy:

1. Click the **Navigator** icon, and then under **Workflow**, click **Import and Export**.
2. Select **Import Approval Unit Hierarchy**.
3. In **Existing Approval Unit Hierarchy Name**, select the approval unit hierarchy receiving the exported information.

**Note:**

The imported approval unit hierarchy includes the defined owner, reviewers, and rules for determining the promotional path.

4. For **Approval Unit Hierarchy with Ownership**, click **Browse** to select the exported approval unit hierarchy file to import.
5. Click **OK**.
6. Click **Import** or **Done**. **Import** performs the action, and **Done** closes the dialog box.

   If the message **Import successful** is displayed, the approval unit hierarchy information in the exported file was successfully copied to the approval unit hierarchy that you selected in **Existing Approval Unit Hierarchy Name**.

   If the message **Import not successful. Some items have not been imported** is displayed, click **Details** to view the log file. Correct the errors and retry importing the approval unit hierarchy.

### Assigning Approval Unit Hierarchy Scenario and Version Combinations

During the budget process, calculations are run for various scenarios, such as Q1 or FY10. For each scenario, calculations can be run for various versions, for example Initial or Final. Before beginning the budget process, assign approval unit hierarchies to the scenario and version combinations included in the budget process.

To assign approval unit hierarchy scenario and version combinations:

1. Click the **Navigator** icon, and then under **Workflow**, click **Approval Unit Assignment**.
2. Add an approval unit assignment.
   a. Click **Edit**, in the **Actions** column, for the approval unit.
b. Click **Select** in the **Scenario** column, and then select the scenario to associate with the approval unit hierarchy.

c. Click **Select** in the **Version** column, and then select one or more versions to associate with the selected scenario.

d. Click **OK**.

A new assignment row is displayed.

3. **Optional**: Click the **X** to remove an approval unit assignment.

4. Click **Save** to save the approval unit assignments and continue or click **Cancel** to undo any changes made since the last save.
A

Naming Restrictions

Related Topics
• Restrictions for Applications and Databases
• Restrictions for Dimensions, Members, Aliases, and Forms
• Dimension and Member Names in Calculation Scripts, Report Scripts, Formulas, Filters, and Substitution Variables
• Restrictions for User and Group Names

Restrictions for Applications and Databases

When naming an application, follow these rules:
• Use no more than eight characters for an application’s name.
• Don’t use spaces.
• Don’t use these special characters:
  – asterisks
  – backslashes
  – brackets
  – colons
  – commas
  – equal signs
  – greater than signs
  – less than signs
  – periods
  – plus signs
  – question marks
  – quotation marks (double and single)
  – semicolons
  – slashes
  – tabs
  – vertical bars
• For application names in relational database environments, don’t use extended characters (except for underscores).
• For aggregate storage databases, don’t use DEFAULT, LOG, METADATA, or TEMP as application names.

Enter names in the preferred case. Application names are created exactly as entered.
Restrictions for Dimensions, Members, Aliases, and Forms

When naming dimensions, members, and aliases, follow these rules:

- For dimensions, members, or aliases, use no more than 80 characters.
- Don't use dimension names that are the same name as a member property or a column header in the metadata load file (for example, Data Type).
- Don't use HTML tags in member names, dimension names, aliases, and descriptions.
- Don't use quotation marks ("), brackets ([ ]), backslashes (\), slashes (/), or tabs. Brackets are permitted but not recommended in block storage outlines. They cause errors when converting to aggregate storage outlines.
- Don't use these special characters to begin dimension or member names:
  - at sign (@)
  - backslash (\)
  - bracket ([ ])
  - comma (,)
  - dash, hyphen, or minus sign (-)
  - equal sign (=)
  - less than sign (<)
  - parentheses ()
  - period (.)
  - plus sign (+)
  - quotation mark ("")
  - slash (/)
  - underscore (_)
  - vertical bar (|)
- Also, member names with the above special characters are not allowed in Planning data export jobs.
- Don't place spaces at the beginning or end of names.
- For time periods in custom calendars, don't use spaces in prefixes.
- Don't use these words as dimension or member names:
  - Calculation script commands, operators, and keywords.
  - Function names and function arguments.
  - Names of other dimensions and members (unless the member is shared).
  - If Dynamic Time Series is enabled, don't use History, Year, Season, Period, Quarter, Month, Week, or Day.
  - In unique member outlines that contain an attribute dimension, don't use Sum, Count, Min, Max, and Avg as member or dimension names. Using these names causes duplicate name conflicts.
Note:

The members that are created in the attribute calculations dimension (Sum, Count, Min, Max, and Avg) are not considered reserved words because you can change these names in the attribute calculations dimension and then use the standard name in an attribute or standard dimension.

If the outline is tagged as a unique member outline, avoid using Sum, Count, Min, Max, and Avg as member names. For example, if you use Max in a standard dimension and then create an attribute dimension, in which the Max member in the attribute calculations dimension are created, a duplicate name is detected and the following error message is returned:

"Analytic Server Error(1060115): Attribute Calculations dimension/member name already used."

If the outline is tagged as a duplicate member outline, and an attribute dimension (and, therefore, the attribute calculations dimension) exists before you use Sum, Count, Min, Max, and Avg as a base member, the duplicate name is allowed. However, if, you use Sum, Count, Min, Max, and Avg as a base member before creating an attribute dimension, the duplicate name isn't allowed.

These words:

* ALL
* AND
* ASSIGN
* AVERAGE
* CALC
* CALCMBR
* COPYFORWARD
* CROSSDIM
* CURMBRNAME
* DIM
* DIMNAME
* DIV
* DYNAMIC
* EMPTYPARM
* EQ
* EQOP
* EXCEPT
* EXP
* * * 
* EXPERROR 
* FLOAT 
* FUNCTION 
* GE 
* GEN 
* GENRANGE 
* GROUP 
* GT 
* ID 
* IDERROR 
* INTEGER 
* LE 
* LEVELRANGE 
* LOCAL 
* LOOPBLOCK 
* LOOPPARMS 
* LT 
* MBR 
* MBRNAME 
* MBRONLY 
* MINUS 
* MISSING 
* MUL 
* MULOP 
* NE 
* NON 
* NONINPUT 
* NOT 
* OR 
* PAREN 
* PARENPARM 
* PERCENT 
* PLUS 
* RELOP 
* SET 
* SKIPBOTH 
* SKIPMISSING
Dimension and Member Names in Calculation Scripts, Report Scripts, Formulas, Filters, and Substitution Variables

In substitution variable values, calculation scripts, report scripts, filter definitions, partition definitions, or formulas, you must enclose member names in brackets ([ ]) when used within MDX statements and in quotation marks (" ") for block storage databases, in these situations:

- The name starts with one or more numerals (for example, 100).
- The name contains spaces or these characters:
  - ampersand (&)
  - asterisk (*)
  - at sign (@)
  - backslash (\)
  - braces ({ })
  - colon (:)
  - comma (,)
  - dash, hyphen, or minus sign (-)
  - exclamation point (!)
  - equal sign (=)
  - greater than sign (>)
  - less than sign (<)
  - parentheses ()
In calculation scripts and formulas, enclose member names that are also Essbase keywords (""") for block storage databases, and in brackets ([ ]) for aggregate storage databases including these member names:

BEGIN DOUBLE ELSE END FUNCTION GLOBAL IF MACRO MEMBER RANGE RETURN STRING THEN

Enclose in quotation marks names, in calculation scripts, report scripts, or formulas, that contain or start with spaces or these characters: + - * / ( ) : , @ ; ) { } [ ] <

Also enclose in quotation marks names that start with numbers. For example:

@SUM(ac1, "12345");

Restrictions for User and Group Names

User and group names can have up to 80 characters.
Form Formula Functions

Related Topics
- About Form Formula Functions
- Working with Formulas
- Formula Functions

About Form Formula Functions

After you create a formula row or column, define the formula using the Segment Properties pane. Formulas include grid references, mathematical operators, and, optionally, mathematical functions. For a complete list, see Formula Functions.

Note:
The application requires level 0 members that are Dynamic Calc to have a member formula. For Dynamic Calc members that don't have a formula, the application inserts a semicolon (:) when refreshed.

Working with Formulas

Related Topics
- Creating Formulas
- Editing Formulas
- Deleting Formulas

Creating Formulas

To create formulas:
1. Select the row or column for which to associate the formula.
2. If it doesn't automatically expand, select Segment Properties.
3. In Formula, enter the name for the formula and then click validate.
4. In the Formula box that is displayed, select the operation or function that the formula will perform, such as COUNT(), MAX(), and IFThen(). See Formula Functions.
5. Click Validate to ensure that the formula doesn't contain any errors.
Editing Formulas

To edit formulas:

1. Open a form with a formula row or column.
2. In **Layout**, select a numbered formula row or column.
3. Use **Segment Properties** to modify the formula properties.
4. Click \( \mathbb{1} \) to change the operation or function performed.
5. Save the form.

Deleting Formulas

To delete a formula:

1. Select the formula row or column.
2. Click **Delete**.
3. To verify formula deletion:
   a. Click the checkmark in the formula bar.
   b. Click another cell within the grid to reset the formula bar.
   c. Click the cell from which you deleted the formula to verify deletion.

Formula Functions

This section defines the mathematical functions available for creating formulas for form formula rows and columns. To insert formula rows and columns in forms, see Adding Formula Rows and Columns.

The syntax for mathematical functions:

\[
\text{FunctionName}(\text{arguments})
\]

For more information about arguments, see Arguments.

<table>
<thead>
<tr>
<th>Table B-1  Mathematical Function Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variable</strong></td>
</tr>
<tr>
<td><strong>FunctionName</strong></td>
</tr>
<tr>
<td><strong>arguments</strong></td>
</tr>
</tbody>
</table>
# Mathematical Functions

## Table B-2  Mathematical Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abs</td>
<td>Returns the absolute value of numeric values or references. See Abs.</td>
</tr>
<tr>
<td>Average</td>
<td>Returns the average of a group of numeric values or references. See Average.</td>
</tr>
<tr>
<td>AverageA</td>
<td>Returns the average of a group of numeric values or references. The calculation includes #MISSING cells only for rows or columns that are not suppressed. See AverageA.</td>
</tr>
<tr>
<td>Count</td>
<td>Returns the number of values in a group of numeric values or references. See Count.</td>
</tr>
<tr>
<td>CountA</td>
<td>Returns the number of values in a group of numeric values or references. The calculation includes #MISSING cells only for rows or columns that are not suppressed. See CountA.</td>
</tr>
<tr>
<td>Difference</td>
<td>Returns the absolute value of a numeric value or reference subtracted from another numeric value or reference. See Difference.</td>
</tr>
<tr>
<td>Eval</td>
<td>Evaluates an expression. Eval is useful for embedding expressions as function arguments. See Eval.</td>
</tr>
<tr>
<td>IfThen, If</td>
<td>Returns one value if a condition equals true, and another value if a specified condition equals false. See IfThen, If.</td>
</tr>
<tr>
<td>Max</td>
<td>Returns the maximum value from a group of numeric values or references. See Max.</td>
</tr>
<tr>
<td>Min</td>
<td>Returns the minimum value from a group of numeric values or references. See Min.</td>
</tr>
<tr>
<td>Mod</td>
<td>Returns the remainder, modulus, from a division formula. See Mod.</td>
</tr>
<tr>
<td>PercentOfTotal</td>
<td>Returns the result of a numeric value or reference divided by another numeric value or reference, multiplied by 100. See PercentOfTotal.</td>
</tr>
<tr>
<td>Pi</td>
<td>Returns the number 3.14159265358979, to 15 digits. See Pi.</td>
</tr>
<tr>
<td>Product</td>
<td>Multiplies all numbers or references and returns the product. See Product.</td>
</tr>
<tr>
<td>Random</td>
<td>Returns a random number between 0.0 and 1.0. See Random.</td>
</tr>
<tr>
<td>Round</td>
<td>Rounds a number up or down by specified digits. See Round.</td>
</tr>
<tr>
<td>Sqrt</td>
<td>Returns the square root of a numeric value, row, column, or cell. See Sqrt.</td>
</tr>
<tr>
<td>Sum</td>
<td>Returns the sum of a group of numeric values or references. See Sum.</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Truncate / Trunc</td>
<td>Removes the specified number of digits from numeric values. See Truncate/Trunc.</td>
</tr>
<tr>
<td>Variance / Var</td>
<td>Evaluates the difference between the specified values based on the account type for the current account. See Variance/Var.</td>
</tr>
<tr>
<td>VariancePercent / VarPer</td>
<td>Evaluates the percentage difference between the specified values based on account type for the current account. See VariancePercent/VarPer.</td>
</tr>
</tbody>
</table>

### Arguments

Mathematical functions accept numeric values, row, column, or cell references, or embedded functions as arguments. There are four argument types:

- **Numeric**—See Numeric Arguments.
- **Property**—See Property Arguments.
- **Row, column, or cell reference**—See Row, Column, or Cell Reference Arguments.
- **Embedded functions**—See Embedded Functions as Arguments.

### Numeric Arguments

The syntax for a numeric argument is

\[(numeral1, numeral2,...numeraln)\]

where numerals 1 through n are any numbers including decimals and negative values. For example, the expression Average(10,20,30) returns the value 20.

### Row, Column, or Cell Reference Arguments

The row, column, or cell argument identifies a row, column, or cell in a grid. The syntax:

\[\text{FunctionName}(\text{GridName}.\text{GridElement}[\text{segment(range)}].\text{Property})\]

### Table B-3  Argument Components

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
</table>
| *GridName* | The form name. For example:
Difference (grid1.row[5], grid1.row[6]) returns the difference of two rows on form grid1. Optional. If GridName isn't specified, the default is the name of the current form. |
Table B-3  (Cont.) Argument Components

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GridElement</strong></td>
<td>One of the following keywords: row, col, column, or cell. For example, Max(row[1], row[2], row[3]) returns the maximum value of three rows. GridElement is optional. However, a cell reference requires row and column segment identifiers. For example, cell[2, A] and [2, A] both refer to the cell that is the intersection between row 2 and column A. The keyword cell is optional. Cell references can use the [row, col] syntax or [col, row] syntax. Optional. If GridElement isn't specified, letters represent columns and numbers represent rows; for example: Max ([1, A], [2, A], [3, A]) refers to rows 1, 2 and 3 of column A.</td>
</tr>
<tr>
<td><strong>segment</strong></td>
<td>A row, column, or cell reference number. For an expanded row or column, you must specify the segment. For example, row[2] addresses row segment 2. Segments are enclosed in square brackets [.]. Required.</td>
</tr>
<tr>
<td><strong>range</strong></td>
<td>The rows, columns, or cell that are expanded from the specified segment. If range is specified, the system calculates the formula using only the specified range. For example, row[2(3:5)] uses only the third through fifth rows of expanded segment 2. Optional. When range isn't provided, all expanded cells are used.</td>
</tr>
</tbody>
</table>

**Note:**

If a segment expands to only one row or column, don't use the range argument.
Table B-3 (Cont.) Argument Components

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
</table>
| property | One of these keywords: average, averageA, count, countA, max, min, product, or sum. The property specifies how to aggregate the specified expanded rows, columns, or cells. Oracle recommends that property is not specified when a reference is an argument. By not specifying the property, the function calculates the reference in the most appropriate way. For example, the following expression returns the average of the cells within rows 1 and 2: 
\[
\text{Average}(\text{row}[1], \text{row}[2]) 
\]
In contrast, the following example first calculates the average of row[1], then the average of row[2], adds these two results, then divides by 2:
\[
\text{Average}(\text{row}[1].\text{average}, \text{row}[2].\text{average}) 
\]
The default property for a row, column, or cell reference is sum. For example, \text{row}[2] is equivalent to \text{Sum}(\text{row}[2]). |

Because segment is the only required part of a reference, the following references are the same:

\[
\text{Grid1.row}[1].\text{sum} 
\]

\[1\]

AverageA and CountA include #MISSING cells in the calculation. For example, if row 1 is a segment row that expands to Qtr1 = 100, Qtr2 = 200, Qtr3 = #MISSING, and Qtr4 = 400, the following function returns the value four (4):

\[
\text{row}[1].\text{CountA} 
\]

All other functions exclude #MISSING data cells. For example, the previous example of row 1 that expands to Qtr1 = 100, Qtr2 = 200, Qtr3 = #MISSING, and Qtr4 = 400, returns three in this example:

\[
\text{row}[1].\text{Count} 
\]

Property Arguments

Property arguments consolidate expanded references to a single value that is then used in the calculation. Use property arguments to perform calculations on an aggregate row, column, or cell. The two types of property arguments are:

- Aggregate—See Aggregate Property Argument.
Aggregate Property Argument

An aggregate row, column, or cell contains multiple rows, columns, or cells, respectively.

The aggregate property argument is the last argument in the following mathematical function syntax:

\[
\text{FunctionName} (\text{GridName.Gridelement[segment(range)]}, \text{property})
\]

You apply the following aggregate properties to a row, column, or cell reference.

**Table B-4 Aggregate Properties**

<table>
<thead>
<tr>
<th>Properties</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td>Returns the average of a row, column, or cell. The calculation excludes #MISSING and #ERROR values.</td>
</tr>
<tr>
<td>AverageA</td>
<td>Returns the average of a row, column, or cell. The calculation includes #MISSING and #ERROR values.</td>
</tr>
<tr>
<td>Count</td>
<td>Returns the number of values in a row, column, or cell. The calculation excludes #MISSING and #ERROR values.</td>
</tr>
<tr>
<td>CountA</td>
<td>Returns the number of values in a row, column, or cell. The calculation treats #MISSING and #ERROR values as zero (0).</td>
</tr>
<tr>
<td>Max</td>
<td>Returns the maximum value of a row, column, or cell</td>
</tr>
<tr>
<td>Min</td>
<td>Returns the minimum value of a row, column, or cell</td>
</tr>
<tr>
<td>Product</td>
<td>Returns the product of rows or columns</td>
</tr>
<tr>
<td>Sum</td>
<td>Returns the sum of a row, column, or cell</td>
</tr>
</tbody>
</table>

When used as a mathematical function argument, the default for property is the same as the function. In the following example the default property is Average:

\[
\text{Average}(\text{row}[2])
\]

When not used as a mathematical function argument, the default for property is sum. In the following example the default property is the sum of an aggregate row:

\[
\text{row}[2]
\]

Reference Property Argument

A reference property argument specifies how to treat formula reference results and is used in conjunction with the other properties.
There is one reference property argument:

**IfNonNumber/IFFN**

IfNonNumber specifies a replacement of #MISSING and #ERROR values with a specific numeric value. The syntax:

```
AXIS[segment(range)].IfNonNumber(arg).AggregateProperty
```

### Table B-5  Reference Property Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AXIS</td>
<td>One of these keywords: row, column, or cell</td>
</tr>
<tr>
<td>Segment(range)</td>
<td>Indicates any valid axis reference, such as a row number, column letter</td>
</tr>
<tr>
<td>IfNonNumber</td>
<td>Indicates how to treat missing or error data within the Axis Ref</td>
</tr>
<tr>
<td>(arg)</td>
<td>Indicates what number to use if missing or error data is encountered within the AxisRef</td>
</tr>
<tr>
<td>AggregateProperty</td>
<td>The aggregate function is used for aggregate segments</td>
</tr>
</tbody>
</table>

**Example:**

If cell[1,A] = 3 and cell[1,B] = #MISSING,

The expression:

```
cell[1,A] / cell[1,B]
```

returns #ERROR.

The expression:

```
cell[1,A] / cell[1,B].ifNonNumber(1)
```

replaces cell[1,B] with 1 and returns a 3.

**Note:**

If you use suppression for #MISSING in a grid, and the grid contains a formula row or column that uses the IfNonNumber property, #MISSING remains suppressed.
Embedded Functions as Arguments

You can embed functions as arguments within a function.

**Example:**

In this example, the function Average is embedded in the function Sum:

\[
\text{sum} (\text{row}[3:5], \text{avg}(\text{row}[4:6], 40, 50), \text{row}[7; 9], 70, 80)
\]

- Row segments 3, 4 and 5
- The average of row segments 4, 5 and 6, with the numbers 40 and 50
- Row segments 7 and 9
- The numbers 70 and 80

**Abs**

Abs is a mathematical function that returns the absolute value of a numeric value, row, column, or cell. The absolute value of a number is that number without regard to sign. A negative number becomes positive, while a positive number doesn't change. The function syntax:

\[
\text{Abs} \ (\text{argument})
\]

where **argument** is one of the following:

**Table B-6  Arguments for the Abs Function**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, ( \text{Abs} (-20) ) returns the value 20. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: ( \text{GridName/GridElement[segment(range)].Property} ).</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Examples:**

- The following expression includes a numeric argument and returns the value 30:

\[
\text{Abs} \ (-30)
\]

- The following example returns the absolute value of the value in row 1:

\[
\text{Abs}(\text{row}[1])
\]
• The following example calculates the absolute value of the sum of column E:

Abs( column[E].sum )

• The following example points to expanded rows 1 through 3 within design segment 3 of the form Grid1:

Abs( Grid1.row[3(1:3)])

**Average**

Average is a mathematical function that returns the average of a group of numeric values, rows, columns, or cells. Average excludes #MISSING and #ERROR cells when obtaining the average.

*Note:*

The calculation doesn't include missing values regardless of whether they are suppressed or not.

The function syntax:

Average(arguments) or Avg(arguments)

where *arguments* is one or more of the following:

**Table B-7  Arguments for the Average Function**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Average (10, 20, 30) returns the value 20. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].Property For example Avg(Grid1.row[4(3:5)]) returns the average of form grid1, row 4, range 3 through 5.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Examples:**

• The following expression returns the value 20:

Avg( 10, 30, 20)
• The following example returns the average of all numbers that are part of three aggregate rows:

\[ \text{Average( row[1], row[6], row[8] )} \]

• The following example calculates the average of three aggregate columns; E, G, and I. The calculation produces three numbers and calculates the average of the three numbers:

\[ \text{Avg(column[E].avg, column[G].avg, column[I].avg)} \]

• The following example calculates the average of aggregate row 3 and divides the average by 100:

\[ \text{Avg(row[3])/100} \]

**AverageA**

AverageA is a mathematical function that returns the average of a group of numeric values, rows, columns, or cells. AverageA includes #MISSING and #ERROR cells, which are treated as zero values when obtaining the average.

---

**Note:**

#MISSING and #ERROR are included only for rows or columns that are not suppressed.

---

The function syntax:

\[ \text{AverageA(arguments) or AvgA(arguments)} \]

where arguments is one or more of these arguments:

**Table B-8  Arguments for the AverageA Function**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, AverageA (10, 20, 30) returns the value 20. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)]. Property. For example, AvgA(Grid1.row[4(3:5)]) returns the average of form grid1, row segment 4, range 3 through 5.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Example**
In the following example, a grid has 4 rows with the values 10, 20, 30, and, #ERROR. The following formula in the fifth row returns the value 15:

\[
\text{AverageA([1:4])}
\]

Count

Count is a mathematical function that returns the number of values in a group of numeric values, rows, columns, or cells. Count excludes #MISSING and #ERROR when obtaining the count. The function syntax:

\[
\text{Count (arguments)}
\]

where \text{arguments} is one or more of these arguments:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Count (10, 20, 30) returns the value 3. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].Property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

Examples:

- The following example returns the count of three rows, 1, 6, and 8:

\[
\text{Count(row[1], row[6], row[8])}
\]

- The following example returns the count of 3 columns:

\[
\text{Count( column[E], column[G], column[I] )}
\]

- The following example calculates the count of the cell located at row 4, column D:

\[
\text{Count(cell[D,4])}
\]

- The following example calculates the count of aggregate row 3 in grid 5:

\[
\text{Count(grid1.row[3])}
\]

CountA

CountA is a mathematical function that returns the number of values in a group of numeric values, rows, columns, or cells. CountA includes #MISSING and #ERROR cells
when obtaining the count only for rows or columns that are not suppressed. The function syntax:

\[
\text{CountA}(\text{arguments})
\]

where \textit{arguments} is one or more of these arguments:

**Table B-10  Arguments for the CountA Function**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, \text{CountA}(10,20,30,50) returns the value 4. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: \text{CountA}(Grid-Name.GridElement[segment(range)]).property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Examples:**

- In the following example, if a grid has 4 rows with the values 10, 20, 30, and, #ERROR. The following formula in the fifth row returns the count of four rows:

  \[
  \text{CountA}(\{1:4\})
  \]

- The following example returns the count of four rows:

  \[
  \text{CountA}(\text{row}[1], \text{row}[6], \text{row}[8] \text{where data yields #ERROR})
  \]

**Difference**

Difference is a mathematical function that returns the absolute value of the difference of a numeric value, row, or column subtracted from another numeric value, row, or column. This is also known as the variance. The function syntax:

\[
\text{Difference}(\text{arg1}, \text{arg2})
\]

where \textit{arg2} is subtracted from \textit{arg1} and are one or more of the following arguments:

**Table B-11  Arguments for the Difference Function**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, \text{Difference}(3, 5) returns the absolute value 2. Numeric values can include decimals and negative values.</td>
</tr>
</tbody>
</table>
### Table B-11 (Cont.) Arguments for the Difference Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>row, column, or reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].Property The following example returns the difference of two rows in form grid1: Difference( grid1.row[1], grid1.row[6] )</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Note:**

The Difference function returns the absolute value of arg2 subtracted from arg1, whereas the minus sign in subtraction negates a number.

**Examples:**

- The following example returns the absolute value of 8:
  
  \[
  \text{Difference}(3, -5) 
  \]

- The following example calculates the difference of two aggregate columns:
  
  \[
  \text{Difference}( \text{column}[\text{E}], \text{column}[\text{G}] ) 
  \]

**Note:**

You can type the text label "Difference" or "Variance."

### Eval

Eval is a mathematical function that evaluates an expression. You use Eval as an embedded function argument to consolidate multiple expressions into one expression. The function syntax:

\[
\text{Eval}(\text{expression})
\]

where expression is one or more of these arguments:
Table B-12 Arguments for the Eval Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: <code>GridName.GridColumn[segment(range)].Property</code></td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
<tr>
<td>operators</td>
<td>Use any of the supported arithmetic operators (+, -, *, /, ^, %, and so on).</td>
</tr>
</tbody>
</table>

Example:

The following example divides row 1 by row 2 and then rounds the data to four places:

```
Round(Eval([1]/[2]),4)
```

IfThen, If

IfThen is a conditional function that returns a value when the condition equals True, and another value when the condition equals False.

The function syntax:

```
IfThen(Condition, TrueParameter, FalseParameter)
```

- **Condition** is a logical expression that evaluates to true or false. Full conditional logic can be used as well as complex Boolean operators (And, Not, and Or). A condition can also test for #MISSING and #ERROR values. See the following table for a list of valid conditional operators.
- **TrueParameter** and **FalseParameter** are any valid expression that are evaluated based on the outcome of the condition.

The following table describes the conditional operators that are fully supported. Alternate syntax is listed wherever it’s supported by a conditional operator.
<table>
<thead>
<tr>
<th>Conditional Operator</th>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal To</td>
<td>expression = expression</td>
<td>Tests whether the left expression is equal to the right expression. The routine that evaluates the condition doesn't consider any rounding. If rounding is required, use the Round function. Example: 1 = 4 Returns false</td>
</tr>
<tr>
<td>Greater Than</td>
<td>expression &gt; expression</td>
<td>Tests whether the left expression is greater than the right expression. Example: 1 &gt; 4 Returns false</td>
</tr>
<tr>
<td>Greater Than or Equal To</td>
<td>expression &gt;= expression</td>
<td>Tests whether the left expression is greater than or equal to the right expression. The correct syntax is “&gt;=”. The syntax “=&gt;” isn't supported. Example: 1 &gt;= 4 Returns false</td>
</tr>
<tr>
<td>Less Than</td>
<td>expression &lt; expression</td>
<td>Tests whether the left expression is less than the right expression. Example: 1 &lt; 4 Returns true</td>
</tr>
<tr>
<td>Less Than or Equal To</td>
<td>expression &lt;= expression</td>
<td>Tests whether the left expression is less than or equal to the right expression. The correct syntax is “&lt;=”. The syntax “=&lt;” isn't supported. Example: 1 &lt;= 4 Returns true</td>
</tr>
</tbody>
</table>
### Table B-13  (Cont.) Conditional Operators

<table>
<thead>
<tr>
<th>Conditional Operator</th>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not Equal To</td>
<td>expression &lt;&gt; expression</td>
<td>Tests whether the left expression isn't equal to the right expression. The routine that evaluates the condition doesn't consider any rounding. If rounding is required, use the Round function.</td>
</tr>
<tr>
<td></td>
<td>expression != expression</td>
<td>Examples:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 &lt;&gt; 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 != 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true</td>
</tr>
<tr>
<td></td>
<td>IsMiss (reference)</td>
<td>If the reference is an expanded row or column, then all resulting cells must be #MISSING in order for the condition to be true.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IsMissing([1])</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true if row 1 has a #MISSING value.</td>
</tr>
<tr>
<td>IsError</td>
<td>IsError (reference)</td>
<td>Tests whether the reference contains an #ERROR result.</td>
</tr>
<tr>
<td></td>
<td>IsErr (reference)</td>
<td>If the reference is an expanded row or column, all resulting cells must be #ERROR in order for the condition to be true. Only formula rows and columns can result in #ERROR.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IsError([2])</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true if row 2 has a #ERROR value.</td>
</tr>
<tr>
<td>IsNonNumeric</td>
<td>IsNN (reference)</td>
<td>Tests whether the reference contains a #MISSING or #ERROR results.</td>
</tr>
<tr>
<td></td>
<td>IsNonNumerid (reference)</td>
<td>If the reference is an expanded row or column, all resulting cells must be #MISSING and/or #ERROR in order for the condition to be true.</td>
</tr>
<tr>
<td></td>
<td>IfNN (reference)</td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>IfNonNumber (reference)</td>
<td>IsNN([3])</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true if row 3 has a #MISSING or #ERROR value.</td>
</tr>
<tr>
<td>Parenthesis</td>
<td>(condition)</td>
<td>Used to group a condition. Used mostly for visual clarity.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1 &gt; 4)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns false</td>
</tr>
</tbody>
</table>
### Table B-14  Conditional Operators

<table>
<thead>
<tr>
<th>Complex Conditions</th>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>And</td>
<td><em>(condition AND condition)</em></td>
<td>Complex condition used to compare two conditions. Returns true if all conditions result in true. Example: <em>(1 &gt; 4 AND 5 &gt; 2)</em> Returns false</td>
</tr>
<tr>
<td></td>
<td><em>(condition &amp; condition)</em></td>
<td></td>
</tr>
<tr>
<td>Not</td>
<td>NOT <em>(condition)</em></td>
<td>Used to negate the result by reversing the result of the condition. Example: Not <em>(1 &gt; 4)</em> Returns true</td>
</tr>
<tr>
<td></td>
<td>! <em>(condition)</em></td>
<td></td>
</tr>
<tr>
<td>Or</td>
<td><em>(condition OR condition)</em></td>
<td>Complex condition used to compare two conditions. Returns true if any of the conditions result in true. Example: <em>(1 &gt; 4 OR 5 &gt; 2)</em> Returns true</td>
</tr>
<tr>
<td></td>
<td>*(condition</td>
<td>condition)*</td>
</tr>
</tbody>
</table>

### Notes on Conditions

- **Expression**—Any valid formula expression. The expression can be any combination of a constant (integer or real number), a reference, or another function.

- **Reference**—Any valid reference; thus the IFNN reference property can be used as part of the reference.

- **Condition**—Any valid condition applied to the complex conditions And, Not, and Or. Those operators can have embedded conditions.

**Note:**

And, Not, and Or operators require surrounding parentheses.

- When any expression within the condition returns an #ERROR or #MISSING value, the If function returns #MISSING or #ERROR. This doesn't apply when you use the IsMissing, IsError, or IsNonNumeric conditions.

### Complex Conditions

Complex conditions And, Or, and Not are fully supported. However, they must be surrounded by parentheses.

Valid example:

```
If (([A] > [B] and [A] > 1000), [A], [B])
```
Max

Max is a mathematical function that returns the maximum value in a group of numeric values, rows, columns, or cells. The function syntax:

\[ \text{Max} \left( \text{arguments} \right) \]

where \( \text{arguments} \) is one or more of these arguments:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, ( \text{Max}(10, 20, 30) ) returns the value 30. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].Property.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

Examples:

- The following example returns the maximum value in rows 1, 6, and 8:

\[ \text{Max}(\text{row}[1], \text{row}[6], \text{row}[8]) \]

- The following example calculates the maximum of the sums of aggregate rows:

\[ \text{Max}(\text{row}[1].\text{sum}, \text{row}[2].\text{sum}, \text{row}[3].\text{sum}) \]

Min

Min is a mathematical function that returns the minimum value in a group of numeric values, rows, columns, or cells. The function syntax:

\[ \text{Min} \left( \text{arguments} \right) \]

where \( \text{arguments} \) is one or more of these arguments:
Table B-16  Arguments for the Min Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Min (10, 20, 30) returns the value 10. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].Property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

Examples:

- The following example returns the minimum value in rows 1, 6, and 8:

  \[ \text{Min} \left( \text{row}[1], \text{row}[6], \text{row}[8] \right) \]

- The following example calculates the minimum of the sums of aggregate rows:

  \[ \text{Min} (\text{row}[1].\text{sum}, \text{row}[2].\text{sum}, \text{row}[3].\text{sum}) \]

Mod

Mod is a mathematical function that returns the remainder, or modulus, from a division. The function syntax:

\[ \text{Mod} (\text{arg1}, \text{arg2}) \]

where \( \text{arg2} \) is the divisor and \( \text{arg1} \) and \( \text{arg2} \) are one of these arguments:

Table B-17  Arguments for the Mod Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Mod (6, 4) returns the value 2. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].Property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

Example:

The following example divides 10 by 5 and returns the remainder of 0:

\[ \text{Mod} (10, 5) = 0 \]
PercentOfTotal

PercentOfTotal is a mathematical function that returns the result of a numeric value, row, column, or cell divided by another numeric value, row, column, or cell which is multiplied by 100. The function syntax:

\[
\text{PercentOfTotal} (\text{arg1, arg2})
\]

- where \( \text{arg1} \) is a component of the running total (\( \text{arg2} \)). Normally, this is a row or column reference.
- where \( \text{arg2} \) is the running total relative to \( \text{arg1} \). Normally this is a cell reference containing the grand total.
- \( \text{arg1} \) is divided by \( \text{arg2} \), with the result multiplied by 100. \( \text{arg1} \) and \( \text{arg2} \) are one or more of these arguments:

### Table B-18 Arguments for the PercentOfTotal Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, PercentOfTotal (100, 20) returns the value 500. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: Grid-Name.GridElement<a href="range">segment</a>].Property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Note:**

This function requires two arguments.

**Examples:**

- The following example returns the value of 5 percent.

\[
\text{PercentOfTotal}(20, 400)
\]

- The following example divides the value of each cell in column A by the Total Mkt value in cell A5, multiplies the result by 100, and displays the resulting PercentOfTotal in column B. The formula:

\[
\text{PercentOfTotal} ([A], [A,5]),
\]
Results for the PercentOfTotal Function

Using the above example, the following table shows the PercentOfTotal results in column B:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Sales</td>
</tr>
<tr>
<td>2</td>
<td>Mkt1</td>
<td>60</td>
</tr>
<tr>
<td>3</td>
<td>Mkt2</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>Mkt3</td>
<td>120</td>
</tr>
<tr>
<td>5</td>
<td>Total Mkt</td>
<td>300</td>
</tr>
</tbody>
</table>

**Tip:**

You enter the formula by clicking on the header for Column B and using the formula bar.

**Pi**

Pi is a mathematical function that returns the number 3.14159265358979, the mathematical constant, accurate to 15 digits. Pi is the ratio of the circumference of a circle to its diameter. The function syntax:

`PI()`

**Example:**

The following example returns row 3 divided by the product of Pi and 2:

`row[3] / (PI() * 2)`

**Product**

Product is a mathematical function that multiplies all numbers or references and returns the product. The function syntax:

`Product(arguments)`

where `arguments` is one or more of these arguments:

**Table B-19 Arguments for the Product Function**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Product(2, 20) returns the value 40. Numeric values can include decimals and negative values.</td>
</tr>
</tbody>
</table>
Table B-19  (Cont.) Arguments for the Product Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Example:**
The following example returns 40:

Product(2,20)

**Random**

Random is a mathematical function that returns a random number between 0.0 and 1.0. The function syntax:

Random()

**Example:**
The following example returns a random number between 0.0 and 1.0 and multiplies it by 1000:

Random() * 1000

**Round**

Round is a mathematical function that rounds a number up or down by the specified digits. The function syntax:

Round \( arg1, \text{integer} \)

where \( arg1 \) is one or more of these arguments:

Table B-20  Arguments for the Round Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Round(81.3987,3) returns the value 81.399. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>
*Integer* specifies the number of digits to which you want to round the number:

- If *integer* is greater than zero, the number rounds to the specified number of decimal places.
- If *integer* is zero, the number rounds to the nearest integer.
- If *integer* is less than zero, the number is rounded to the left of the decimal point.

**Examples:**

The following example rounds to three decimals:

\[
\text{Round}(3594.5567, 3) = 3594.557
\]

The following example rounds to the nearest integer:

\[
\text{Round}(3594.5567, 0) = 3595
\]

The following example rounds to the thousands. This is also known as scaling:

\[
\text{Round}(3594.5567, -3) = 4000
\]

**Sqrt**

Sqrt is a mathematical function that returns the square root of a numeric value, row, column, or cell. The syntax for the Sqrt function:

\[
\text{Sqrt}(\text{argument})
\]

where *argument* is one of these arguments:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Sqrt(100) returns the value 10. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

**Example**

The following example returns the value of 4:

\[
\text{Sqrt}(16)
\]
Sum

Sum is a mathematical function that returns the summation of a group of numeric values, rows, columns, or cells.

The syntax for the Sum function:

```
Sum (arguments)
```

where arguments is one or more of these arguments:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Sum(10, 20, 30) returns the value 60. Numeric values can include decimals and negative values.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: GridName.GridElement[segment(range)].property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

Examples:

- The following example returns the value 30:
  
  ```
  sum(10, 20)
  ```

- The following example returns the sum of three rows:
  
  ```
  sum(row[1], row[6], row[8])
  ```

- The following example calculates the sum of three aggregate columns:
  
  ```
  sum(column[E], column[G], column[I])
  ```

When a formula row includes IDESC, the sum includes all of the parents and their descendants.

For example, a form is created with the following rows (and each member has children):

```
IDESC("Mem1"), IDESC("Mem2"), IDESC("Mem3"), IDESC("Mem4")
```

If a formula row is added with the following formula:

```
SUM(ROW[1], ROW[2], ROW[3], ROW[4])
```

When the form is opened for data entry, the formula row will return the sum of all the parents and their children.
Truncate/Trunc

Truncate is a mathematical function that removes the specified number of digits from numeric values.

Syntax:

Trunc (arg1, integer)

- where arg1 is one of these arguments:

  Table B-23 Arguments for the Truncate/Trunc Function

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value; for example, 234.567.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: Grid-Name.GridElement[segment(range)].property</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function</td>
</tr>
</tbody>
</table>

- where integer specifies the number of digits you want to remove:
  - A positive integer determines the number of significant digits that remain to the right of the decimal point.
  - A zero (0) integer returns the integer located to the left of the decimal point.
  - A negative integer indicates the number of significant digits that are truncated to the left of the decimal point.

Examples:

The following statement uses a positive integer of 2. The first two digits to the right of the decimal point remain, and the following digit is removed:

\[
\text{Trunc}(234.567, 2) = 234.56
\]

The following statement uses a zero (0) integer. All digits to the right of the decimal point are removed:

\[
\text{Trunc}(234.567, 0) = 234
\]

The following statement uses a negative integer of -2. All digits to the right of the decimal point are removed and the last two digits of the integer are truncated.

\[
\text{Trunc}(234.567, -2) = 200
\]
Note:
Formatting previously applied to a cell, column, or row is maintained when you use the Trunc function. The following example shows the results of a Trunc function where the cell value was previously formatted to display three decimal places: \( \text{Trunc}(234.567, 0) = 234.000 \)

Variance/Var

Variance is a financial function that evaluates the difference between the specified values based on account type for the current account. For example, for Expense or Liability accounts, a positive result represents a decrease, so the result appears as a negative number. You can use this function with these UDA account types: Asset, Liability, Equity, Revenue, and Expense.

Syntax:

\[
\text{Var (reference1, reference2)}
\]

where \( \text{reference1} \) and \( \text{reference2} \) are references to a row, column, or cell that correspond to members of the same Account dimension whose variance results are to be calculated.

Expected Results

The following table describes the expected results when using the Variance function with Accounts.

<table>
<thead>
<tr>
<th>Column A</th>
<th>Column B</th>
<th>( \text{Var ([A] , [B])=0} )</th>
<th>( \text{Var ([A] , [B])&gt;0} )</th>
<th>( \text{Var ([A] , [B])&lt;0} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Asset</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Liability</td>
<td>Liability</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Equity</td>
<td>Equity</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Revenue</td>
<td>Revenue</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
</tbody>
</table>

Variance Behavior

- The Variance function expects comparison of the same account type. When you compare two different account types, like Sales & Expense, the Variance function performs the straight math without applying the logic of the account type. For example:
Table B-25  Results for the Variance Function When Comparing Two Different Account Types

<table>
<thead>
<tr>
<th>Sales</th>
<th>Expense</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>-400</td>
<td>100</td>
<td>-500</td>
</tr>
</tbody>
</table>

- When the Variance function is applied to a dimension that isn’t tagged as an Accounts type, an #ERROR results at runtime.
- #MISSING is treated as zero (0), unless specified differently using the ifnonumber property.

Examples

The Variance function accepts cell, column, or row references only.

Table B-26  Syntax Examples for the Variance Function

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample syntax referencing a column:</td>
<td>Var ([A], [B])</td>
</tr>
<tr>
<td>Sample syntax referencing a row:</td>
<td>Var ([3], [4])</td>
</tr>
<tr>
<td>Sample syntax referencing a cell:</td>
<td>Var (Cell [3,A], [3,B])</td>
</tr>
</tbody>
</table>

In this example, the variance between column A (Actual) and column B (Budget) is calculated as:

\[ \text{Var}([A], [B]) \]

This example produces the following report:

<table>
<thead>
<tr>
<th>Year</th>
<th>Product Market</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual</td>
<td>Budget Variance</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>========</td>
<td>==============</td>
</tr>
<tr>
<td>Sales (Income)</td>
<td>400,855</td>
</tr>
<tr>
<td>COGS (Expense)</td>
<td>179,336</td>
</tr>
</tbody>
</table>

VariancePercent/VarPer

VariancePercent is a financial function that evaluates the difference, in percent, between the specified values based on account type for the current account. For example, for an Income, Flow, Asset, or Balance account, a positive result represents an increase, so the result appears as a positive number. For Expense or Liability accounts, a positive result represents a decrease, so the result appears as a negative number.
Syntax:

VarPer (reference1, reference2)

where reference1 and reference2 are references to a row, column, or cell that correspond to members of the same Account dimension whose VariancePercent results are to be calculated.

Expected Results

The following table describes the expected results when using the VariancePercent function with Accounts tagged with the below UDAs.

<table>
<thead>
<tr>
<th>Col A</th>
<th>Col B</th>
<th>VarPer ([A], [B])=0</th>
<th>VarPer ([A], [B])&gt;0</th>
<th>VarPer ([A], [B])&lt;0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Asset</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Liability</td>
<td>Liability</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Equity</td>
<td>Equity</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Revenue</td>
<td>Revenue</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
</tbody>
</table>

VariancePercent Behavior

• The VariancePercent function expects comparison of the same account type. When you compare two account types, like Sales & Expense, the VariancePercent function performs the straight math without applying the logic of the account type. For example:

<table>
<thead>
<tr>
<th>Sales</th>
<th>Expense</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>-400</td>
<td>100</td>
<td>-5.</td>
</tr>
</tbody>
</table>

• When the VariancePercent function is applied to a dimension that isn't of type Accounts, an #ERROR results at runtime.

• #MISSING is treated as zero (0), unless specified differently using ifnonnumber property.

Examples

The VariancePercent function accepts, cell, column, or row references only.
Table B-29  Syntax Examples of the VariancePercent Function

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample syntax referencing a column</td>
<td>VarPer ([A], [B])</td>
</tr>
<tr>
<td>Sample syntax referencing a row</td>
<td>VarPer ([3], [4])</td>
</tr>
<tr>
<td>Sample syntax referencing a cell</td>
<td>VarPer (Cell [3,A], [3,B])</td>
</tr>
</tbody>
</table>

In this example, the VariancePercent between column A (Actual) and column B (Budget) is calculated as follows:

VarPer([A],[B])

This example produces the following report:

Example Result of the VariancePercent Function when Comparing Columns

<table>
<thead>
<tr>
<th>Sales (Income)</th>
<th>Year Actual</th>
<th>Product Budget</th>
<th>Market VariancePercent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>400,855</td>
<td>373,080</td>
<td>7%</td>
</tr>
<tr>
<td>COGS (Expense)</td>
<td>179,336</td>
<td>155,940</td>
<td>-13%</td>
</tr>
</tbody>
</table>
Optimizing the Application

Related Topics
• Optimizing Performance

Optimizing Performance

Use these methods to optimize performance:
• Strategically assign dimensions as dense or sparse and order them from most to least dense (see About Sparse and Dense Dimensions, About Reordering Dimensions, and Editing Dimensions in the Simplified Dimension Editor).
• Set the Suppress Missing Blocks option (see Setting Form Grid Properties).

See also:
• Writing #MISSING Values
• Other Performance Optimization Tips

About Reordering Dimensions

The order of dimensions is critical for the structure and performance of an application. Optimize performance when ordering dimensions:
• Make Period and Account dense, and order dense dimensions from most to least dense. The most dense is usually Period, followed by Account. Dense dimensions calculate faster than sparse dimensions.
• Separate sparse dimensions into aggregating and nonaggregating dimensions. Place aggregating dimensions before nonaggregating dimensions. Order sparse dimensions from most to least dense. Aggregating dimensions, such as Entity, consolidate children into the parent to create new data. Nonaggregating dimensions, such as Scenario, don't consolidate children to create data.

Writing #MISSING Values

#MISSING in a cell indicates that the cell contains no data, whereas zero in a cell is a data value. #MISSING decreases the database size and improves performance.

To write #MISSING to cells:
1. Select the cells to change.
   You can select contiguous cells by clicking in the upper-left cell in the range, and then pressing Shift+Click to select the lower-right cell in the range. You can select rows and columns by clicking row and column headings. Select a range of rows or columns by using Shift+Click.

   If the designer sets a form to suppress missing data, and an entire row contains #MISSING (no data), that row doesn't display on the form.
2. Take an action:
   • Press Delete.
   • Enter #missing.

You can also enter #MISSING using Smart Lists.

3. Click Save.

The cells are set to #MISSING when you save the form.

Other Performance Optimization Tips

• Set upper-level members in a dense dimension to Dynamic Calc.
• Clear unnecessary or historical data. For example, move historical data to a separate cube to reduce the database size in your current cube.
• Split large forms into multiple smaller forms having fewer rows and columns. Organize related forms into folders.
• Using account annotations affects performance, so use them sparingly.
• The first time the application is launched and forms are opened, caches are loaded, which takes more time than in subsequent sessions. So, before your users start using the application, launch the application and open the most commonly used forms to reduce the loading time for your users.
Understanding Application Artifacts

This appendix describes the artifacts for the components in Planning. For each artifact, the following information is displayed:

- **Artifact**—Artifact name
- **Description**—Artifact description
- **Last Modified Time**—If this parameter is supported, reflects the time the artifact was imported. Some artifacts don't support the last modified time parameter.
- **Last Modified User**—If this parameter is supported, reflects the user who last modified the artifact. Some artifacts don't support the last modified user parameter.
- **Editable**—If yes, you can download the artifact to the desktop by right-clicking the artifact and selecting **Export Artifact for Edit**. After it's modified, you can import the artifact back into Planning by right-clicking the artifact and selecting **Import Artifact After Edit**.
- **Dependencies**—Lists artifact dependencies

Planning Artifacts

Planning artifacts are listed under the Planning node on the **Categories** tab of Migration.

Configuration Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adhoc Options</td>
<td>Suppress options, precision options, replacement options, and other miscellaneous options that affect data in web grid</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>Ad Hoc Forms</td>
</tr>
<tr>
<td>Data Load Settings</td>
<td>Parameters that users can set to enable data to be loaded directly into an Essbase database</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>Associated Dimensions</td>
</tr>
<tr>
<td>Properties—Application Definition and Application Settings</td>
<td>Functionality that enables users to set preferences for aspects such as email notification, alias tables, and display options</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>User Preferences</td>
<td>Preferences that users can set for applications, display, printing, and user variables</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>User Variables</td>
</tr>
</tbody>
</table>
Table D-1  (Cont.) Configuration Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Variables</td>
<td>Dynamically render forms based on a users member selection, displaying only the specified entity; for example, a user variable named Department displays specific departments and employees</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>Associated Dimensions</td>
</tr>
</tbody>
</table>

Essbase Data Artifacts

Note:
You import Essbase data only when migrating an application from production to production/test (cloning). You should not import Essbase data artifacts when you’re doing an incremental update.

Table D-2  Essbase Data Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable on the File System?</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Essbase Data</td>
<td>Planning Essbase data</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>None</td>
</tr>
</tbody>
</table>

Global Artifacts

Table D-3  Global Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute Dimensions</td>
<td>Dimensions whose type enables analysis based on the attributes or qualities of dimension members</td>
<td>Yes</td>
<td>No</td>
<td>Yes—CSV</td>
<td>None</td>
</tr>
<tr>
<td>Calculation Manager Rule-sets</td>
<td>Objects that contain rules and other rule-sets that can be calculated simultaneously or sequentially</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>Composite Forms</td>
<td>Displays members from several forms simultaneously so you can, for example, enter data into one grid and see the results—such as Total Revenue—aggregated in another</td>
<td>Yes</td>
<td>No</td>
<td>Yes—XML</td>
<td>Associated Forms</td>
</tr>
<tr>
<td>Artifact</td>
<td>Description</td>
<td>Last Modified Time</td>
<td>Editable</td>
<td>Dependencies</td>
<td></td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
<td>----------</td>
<td>---------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Custom Menus</td>
<td>Menus that administrators create that are company- or application-specific. Users can right-click a member and select a menu item to open a URL, data form, or workflow.</td>
<td>Yes</td>
<td>No</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If menu is of type Workflow, then Approval Units</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dashboards</td>
<td>At-a-glance views of key information, organized and presented in a way meaningful to a business need.</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Associated Data Forms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exchange Rates</td>
<td>A numeric value for converting one currency to another; for example, to convert 1 USD into EUR, the exchange rate of 0.8936 is multiplied with the U.S. dollar. The European euro equivalent of $1 is 0.8936.</td>
<td>Yes</td>
<td>No</td>
<td>Yes—CSV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Currency, Period, and Year</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jobs</td>
<td>Customized actions</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Associated Dimensions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Approval Unit Hierarchies</td>
<td>Specifies which application approval units and members are part of the budget process</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Entity, Scenario, Version and other associated dimensions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Report Mappings</td>
<td>Maps dimensions between Planning applications and reporting applications to enable reporting on Planning data in a reporting application, aggregations and queries on Smart Lists, and linking Planning data to multiple reporting applications for consolidations</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Associated dimensions and Smart Lists</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Schedules</td>
<td>Scheduling information of actions that are set to run at intervals.</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Associated Business Rules, Report Mappings, Jobs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smart Lists</td>
<td>Custom drop-down lists that users access from data form cells (instead of entering data)</td>
<td>No</td>
<td>No</td>
<td>Yes—CSV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>None</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spread Patterns</td>
<td>A custom spreading pattern that determines how data is distributed from a parent to its children. The pattern is available from the Mass Allocate and Grid Spread menus.</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>None</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard Dimensions</td>
<td>Dimensions associated with a single cube</td>
<td>Yes</td>
<td>No</td>
<td>Yes—CSV</td>
<td></td>
</tr>
<tr>
<td></td>
<td>None</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Substitution Variables</td>
<td>Global placeholders for information that changes regularly</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td></td>
<td>None</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Lists</td>
<td>A detailed status list of tasks for a particular user</td>
<td>Yes</td>
<td>No</td>
<td>Yes—XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If task is of type Data Form, then associated Data Form. If task is of type Workflow, then Approval Units</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Cube Artifacts

**Table D-4  Cube Artifacts**

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute Dimensions</td>
<td>A list of dimensions whose type enables analysis based on the attributes or qualities of dimension members</td>
<td>Yes</td>
<td>No</td>
<td>Yes—CSV</td>
<td>None</td>
</tr>
<tr>
<td>Calculation Manager Rules</td>
<td>Objects that can contain templates and calculations that are grouped in components</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>Data Forms</td>
<td>A grid display that enables users to enter data into the database from an interface such as a web browser, and to view and analyze data or related text. Certain dimension member values are fixed, giving users a specific view into the data. Forms can include predefined data validation rules that help implement business policies and practices. Errors or warnings are generated on the data form if entered data violates a validation rule.</td>
<td>Yes</td>
<td>No</td>
<td>Yes—XML</td>
<td>Associated menus, user variables, and dimensions</td>
</tr>
<tr>
<td>Standard Dimensions</td>
<td>A list of dimensions associated with a single cube</td>
<td>Yes</td>
<td>No</td>
<td>Yes—CSV</td>
<td>Attribute Dimensions, if any</td>
</tr>
<tr>
<td>Substitution Variables</td>
<td>Global placeholders for information that changes regularly</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>None</td>
</tr>
</tbody>
</table>

Relational Data Artifacts

**Table D-5  Relational Data Artifacts**

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Annotations</td>
<td>Comments associated with accounts that can be plain text or URL links</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>Account, Entity, Scenario, and Version Dimensions</td>
</tr>
<tr>
<td>Announcements</td>
<td>Announcements that typically represent company information, demonstrations, and so on</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>Cell Texts</td>
<td>Text annotations associated with cells</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>Approval Units</td>
<td>A data slice at the intersection of a scenario, version, and entity; the basic unit for preparing, reviewing, annotating, and approving plan data</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>Entity, Scenario, and Version Dimensions</td>
</tr>
<tr>
<td>Supporting Detail</td>
<td>Calculations and assumptions from which the values of cells are derived</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
</tbody>
</table>
# Table D-5  (Cont.) Relational Data Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tablet Access</td>
<td>Planning artifacts that are enabled for mobile devices</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>Text Values</td>
<td>Text that is stored as data in cells whose data type is text</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
</tbody>
</table>

## Security Artifacts

---

**Note:**

Access permissions are a set of operations that a user can perform on a resource.

---

# Table D-6  Security Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Description</th>
<th>Last Modified Time</th>
<th>Last Modified User</th>
<th>Editable</th>
<th>Dependencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Permissions—Users</td>
<td>Company personnel who are provisioned as valid system users</td>
<td>No</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
<tr>
<td>Access Permissions—Groups</td>
<td>A container for assigning similar access permissions to multiple users</td>
<td>Yes</td>
<td>No</td>
<td>Yes—XML</td>
<td>None</td>
</tr>
</tbody>
</table>
Using Smart View to Manage Applications

Related Topics

• About Managing Applications in Smart View
• Planning Admin Extension and Office AutoCorrect
• Controlling the Display of Application Management Options in Smart View
• Downloading the Application Templates
• Creating an Application
• Working with Artifacts in the Application Template
• Updating an Application in Smart View
• Deleting an Application

About Managing Applications in Smart View

Administrators can use Excel to create and manage applications in Oracle Smart View for Office.

Using a downloadable Excel workbook template, you can quickly create, model, and manage applications in Smart View. By modifying the worksheets within the application template workbook, you can define application properties, dimension and member properties, security (access permissions), and substitution variables to fit your needs. You can also load data from the template.

Once the application is created, it’s available to users from both Smart View and the web application. The application can be further updated and refreshed in Smart View as needed. You can also delete the application from Smart View.

Application management in Smart View is a quick and easy way to get an application up and running in your business process environment.

Get started by downloading and then editing the application template workbook available in Smart View. Or, download the application template zip file from Planning. The template zip file contains starter workbooks for Planning applications, and a sample Planning application.

Watch this overview video to learn about creating an application in Smart View.

Overview video

Planning Admin Extension and Office AutoCorrect

In the Planning Admin Extension, Office AutoCorrect can affect member editing with the Planning Admin Extension in Oracle Smart View for Office.
When editing member properties, such as member names, member descriptions, aliases, or member formulas of the data type Text, you may have names that begin with two initial capital letters; for example

DGreen

Excel will autocorrect this name to one initial capital; for example:

Dgreen

In Excel, you can set an AutoCorrect option to retain two initial capital letters in a name.

To retain two initial capital letters in a name:

1. In Excel, access the Excel Options dialog.
2. Select Proofing, and then click the AutoCorrect Options button.
3. In the AutoCorrect dialog, clear the Correct Two Initial Capitals check box.
4. Click OK to close the AutoCorrect dialog, and then click OK to close the Excel Options dialog.

The change takes effect immediately. There is no need to restart Excel.

Controlling the Display of Application Management Options in Smart View

When a new application is created in either Planning or Oracle Smart View for Office, all application management options are displayed by default to administrators in Smart View.

You can choose to hide or display two of the options—Update Application and Delete Application—using the Suppress Application Management Options in Smart View option in Planning.

This setting, and the display of the application management options, applies to administrators only.

To control the display of the Update Application and Delete Application options in Smart View:

1. From the Home page, click Application, and then click Settings.
2. Choose a setting for Suppress Application Management Options in Smart View:
   • Yes—Application options are not displayed in Smart View.
   • No—Default. Application options are displayed in Smart View.
3. Click Save.
Note:

Any time a new application is created, whether in Planning or Smart View, the Suppress Application Management Options in Smart View setting defaults to No and the Update Application and Delete Application options will automatically be displayed. To hide the options, perform the above procedure.

Downloading the Application Templates

Related Topics

• Downloading the Template in Smart View
• Downloading the Application Template Zip File from the Business Process

Downloading the Template in Smart View

When you’re logged into Oracle Smart View for Office as an administrator, the Download Template command is available in the Smart View Panel.

The default template that you download can be used immediately to begin creating a Planning application. You can modify the template to create your own custom application.

To immediately create the Vision application for Planning, complete the task in Downloading the Application Template Zip File from the Business Process to download the PlanningApplicationTemplate_sample.xlsx

Note:

Before you begin, ensure that you have installed Smart View. Additionally, you must have administrator privileges to create, edit, and delete applications.

To download the application template:

1. Launch Excel and, from Smart View, log in to the applicable data source.
2. In the Smart View Panel, select the Download Template command.

   The default application template, ApplicationTemplateFile.xlsx, is opened. The default template contains the basic elements required to begin creating a standard Planning application.

   You can edit the default template to create a custom application, as described in Working with Artifacts in the Application Template.
Downloading the Application Template Zip File from the Business Process

When you’re logged into Planning as administrator, you can download a zip file containing these application template workbook files:

- PlanningApplicationTemplate.xlsx—For creating a Planning application.
- PlanningApplicationTemplate_epbcs.xlsx—For creating a Planning Modules application.
- PlanningApplicationTemplate_sample.xlsx—For creating the Planning sample application, Sample.

Note:

Before you begin, ensure that you have installed Oracle Smart View for Office. Additionally, you must have administrator privileges to create, edit, and delete applications.

To download the application template zip file:

1. From the Windows computer where you want to install the client software, access a business process instance.
2. On the Home page, access Settings and Actions by clicking your user name at the top right corner of the screen.
3. Click Downloads.
4. In the Downloads page, Planning Extension section, click Download for Application Templates.

Note:

The Planning Admin Extension isn't required to create and manage applications in Smart View.

5. Save the zip file locally to a server location that you can access, and unzip the contents.

You should now have these files saved:

- PlanningApplicationTemplate.xlsx
- PlanningApplicationTemplate_epbcs.xlsx
- PlanningApplicationTemplate_sample.xlsx

6. Open the template file you require and modify according to the information in Working with Artifacts in the Application Template.
Creating an Application

From Oracle Smart View for Office, administrators can create a new application based on either the downloaded template with no modifications or a modified template. In either case, the template must adhere to template rules and guidelines.

When you create an application, you can:

• Define cubes
• Define dimensions
• Add a limited amount of data
• Define security and access permissions
• Define substitution variables

Only administrators can create an application and only when no application exists in the cloud business process. Once an application is created, the option is no longer available in the Smart View Panel.

Before you begin, you should launch Excel and, from Smart View, log in to the applicable data source.

To create a new application in Smart View:

1. If you have not already done so, perform one of the following actions:
   • Download the application template as described in Downloading the Template in Smart View.
     Downloading the application template simply opens the template in Smart View.
   • Download and unzip the application template zip file as described in Downloading the Application Template Zip File from the Business Process, then open one of the template files.
   • Open a saved template file.

An Excel application template workbook must be open when creating an application.

2. Optional: Edit and then save the application template as needed.

   See Working with Artifacts in the Application Template for information on modifying the worksheets in the application template workbook.

3. With the application template workbook open in Smart View, select Create New Application.

   The application creation status appears in the lower left corner of Excel.

4. When the application creation process is complete, verify that the application appears in the Smart View Panel.

   Note that only one application per cloud business process is allowed. Once an application is created, the Create New Application command no longer appears in the Smart View Panel.

   The application is now available to the applicable Planning users.
Next steps include updating the application template workbook. You can also delete the application and create a new one. See:

- Working with Artifacts in the Application Template
- Updating an Application in Smart View
- Deleting an Application

**Working with Artifacts in the Application Template**

**Related Topics**

- About Working with Artifacts in the Application Template
- Application Definition
- Dimension Definition
- Attribute Dimension Definition
- Data Definition
- Substitution Variable Definition
- Security Definition
- Advanced Settings Definition

**About Working with Artifacts in the Application Template**

Editing the application template involves editing Excel worksheets in the application workbook.

The application template workbook contains sheets for defining application properties, cubes, dimension and member properties, substitution variables, and security. You can also load data using the template.

**Application Definition**

In the application template workbook, the Definition worksheet defines your application and cubes. Using the application definition worksheet provided in the template for the Sample application as your guideline, you can edit the worksheet provided in the template.

Follow this workflow for setting up a data worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define Application Properties
4. Define Cubes

**Define the Worksheet Name**

The worksheet name is in the Excel worksheet tab. The first tab in the workbook is for the application definition worksheet. Ensure that the application definition worksheet is named, **Definition. For example:**
Define the Worksheet Type

In cell A1 in the application worksheet, type Application Definition for the worksheet type. In Figure 1, cell A1 contains the worksheet type, Application Definition.

Figure E-1 Worksheet Type and Application Properties Shown in Application Definition Worksheet

Define Application Properties

Application properties are the basic information needed to create a Planning application. In general, only the Application name and Description are required properties. For all other properties, if not specified, the default values will be provided during application creation.

Table 1 shows the supported properties, their default values, and valid values.

Table E-1 Application Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Required</th>
<th>Value Type</th>
<th>Default Value</th>
<th>Valid Values</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Name</td>
<td>Yes</td>
<td>Text</td>
<td></td>
<td></td>
<td>Name of Planning application</td>
</tr>
<tr>
<td>Property</td>
<td>Required</td>
<td>Value Type</td>
<td>Default Value</td>
<td>Valid Values</td>
<td>Comments</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------</td>
<td>------------</td>
<td>---------------</td>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Description</td>
<td>Yes</td>
<td>Text</td>
<td></td>
<td></td>
<td>Planning application description</td>
</tr>
<tr>
<td>Application Type</td>
<td>No</td>
<td>Text</td>
<td>Standard</td>
<td>Standard</td>
<td>Similar to choices on web interface</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>An advanced application with two cubes and the option to add more later</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Standard—Advanced</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>also referred to as EPBCS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Advanced</td>
<td></td>
</tr>
<tr>
<td>Planning Frequency</td>
<td>No</td>
<td>Text</td>
<td>Monthly</td>
<td>Monthly</td>
<td>Applicable for Monthly Planning frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Weekly</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Quarterly</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Custom</td>
<td></td>
</tr>
<tr>
<td>Weekly Distribution</td>
<td>No</td>
<td>Text</td>
<td>445</td>
<td>Even</td>
<td>Applicable for Monthly Planning frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>445</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>454</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>544</td>
<td></td>
</tr>
<tr>
<td>Periods Per Year</td>
<td>No</td>
<td>Numeric</td>
<td>13</td>
<td></td>
<td>Only for Custom Planning Frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Year in 4 digits</td>
<td></td>
</tr>
<tr>
<td>Prefix</td>
<td>No</td>
<td>Text</td>
<td>TP</td>
<td></td>
<td>Only for Custom Planning Frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Year in 4 digits</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Calculated as (Start Year + Number of Years + 1)</td>
<td></td>
</tr>
<tr>
<td>Start Year</td>
<td>No</td>
<td>Numeric</td>
<td>2010</td>
<td></td>
<td>Year in 4 digits</td>
</tr>
<tr>
<td>End Year</td>
<td>No</td>
<td>Numeric</td>
<td>2019</td>
<td></td>
<td>Year in 4 digits</td>
</tr>
<tr>
<td>First Month of Fiscal Year</td>
<td>No</td>
<td>Text</td>
<td>January</td>
<td></td>
<td>Only for Monthly Planning Frequency</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Jan through Dec</td>
<td></td>
</tr>
<tr>
<td>First Fiscal Year Start Date</td>
<td>No</td>
<td>Date</td>
<td></td>
<td></td>
<td>Only for Quarterly or Custom Planning frequency</td>
</tr>
<tr>
<td>Rolling Forecast</td>
<td>No</td>
<td>Boolean</td>
<td>Yes</td>
<td>True</td>
<td>False</td>
</tr>
</tbody>
</table>
Table E-1  (Cont.) Application Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Required</th>
<th>Value Type</th>
<th>Default Value</th>
<th>Valid Values</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rolling Forecast Period</td>
<td>No</td>
<td>Numeric</td>
<td>6</td>
<td>1 to 36</td>
<td>Only when Rolling Forecast is True</td>
</tr>
<tr>
<td>Duration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Main Currency</td>
<td>No</td>
<td>Text</td>
<td>USD</td>
<td>Any supported currency code</td>
<td></td>
</tr>
<tr>
<td>Multi Currency</td>
<td>No</td>
<td>Boolean</td>
<td>Yes</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>Multi Currency Type</td>
<td>No</td>
<td>Text</td>
<td>Standard</td>
<td>Standard</td>
<td>Simplified</td>
</tr>
<tr>
<td>Sandboxes</td>
<td>No</td>
<td>Boolean</td>
<td>Yes</td>
<td>True</td>
<td>False</td>
</tr>
<tr>
<td>Also referred to as Sandbox</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Define Cubes

In the application worksheet, the same sheet where you define the application properties, list all Planning cubes.

To define cubes in the application worksheet, add a line with the label, Cubes after the application definition section. For example, in Table 2, a line with the word Cubes is added to line 18 of the worksheet.

In the next row, add the Name label to column A, and the Type label to column B.

Figure E-2  Cube Definition Section of Application Definition Worksheet
Table 2 shows the properties related to cube definition. Only the cube name is required. For the Type column, use BSO or ASO. Note that if the type is left blank, BSO is used by default.

Table E-2  Cube Properties

<table>
<thead>
<tr>
<th>Header Name</th>
<th>Required</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Yes</td>
<td>Name of the cube</td>
</tr>
<tr>
<td>Type</td>
<td>BSO</td>
<td>BSO is the default. If cell is left blank, BSO is used</td>
</tr>
<tr>
<td></td>
<td>ASO</td>
<td></td>
</tr>
</tbody>
</table>

Dimension Definition

In the template workbook, create a new worksheet for each dimension in your application. You can use the worksheets provided in the template for the Sample application as your guideline.

Follow this workflow for setting up a dimension worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Dimension Name
4. Define Members

Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. For a dimension worksheet name, use the dimension name prefixed with "dim.". For example, for the Scenario dimension, you would name the worksheet Dim.Scenario. For example:

Define the Worksheet Type

In cell A1 in the dimension worksheet, type Dimension for the worksheet type. In Figure 1, cell A1 contains the worksheet type, Dimension.

Figure E-3  Worksheet Type and Dimension Name Shown in a Dimension Worksheet of Application Template
Define the Dimension Name

In the dimension worksheet, type the dimension name. In Figure 1, cell B3 contains the dimension name, Scenario.

If the dimension name that you provide doesn’t exist, then a new custom type dimension is automatically created when you create the application.

Define Members

In the dimension worksheet, under the Members heading in cell A5, add members and their properties. Your list of members should conform to the format described in Importing and Exporting Data and Metadata. You may add columns only for required properties. Any missing properties will be added based on application, cube, and dimension defaults.

Figure E-4    Example Dimension Worksheet for the Scenario Dimension

Attribute Dimension Definition

In the template workbook, create a new worksheet for each attribute dimension in your application. You can use the attribute dimension worksheet provided in the template for the Vision application as your guideline.

Follow this workflow for setting up an attribute dimension worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Attribute Dimension Name
4. Define the Attribute Dimension Type
5. Define the Base Dimension
6. Define Attribute Members
Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. For the worksheet name, use the dimension name prefixed with "Attribute.". For example, for the Location attribute dimension, you would name the worksheet Attribute.Location. For example:

![Worksheet Name Example](image)

Define the Worksheet Type

In cell A1 in the attribute dimension worksheet, type Attribute for the worksheet type. In Figure 1, cell A1 contains the worksheet type, Attribute.

Figure E-5    Worksheet Type and Attribute Dimension Properties Shown in an Attribute Dimension Worksheet of Application Template

Define the Attribute Dimension Name

In the attribute dimension worksheet, type the dimension name. In Figure 1, cell B3 contains the attribute dimension name, Location.

Define the Attribute Dimension Type

In the attribute dimension worksheet, type the attribute dimension type. In Figure 1, cell B4 contains the attribute dimension type, Text.

Valid attribute dimension types are:

- Text
- Numeric
- Boolean
- Date

Define the Base Dimension

In the attribute dimension worksheet, type the base dimension for the attribute dimension. In Figure 1, cell B5 contains the base dimension, Entity.

Define Attribute Members

In the dimension worksheet, under the Members heading in cell A5, add attribute members and their properties. Your list of attribute members should conform to the format
described in Importing and Exporting Data and Metadata. You may add columns only for required properties. Any missing properties will be added based on application, cube, and dimension defaults.

Data Definition

With the application creation template, you can load up to 1,000 rows of data using the Essbase data format.

In the template workbook, create a worksheet for your application data, using the data worksheet provided in the template for the Sample application as your guideline. Or, edit the worksheet provided in the template.

Follow this workflow for setting up a data worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Cube Name
4. Define Data

Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. For the worksheet name, use the cube name where data will be loaded, prefixed with Data.. For example, to load data to the Plan1 cube, you would name the worksheet Data.Plan1 to indicate that you’re loading data to Plan1. After the required Data. prefix, you may add any other indicators to the name to help you identify the data. For example, to show that you’re loading actual data versus historical data to Plan 1, you would name the worksheet Data.Plan1.Actual.

Define the Worksheet Type

In cell A1 in the data worksheet, type Cube to signify that data in this sheet will be loaded to a cube. In Figure 1, cell A1 contains Cube to indicate that the worksheet will contain data for loading to a cube.
Define the Cube Name

In the data worksheet, type the name of the cube to which to load data. In Figure 1, cell B3 contains the cube name, Plan1.

Define Data

Define data in the Essbase columnar format. Use “Dimension” as the header for each member intersection.

Substitution Variable Definition

In the application template workbook, include a worksheet for your substitution variables, using the substitution variable definition worksheet provided in the template for the Sample application as your guideline.

Follow this workflow for setting up a substitution variable worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Type
3. Define the Substitution Variables

Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. Name the substitution variable definition worksheet, Variables. For example:

Define the Worksheet Type

In cell A1 in the data worksheet, type Substitution Variables to signify that data in this sheet will be loaded to a cube. In Figure 1, cell A1 contains Substitution Variables to indicate that the worksheet will contain substitution variable information for loading to the specified cubes in the application.
Define the Substitution Variables

In the Variables worksheet, using Figure 1 for reference, add information for each substitution variable in the following format:

- In the **Cube** column, enter the name of the cube to which the substitution variable will apply. For example:
  - If a variable applies to all cubes, enter *All Cubes*
  - If a variable applies to a specific cube, enter the cube name; for example, enter *Plan3*
  - If a variable applies to more than one cube, but not all cubes, list each cube name on its own row individually; for example, if a variable applies to both Plan2 and Plan3 only, add an entry for *Plan2* on one row, and an entry for *Plan3* on another row.

- In the **Name** column, enter the substitution variable name. In Figure 1, the substitution variable names are *CurrYr, NextYr, and CurVersion*.

- In the **Value** column, enter the value to be used for the substitution variable.

Security Definition

In the application template workbook, include a worksheet for user permissions, using the security definition worksheet provided in the template for the Vision application as your guideline.

Follow this workflow for setting up a user permissions worksheet in your application template workbook:

1. **Define the Worksheet Name**
2. **Define the Worksheet Type**
3. **Define Security Properties**
Define the Worksheet Name

Define the worksheet name in the Excel worksheet tab. Name the security worksheet where you'll define user permissions, Security. For example:

![Worksheet Name](image)

Define the Worksheet Type

In cell A1 in the data worksheet, type Security to signify that data in this sheet will be loaded to a cube. In Figure 1, cell A1 contains Substitution Variables to indicate that the worksheet will list all user access permissions for the application. The user permission data in the security sheet should be in Planning .sec file format.

Figure E-8   Worksheet Type Shown in Security Worksheet of Application Template

![Worksheet Type](image)

Define Security Properties

Security properties include the user name and other user access information. Table 1 shows the supported security properties.

Table E-3   Security Properties

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the user or group</td>
</tr>
<tr>
<td>Object Name</td>
<td>Planning artifact name</td>
</tr>
<tr>
<td>Access Mode</td>
<td>Permission granted. Valid values:</td>
</tr>
<tr>
<td></td>
<td>• READ</td>
</tr>
<tr>
<td></td>
<td>• WRITE</td>
</tr>
<tr>
<td></td>
<td>• READWRITE</td>
</tr>
<tr>
<td></td>
<td>• LAUNCH</td>
</tr>
<tr>
<td></td>
<td>• NOLAUNCH</td>
</tr>
<tr>
<td></td>
<td>• NONE</td>
</tr>
<tr>
<td>Flag</td>
<td>Member function to be used while applying the access</td>
</tr>
<tr>
<td>Object Type</td>
<td>Planning artifact object type. Default is Member.</td>
</tr>
</tbody>
</table>
Advanced Settings Definition

In the application template workbook, include a worksheet for advanced settings, using the advanced settings worksheet provided in the template for the Sample application as your guideline.

Advanced settings are:

• Evaluation order
• Performance settings
• Dimension properties such as hierarchy type, two-pass calculations, and data storage options
• Density (for aggregate storage applications only)
• Dimension assignments to cubes

Follow this workflow for setting up an advanced settings worksheet in your application template workbook:

1. Define the Worksheet Name
2. Define the Worksheet Name
3. Define Evaluation Order
4. Define Performance Settings
5. Define Dimension Settings
6. Define Density Settings
7. Assign Dimensions to Cubes

Define the Worksheet Name

Name the advanced settings definition worksheet, Advanced Settings. For example:

![Advanced Settings](image)

Define the Worksheet Type

In cell A1 in the data worksheet, type Advanced Settings to signify that this sheet contains the advanced, or miscellaneous settings for the application. In Figure 1, cell A1 contains the worksheet type, Advanced Settings, to indicate that the worksheet will contain advanced settings for the application.

**Figure E-9  Worksheet Type Shown in Advanced Settings Worksheet of Application Template**

![Advanced Settings](image)
Define Evaluation Order

In the Advanced Settings worksheet, define the evaluation order of dimensions in the "Evaluation Order" section. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns. The dimensions in each cube can have a different evaluation order.

<table>
<thead>
<tr>
<th>Evaluation Order</th>
<th>Cubes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimensions</td>
<td>Plan1</td>
</tr>
<tr>
<td>Account</td>
<td>3</td>
</tr>
<tr>
<td>Period</td>
<td></td>
</tr>
<tr>
<td>Entity</td>
<td>1</td>
</tr>
<tr>
<td>Year</td>
<td></td>
</tr>
<tr>
<td>Scenario</td>
<td>2</td>
</tr>
<tr>
<td>Version</td>
<td></td>
</tr>
<tr>
<td>HSP_View</td>
<td></td>
</tr>
</tbody>
</table>

Define Performance Settings

For Planning applications only.

In the Advanced Settings worksheet, define dimension storage property of each dimension for each cube in the "Performance Settings" section. Valid values are Dense or Sparse. Each cube requires at least one dense dimension. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns. The sparse or dense value of each dimension can differ between cubes.

<table>
<thead>
<tr>
<th>Performance Setting</th>
<th>Cubes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimensions</td>
<td>Plan1</td>
</tr>
<tr>
<td>Account</td>
<td>Dense</td>
</tr>
<tr>
<td>Period</td>
<td>Dense</td>
</tr>
<tr>
<td>Entity</td>
<td>Sparse</td>
</tr>
<tr>
<td>Year</td>
<td>Sparse</td>
</tr>
<tr>
<td>Scenario</td>
<td>Sparse</td>
</tr>
<tr>
<td>Version</td>
<td>Sparse</td>
</tr>
<tr>
<td>HSP_View</td>
<td>Sparse</td>
</tr>
</tbody>
</table>

Define Dimension Settings

In the Advanced Settings worksheet, define dimensions properties in the “Dimension Settings” section. Dimensions are listed in the first column under the heading “Dimensions”. Dimension properties are listed in subsequent columns. See Table 1 for the list of dimension properties.
Table E-4  Dimension Properties Used in Dimension Settings Section of the Advanced Settings Worksheet

<table>
<thead>
<tr>
<th>Dimension Property</th>
<th>Description or Valid Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Optional descriptive text</td>
</tr>
<tr>
<td>Alias Table</td>
<td>Optional alias table</td>
</tr>
<tr>
<td>Hierarchy Type</td>
<td>• Dynamic</td>
</tr>
<tr>
<td></td>
<td>• Store</td>
</tr>
<tr>
<td>Two Pass Calculation</td>
<td>Yes</td>
</tr>
<tr>
<td>Apply Security</td>
<td>Yes</td>
</tr>
<tr>
<td>Data Storage</td>
<td>• Store</td>
</tr>
<tr>
<td></td>
<td>• Dynamic Calc and Store</td>
</tr>
<tr>
<td></td>
<td>• Dynamic Calc</td>
</tr>
<tr>
<td></td>
<td>• Never Share</td>
</tr>
<tr>
<td></td>
<td>• Label Only</td>
</tr>
<tr>
<td></td>
<td>• Shared</td>
</tr>
<tr>
<td>Display Option</td>
<td>Set application default display options for the Member Selection dialog box. Select Member Name or Alias to display members or aliases. Member Name:Alias displays members on the left and aliases on the right. Alias:Member Name displays aliases on the left and members on the right.</td>
</tr>
<tr>
<td></td>
<td>• Member Name</td>
</tr>
<tr>
<td></td>
<td>• Alias</td>
</tr>
<tr>
<td></td>
<td>• Member Name:Alias</td>
</tr>
<tr>
<td></td>
<td>• Alias:Member Name</td>
</tr>
</tbody>
</table>

You can also use Figure 2 for reference in adding information to the Dimension Settings section.

Figure E-10  Dimension Settings Section of the Advanced Settings Worksheet

Define Density Settings

For Planning Modules applications only.

In the Advanced Settings worksheet, define the dimension storage property of each dimension for each cube in the "Density" section. Each cube requires at least one dense dimension. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns. The sparse or dense value of each dimension can differ between cubes.
Assign Dimensions to Cubes

In the Advanced Settings worksheet, assign dimensions to cubes in the "Dimension Valid for" section. Dimensions are listed in the first column under the heading "Dimensions". Cubes are listed in subsequent columns.

For each dimension and cube intersection:

- Type Yes to include the dimension in the cube.
- Type No or leave the cell blank to exclude the dimension from the cube.

### Updating an Application in Smart View

From Oracle Smart View for Office, administrators can update Planning applications based on either the downloaded template or a modified copy of one of the templates. The modified template must always adhere to template guidelines (described in Working with Artifacts in the Application Template).

When you update an application, you can:

- Define new or modify existing dimensions
- Add or modify a limited amount of data
- Define new or modify existing access permissions
- Define new or modify existing substitution variables
- Modify advanced application setting

Only administrators can update an application.
Before you begin, you should launch Excel and, from Smart View, log in to the applicable data source.

To update an application in Smart View:

1. If you have not already done so, open the modified application template file.
2. In the Smart View Panel, select the application.
3. In the Action Panel, select **Application Management**.
4. In the popup dialog, select **Update Application**.
   The application update status appears in the lower left corner of Excel.
5. When the application update process is complete, access the application, either in the Planning web application or in Smart View, and verify your changes.

**Deleting an Application**

Administrators may delete applications from Oracle Smart View for Office.

Before you begin, you should launch Excel and, from Smart View, log in to the applicable data source.

To delete an application from Smart View:

1. In the Smart View Panel, select the application.
2. In the Action Panel, select **Application Management**.
3. In the popup dialog, select **Delete Application**, and confirm in the subsequent dialog.
4. In the Smart View Panel, verify that the application is removed.
Using Smart View to Import and Edit Application Metadata

Related Topics
- About Smart View
- Planning Admin Extension and Office AutoCorrect
- Importing Planning Dimensions in Smart View
- Editing Members in Smart View
- Adding Application Members in Smart View
- Moving Members in Smart View
- Guidelines for Moving Members in Smart View
- Working with Attribute Dimensions
- Designating Shared Members in Smart View
- Refreshing and Creating Databases in Smart View

About Smart View

Oracle Smart View for Office is the Microsoft Office interface to the application, where end users can perform their planning and forecasting activities. They can analyze application data in Microsoft Outlook, Excel, Word, and PowerPoint. Ad hoc grids in Smart View enable application users to personalize focused data slices that they frequently access and share them with others through Smart View or Planning. See Oracle Smart View for Office User’s Guide and the Focusing Your Analysis with Ad Hoc Grids chapter in Working with Planning.

Administrators can use Smart View to quickly import and edit application metadata.

Note:

All procedures described in this appendix are performed within Smart View using the Planning Admin Extension. See Getting Started with Oracle Enterprise Performance Management Cloud for Administrators for instructions on installing the extension.

Watch this video to learn about entering data in the application using Smart View.

Watch Video
Planning Admin Extension and Office AutoCorrect

In the Planning Admin Extension, Office AutoCorrect can affect member editing with the Planning Admin Extension in Oracle Smart View for Office.

When editing member properties, such as member names, member descriptions, aliases, or member formulas of the data type Text, you may have names that begin with two initial capital letters; for example, DGreen.

Excel will autocorrect this name to one initial capital; for example: Dgreen.

In Excel, you can set an AutoCorrect option to retain two initial capital letters in a name.

To retain two initial capital letters in a name:

1. In Excel, access the Excel Options dialog.
2. Select Proofing, and then click the AutoCorrect Options button.
3. In the AutoCorrect dialog, clear the Correct Two Initial Capitals check box.
4. Click OK to close the AutoCorrect dialog, and then click OK to close the Excel Options dialog.

The change takes effect immediately. There is no need to restart Excel.

Importing Planning Dimensions in Smart View

Importing a Planning dimension into Oracle Smart View for Office allows you to rapidly add, edit, and move the members of the dimension.

- Importing Dimensions in Smart View
- Using Smart View Grids

Importing Dimensions in Smart View

Importing dimensions into the Oracle Smart View for Office grid refers to placing the dimension and its members, and their respective properties, on the grid in preparation for editing.

Note:

You must have administrator privileges to import dimensions in Smart View.

To import a Planning dimension into a Smart View grid:

1. From the Smart View ribbon, click Panel.
2. In the **Smart View Home** panel, click **Shared Connections**, and provide your domain name, user name, and password in the login windows.

3. From the drop-down list in the Smart View Panel, select **Planning**.

4. Expand the **Dimensions** folder to view the application dimensions in the folder.
   - If attribute dimensions are defined, they are displayed in the **Attributes** folder.
   - If a Time Period dimension is defined, it’s displayed in the Dimensions folder. Note that for the Time Period dimension, Data Storage is the only editable property.

5. Right-click a dimension name and select **Edit Dimension**.

**Using Smart View Grids**

The Oracle Smart View for Office grid allows you to rapidly add, edit, and move the members of a Planning dimension.

- **Overview of the Smart View Grid**
- **The Smart View Grid and Ribbon Display**
- **Guidelines for Using the Smart View Grid**
- **Default Metadata Dimension Member Properties**

**Overview of the Smart View Grid**

A Oracle Smart View for Office grid consists of two dimensions:

- A Planning dimension on one axis
- A metadata dimension on another axis

A metadata dimension represents a flat list of metadata members. For each dimension, a predefined set of metadata members is presented by default. Each metadata member corresponds to a specific member property valid for the Planning dimension that was imported onto the grid. Instead of containing numeric values, the metadata grid data cells hold the specific property values. To add metadata dimension member properties that are not on the grid by default, use the **Member Selection** dialog in Smart View.

**The Smart View Grid and Ribbon Display**

The Oracle Smart View for Office grid displays the Planning dimension with member names in rows and member properties in columns. When you first import a dimension for editing, a predefined set of member properties is displayed by default in the grid.

In **Figure 1**, you can see an example of the default member properties for the Vision Account dimension.
Also note in Figure 1 that the Planning Ad Hoc ribbon is displayed. When importing a dimension on to a grid, the ribbon displayed is the same as the Planning Ad Hoc ribbon displayed for a conventional Smart View ad hoc grid, with fewer options enabled.

For all dimensions, including attribute dimensions, these are the valid options in the ribbon:

- Zoom In
- Zoom Out
- Keep Only
- Remove Only
- Member Selection
- Refresh
- Submit Data

The following options may appear enabled, but are not valid for metadata grids: Preserve Format, Cascade, Insert Attributes, POV, and Adjust.

You can add metadata dimension member properties using the Member Selection dialog box, accessed from the ribbon, or accessed by right-clicking a property in the grid, then selecting Smart View, and then Member Selection.

For detailed information on using the Smart View ribbon, see "Ad Hoc Analysis" in Oracle Smart View for Office User's Guide.

For information on using the Member Selection dialog box in Smart View to add metadata member properties to the grid, see "Selecting Members from the Member Selector" in Oracle Smart View for Office User's Guide.

Guidelines for Using the Smart View Grid

The following guidelines will assist you in using the Oracle Smart View for Office grid to edit Planning metadata:

- The following functionality isn't available in Smart View grids with Planning metadata:
  - Pivot
  - Pivot to POV
  - Cell Text
  - Cell Notes
Supporting Details

- Data cell values can be textual or enumeration or numerical.
- The Parent Member is used to specify or modify the parent/child relationship.
- The position of a member in a grid doesn't necessarily represent the actual position of siblings in the outline.
- Each metadata grid must be linked to a corresponding Planning dimension.
- Columns for each Planning dimension are based on the corresponding set of member properties available in the Planning dimension editor.

When you first import a dimension, a predefined set of metadata dimension member properties is displayed. See Default Metadata Dimension Member Properties for a list of those member properties.

- After a metadata grid is opened, it can't be relinked to a different dimension.
- The corresponding valid set of metadata members is specific to each dimension.
- Planning dimension members are valid for corresponding dimensions only.
- For the Time Period dimension, Data Storage is the only editable property.

Default Metadata Dimension Member Properties

When you first import a dimension in the Planning Admin Extension, a predefined set of dimension metadata member properties is displayed by default. Use the Member Selection dialog box to add other metadata member properties to the grid.

The sections below list the default member properties for the Planning dimensions.

For information on using the Member Selection dialog box in Oracle Smart View for Office to add other member properties to the grid, see “Selecting Members from the Member Selector” in Oracle Smart View for Office User’s Guide.

Account Dimension

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
- Account Type
- Variance Reporting
- Time Balance
- Default Alias Table
- Default Formula

Entity Dimension

- Parent Member
- Default Data Storage
- Data Type
- Consol op. for all Cubes
• Base Currency
• Default Alias Table

**Custom Dimensions**
• Parent Member
• Default Data Storage
• Data Type
• Consol op. for all Cubes
• Default Alias Table

**Scenario Dimension**
• Parent Member
• Default Data Storage
• Data Type
• Consol op. for all Cubes
• Start Yr.
• End Yr.
• Start Period
• End Period
• Default Alias Table
• Default Formula

**Version Dimension**
• Parent Member
• Default Data Storage
• Data Type
• Consol op. for all Cubes
• Type
• Default Alias Table
• Default Formula

**Period and Years Dimensions**
• Leave as is or limit to Parent Member
• Default Data Storage – Note that for the Time Period dimension, Data Storage is the only editable property
• Data Type
• Default Alias Table

**Currency Dimension**
• Parent Member
Editing Members in Smart View

The Oracle Smart View for Office grid allows you to rapidly edit the properties of members of a Planning dimension.

To edit member properties in Smart View:

1. Within Smart View, import a Planning dimension into a Smart View grid (see Importing Dimensions in Smart View).
2. Highlight a member property in the grid.
   Note that for the Time Period dimension, Data Storage is the only editable property.
3. In the drop-down menu, select a value.
4. Click Submit Data to save the grid.

Note:
Modified cells are displayed in a different color.
Adding Application Members in Smart View

The Oracle Smart View for Office grid allows you to rapidly add members to a Planning dimension.

- Adding Members in Smart View
- Guidelines for Adding Members in Smart View

Adding Members in Smart View

The Oracle Smart View for Office grid allows you to rapidly add members to a Planning dimension.

- Watch this video to learn about adding Entity dimension members in Smart View.

- Watch this video to learn about adding Account dimension members in Smart View.

- Watch this video to learn about adding Scenario dimension members in Smart View.

To add members in Smart View:

1. Within Smart View, import a Planning dimension into a Smart View grid (see Importing Dimensions in Smart View).

   Note that Smart View does not support adding members to the Time Period dimension.

2. Enter the name of a new member in the name column.

   Note:

   To add a member with a numeric name, place a single quote (') in front of the numeric member name to inform the system that the number is a member name and not data. For example, to add a member named 123, enter ‘123.'
3. Click **Refresh**.
   
   New members are marked by an asterisk (*) in the grid after you perform the Refresh. A default set of properties is automatically applied to the new member. The default Parent Member is the root member of the dimension.

4. **Optional:** To change any property from the default value (the root member), highlight the appropriate cell in the grid, and then from the drop-down menu, select another value. (See *Editing Members in Smart View*.)

   **Note:**
   
   Perform a Refresh *before* you modify any properties in your new member. The Refresh operation will replace any changed values with default values from server.

5. Click **Submit Data** to save the grid.

   **Note:**
   
   - To modify the properties of the new member, see *Editing Members in Smart View*.
   - Member properties, including member names, can be localized based on the particular Locale set by the Smart View application.

### Guidelines for Adding Members in Smart View

- A new member is added as the last sibling under the specified parent.
- Parent-child relationships are determined by the Parent Name column property.
- The relative position of a new member in the grid is insignificant.
- The relative position of a new member in the grid will not be changed after performing the **Submit Data** operation. To see the actual position of a new member in the outline reflected in the ad hoc grid, perform **Zoom Out**, then **Zoom In**.
- A full validity check of a new member name, including a check for invalid characters and a check for duplicate names, is performed during the **Submit Data** operation.
- Default properties are automatically applied by the application to a new member. The specific default properties are based upon those of the dimension.
- When adding new currency members to a Currency dimension in a simplified application, the parent member for the new currency defaults to the "Currency" dimension member. It should default to the "Input Currencies" member. You can correct this by doing the following:
  1. Replace "Currency" with "Input Currencies" in the Parent Member column for each new currency member that you added.
  2. In the Planning Ad Hoc ribbon, click **Submit Data**.
  3. Click **Refresh**.
4. Verify that for each new currency you added, the entry for the Parent Member column is “Input Currencies.”

Moving Members in Smart View

The Oracle Smart View for Office grid allows you to rapidly move members from one parent to another within a dimension.

For detailed information on the use of dimensions and members in Smart View, see "Dimensions and Members" in the Oracle Smart View for Office User’s Guide.

To move a member in Smart View:

1. Within Smart View, import a Planning dimension into a Smart View grid (see Importing Dimensions in Smart View.)
2. Highlight the member in the Parent Member column in the grid.

Note:
An empty Parent Member property causes the value of the Parent Member to default to the root member.

3. Enter a parent name at the intersection of the member name and Parent column.
4. Click Submit Data to save the grid.

Note:
You can save moves of multiple members or subtrees in the same Submit Data operation. If the Submit Data operation fails for one member, the application server stops the operation and doesn’t save changes.

Guidelines for Moving Members in Smart View

- An empty Parent Member in the grid denotes a root member in the dimension.
- The value of a Parent Member follows the rules applicable to the corresponding value in the metadata load file used by Oracle Smart View for Office.

Working with Attribute Dimensions

Related Topics
- Adding Attribute Dimension Members
- Associating Attribute Dimension Members with Dimension Members
Adding Attribute Dimension Members

You create attribute dimensions in the Planning web interface. You can then add members to attribute dimensions using the Planning Admin Extension in Oracle Smart View for Office.

To add members to attribute dimension members:

1. From the Smart View ribbon, click Panel.
2. In the Smart View Panel, click Shared Connections or Private Connections, and provide your domain name, user name, and password in the login windows.
   If you chose Private Connections, select the private connection in the drop-down list of connections, then provide your login credentials.
3. From the Smart View Panel, select EPM Cloud.
4. In the tree in the Smart View Panel, expand the Attribute Dimensions node.
   In the following example, the Attribute Dimensions node contains four attributes, one for each attribute type: text, date, Boolean, and numeric.

5. Select an attribute dimension, and then select Edit Dimension.
   To illustrate, we selected the AttText attribute dimension to edit. An example of the initial grid for the attribute dimension is shown. This dimension has no members yet, and just two properties, Parent member and Default Alias Table.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Parent Member</td>
<td>Default Alias Table</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AttText</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6. Type the attribute member names to add; for example:
7. From the Smart View ribbon, click **Submit**.

The attribute members are submitted to the Planning application.

8. **Optional**: Continue editing the attribute members, entering data for other properties by typing directly in the grid, and then submit your changes.

### Associating Attribute Dimension Members with Dimension Members

1. From the Smart View ribbon, click **Panel**.

2. In the Smart View Panel, click Shared Connections or Private Connections, and provide your domain name, user name, and password in the login windows.

   If you chose Private Connections, select the private connection in the drop-down list of connections, then provide your login credentials.

3. From the Smart View Panel, select EPM Cloud.

4. In the tree in the Smart View Panel, expand the **Dimensions** node, select the dimension to work with.

   In the following example, the Dimensions node shows the standard Planning dimensions along with some custom dimensions such as AltYear and Channel. The Product dimension is selected.
5. Select **Edit Dimension** and note the initial layout of the grid on the sheet. If the attribute dimensions are not displayed on the sheet, then use the Member Selector to add them. In the Planning Ad Hoc ribbon, select Member Selection, and then add the attribute dimensions.

6. In the grid, expand the dimension to view its members.

7. Associate the attribute member values to the dimension members using the cell-based member selector, as shown in Figure 1.

![Figure F-2 Associating Attribute Values to Dimension Members](image)

8. Repeat the previous step for any other dimension member and attribute value associations you want to define.

9. At any time, click **Submit** in the Planning Ad Hoc ribbon to submit your changes to the database.

   You can perform interim submits, or submit all changes at once.
Designating Shared Members in Smart View

The Oracle Smart View for Office grid allows you to rapidly designate members for sharing within a dimension.

Note:
In Smart View, the Shared data storage option isn't available for members of the Currency dimension or any attribute dimensions.

To designate a shared member in Smart View:
1. Verify that the base member exists.
2. Highlight the base member in the Parent Member column in the grid.
3. Change the parent name value for the base member.
4. Highlight the base member in the Data Storage column.
5. In the drop-down menu, select Shared.
6. Click Submit Data to save the grid.

The Submit Data operation will refresh the base member with its original Parent Member and Data Storage properties. The shared member will be added under the specified parent on the server. You must zoom in on the parent to see the newly added shared member.

Note:
The new shared member will not be automatically added to the Smart View grid. The list of members displayed in the grid remains unchanged.

Refreshing and Creating Databases in Smart View

The Oracle Smart View for Office grid allows you to quickly refresh a database or create a new one.

To refresh or create a database in Smart View:
1. From the Smart View ribbon, click Panel.
2. From the Smart View Panel, select Planning.
3. Expand the Dimensions folder to view the application dimensions in the folder.
   If attribute dimensions are defined, they are displayed in the Attributes folder.
   If a Time Period dimension is defined, it’s displayed in the Dimensions folder.
4. Right-click the root Dimension folder and select Refresh Database or Create Database.
The Smart View **Refresh Database** dialog box or **Create Database** dialog box is displayed.

5. Select **Refresh** or **Create**. A progress bar is displayed, indicating the percentage of steps completed for the refresh or create operation.
Best Practices for Designing Your Application

Follow these basic steps to design an application based on best practices that meets your business needs.

1. Best Practices for Getting Started
2. Best Practices for Planning Your Application
3. Best Practices for Applying Your Design

Best Practices for Getting Started

Before you create your application, gather information about current processes. This is a best practice to ensure that the application meets your business requirements.

• Analyze your current processes.
  – Look at what is working in your current processes, and plan how to build that into the application. Consider taking this opportunity to improve current processes.
  – Review your company’s financial statements. Determine the key revenue and expense areas that make up 80% of profit and loss.
  – Focus on the major drivers of your business. For this subset of accounts, understand the business drivers and required dimensions. For example, employee compensation typically accounts for 50-60% or more of expenses.
  – Decide if detailed modeling is required, or if another methodology exists. For each major account in the profit and loss statement, understand how to derive appropriate values for the account. Business logic could include simple calculations like Units x Rates or other methods.
  – For the remaining 20% of the accounts, plan or forecast using some simple logic such as a percentage growth or simple entry.

• Gather any existing spreadsheets. Based on your current process, gather your input sources and understand their purpose. Review the sources to determine the business logic, relationships, and other calculations.
  – Analyze the spreadsheets to understand your current processes. Understand the sources of information and the business logic that they support. Find out if your business process logic applies to all organizations.
  – Find out if your business process logic applies to other business units and divisions. If the process differs, decide if you want to make processes consistent.

• Gather your current reporting requirements. Because reports are used to present financial results for management review and analysis, they are a key consideration during design.
– Understand the business logic, relationships, and calculations for your current processes.
– Find out if your business process logic applies to other business units and divisions. If the process differs, decide if you want to make processes consistent.

• Review your current financial statements. Review reports and consider what type of information is included.
  – Identify the dimensions in reports, for example, if reports are done by cost center, division, account, or customer. Decide if you need to add dimensions, such as product.
  – Review financial statements and examine the intersections. Learn if reports are done by account and entity or by other slices.
  – Understand the business drivers for accounts, including the dimensions involved. Review profit and loss line by line to determine if modeling is involved or if the information is simply added by users.
  – Note the layout of reports, including which elements are on the rows and columns.
  – Determine the number of unique report formats such as by Cost Center or by Division. Specify a report format for each type of report you need.

• Gather planning and forecast requirements. Collect the requirements for producing an annual plan and forecast. Conduct an analysis and understand if all processes are the same.
  – Determine currency requirements. Identify the currencies in which you plan and report, and whether you plan or forecast in multiple currencies. Verify foreign currency requirements. Confirm if you perform any currency sensitivity analysis.
  – Determine the planning horizon. Know how many years in the future you plan, such as one, two, three, or five years into the future. Plan how much historical data to bring into the application. It’s typical to include one or two years of historical data.
  – Identify the planning and forecasting process. Know whether you prepare annual budgets and forecasts. Understand how often you revisit your forecast, such as monthly or quarterly.

Determin the planning processes you want to support, such as Plan or Forecast. Establish if planning and forecasting processes are similar. Know whether you want to prepare a rolling forecast.

Decide if you’ll plan toward a target. If you set high level targets for plan or forecast, decide on the level of accounts to target.

Determine if multiple iterations are required. Establish if you need to store multiple submissions of Plan or Forecast. Determine how many forecasts you maintain. Decide if you want to do comparisons, such as comparing forecasts.

Additional questions to help you plan the process:
* What are the expense drivers for your business, such as compensation, material, travel, or capital and equipment expenses?
* What are the revenue drivers for your business, such as products, projects, or services?
* If Product is one of the main revenue drivers, how many products do you have?
* How many business units or entities do you have? What main metrics and KPIs are tracked? How many users are involved in financial planning? Who are the main users?
* Determine the approval process that should be incorporated into the application. Is the approval process the same for Plan and Forecast?

  **Understand your data sources.** Know the file format and the contents of each data source. This helps you plan requirements for dimension and data integrations.
* Determine the source of dimensions such as Account, Entity, and custom dimensions.
* Determine the source and frequency of Actual results.

  **Determine your approach going forward.** Identify process improvements, and note which ones to incorporate in your application. Plan the strengths to incorporate going forward, and identify any weakness and areas to improve in the future.

### Best Practices for Planning Your Application

An important best practice is to establish the goals, key objectives, and scope of your application. Implementation is often broken into phases to keep objectives in focus. After you know the requirements, you can move forward with designing your application.

The information you gathered earlier can be included in your application as follows:

- **Dimensions**
- **Forms**
- **Reports**
- **Calculations**
- **Users**

  **Design dimensions.** Identify the dimensions required to support your process. Your application comes with these dimensions: Account, Entity, Version, Scenario, Years, Period, and Currency, where applicable.

  - Identify the members to be included in the application and the source of those dimensions. Understand the size of each dimension.

  - It's a best practice to document calculations. For each piece of the calculation, document how the result is determined and the source of the number. Confirm if the source is data entry or data fed from another system.

  **Design calculations.** Review your company's profit and loss statement line by line, and confirm how to plan or forecast the account. For accounts that require calculations, you can leverage Calculation Manager to build the logic.

  Understand calculations, and ask any necessary questions. For base accounts, confirm how each account is planned. Sample questions:

  - Does this calculation apply for all entities?

  - Are there any other dimensional aspects to this calculation?
− Is the calculation by product or customer?
− Is the calculation the same for Plan and Forecast?

• **Determine revenue calculations.** It’s a best practice to establish the requirements for revenue and its drivers.
  − Know the business drivers you use, and understand if revenue is driven by product or services. Decide if you want to derive revenue in your model. Identify any other information to capture.
  − Establish the revenue drivers. Determine if other dimensions are required.
  − Decide on the logic for revenue planning. Identify the calculations you want to build to support your process, for example, Units x Price. Plan the logic behind your revenue calculations.
  − Determine the forms and the layout to collect information and input from end users.

• **Determine expense requirements.** Review your expense accounts to identify the key areas of your business.
  Focus on the areas that make up the majority of your expense to identify the major drivers. You can plan the remaining minor expenses in a very simple and straightforward way, such as by using trends or by simply entering a value.

• **Determine expense drivers.** Focus on this group of accounts, and determine the business drivers of the accounts and the required dimensions.
  Review your expense accounts to identify the key areas of your business. Focus on the areas that make up the majority of your expense to identify the major drivers. You can plan the remaining minor expenses in a very simple and straightforward way, such as by using trends or by simply entering a value.
  − Review employee compensation. Confirm how employee compensation will be planned, such as by employee, job, or grade.
  − Know which accounts are included in planning, and understand the sources of data to support compensation. Identify the type of reporting you do on compensation.
  − Assess other expense account requirements. Plan the calculations you need to build to support your process. Know the logic behind your expense calculations.
  − Document calculations. For each piece of the calculation, document how the result is determined and the source of the number. Confirm if the source is data entry or data fed from another system.
  − Determine the forms and the layout to collect information and inputs from the end users.

**Determine the approval process.** Determine the dimensions that drive the approval process for Plan and Forecast. The base dimension of approval process is Entity, Scenario, and Version. Decide if you want to include another dimension, such as Product, Project, or Market.

Document the approval process the application should support.
Best Practices for Applying Your Design

After reviewing the design guidelines and best practices in this section, you're on your way to designing a successful application that meets your business needs. You can now quickly and easily set up your application.

Use the information that you gathered to set up your application. The application setup wizard builds the foundation of your application based on your answers to simple questions.

As an example, here is how some of the information you gathered correlates to your application.

Table G-1  Applying Your Design to the Application

<table>
<thead>
<tr>
<th>Research on Requirements</th>
<th>Application Wizard Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of historical years to include</td>
<td>The Start Year for the application</td>
</tr>
<tr>
<td>Your planning horizon</td>
<td>The number of years chosen for the application</td>
</tr>
<tr>
<td>The planning processes to support</td>
<td>The scenarios included in the application. You can either add scenarios in the application wizard, or you can add them later.</td>
</tr>
<tr>
<td>Whether you need to store multiple submissions of Plan or Forecast</td>
<td>The number of submissions correlates to versions in the application. You can add versions in the application wizard, or you can add them later.</td>
</tr>
<tr>
<td>Whether you plan in multiple currencies</td>
<td>If you plan in multiple currencies, answer Yes in the application wizard. You'll add currencies later. Select your reporting currency in the wizard.</td>
</tr>
<tr>
<td>Time periods</td>
<td>If you require weekly distribution such as 445, 454, or 544, select the base time period as 12 Months, and then select an appropriate Weekly Distribution option. Even monthly distribution is the default.</td>
</tr>
</tbody>
</table>

Best Practices for the Design Walkthrough

Use these best practices to do a design walkthrough that will help you build and roll out your application.

Build Your Application

Start by building the foundation—your company’s accounts and organizational structure. Next, add scenarios to support your internal processes, such as Plan, Actual, and Forecast. Add the variance members you report, such as Actual vs. Plan.

Create forms that will be used to collect data from your users and to perform reviews, analysis, and reporting. To support your business logic, you can leverage Calculation Manager to build your calculations. You can also create reports and apply access permissions before rolling out your application to users.
Create the Application Structure

Add accounts, entities, and other dimensions to support your business process.

Dimensions categorize data values. Planning includes these dimensions: Account, Entity, Scenario, Version, Period, and Years. If you plan in multiple currencies, your application also has a Currency dimension.

You can use the Custom dimension to define your own values, such as Product, Customer, or Market. You can create up to 13 user-defined custom dimensions. However, the best practice recommendation is to include fewer than 12. Dimensions can be added using a load file or built in Oracle Smart View for Office.

Watch this tutorial video to learn how to export and import data in the application.

This tutorial shows you how to load dimensions using a file. View tutorial

About the Entity Dimension

The Entity dimension represents your organizational structure, such as Cost Centers, Departments, Business Units, Divisions, and so on.

You can group Cost Centers by creating rollup members, called parents, to reflect how your organization is viewed. For example, rollups can be by business unit, division, or other functional structure. As an example, you could create Cost Centers that roll up to Business Units that roll up to Divisions.

You can create multiple reporting structures. For example, an alternate structure could be created to support regional reporting. If you plan in multiple currencies, the base currency of each entity should be set.

The Entity dimension is one of the primary dimensions used for the budgeting process. Together with the Scenario and Version dimensions, the Entity dimension is used to define a planning unit, a discrete component that can be promoted or demoted for approval or review by a planner's peers.

Members of all dimensions outside the planning unit will be promoted and demoted along with the planning unit itself. For example, all twelve months are promoted together when a planning unit is promoted. Individual months can't be promoted independently.

For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.

After each dimension is loaded or updated, it's a best practice to refresh the application.

About the Account Dimension

The Account dimension is the place for your chart of accounts. It should include the members to which you plan or forecast. It doesn't necessarily include every account in your chart.
For example, your Account dimension could include accounts for Income Statement, Balance Sheet, and Cash Flow. Or, it could include accounts for KPIs and Ratios. In some cases, your accounts may have sub accounts, but this isn't typical.

The Account dimension includes financial intelligence. The following account types are supported:

- **Expense**—Cost of doing business
- **Revenue**—Source of income
- **Asset**—Company resources
- **Liability and Equity**—Residual interest or obligation to creditors
- **Saved assumption**—Centralized planning assumptions ensuring consistency across the application

The account type settings are used to report Quarterly and Year Total values and for variance analysis.

Planning uses a hierarchical structure to create Account grouping subtotals and totals. Each account group is assigned a consolidation operator that determines how it rolls up to its parent.

Example: Net Income = Total Revenues - Total Expenses

In this example, the consolidation operator for Total Revenues is Addition, and the consolidation operator for Total Expenses is Minus.

The Account dimension can be populated either by loading data or using Smart View. To load data from a file, the file format must meet specific requirements.

For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.

After each dimension is loaded or updated, it's a best practice to refresh the application.

Best practices:

- Upper level members should be set to Dynamic Calc.
- For member formulas used to calculate Ratios and other types of KPIs or percentages, set them to Dynamic Calc, Two Pass. The Two Pass setting properly calculates Percentages at upper levels.

**About the Version Dimension**

You can use versions to preserve different iterations of the planning process. They are also useful for controlling data access to Read or Write.

These two types of versions are available:

- **Standard Target**—Input data can be entered to upper levels.
- **Standard Bottom Up**—Input data can be entered to level 0 only.

Approvals and workflow functionality can be enabled only for Bottom Up versions.

As a best practice, the following versions are recommended:

- **Working**—Where users perform their tasks, including reviewing Actual Results and developing Plan and Forecast.
• 1st Pass—If you want to maintain multiple iterations of your Plan, you can pre-
serve a pass of it in this version. You can create other members if you require
more than one saved iteration. You can leverage the Copy Data functionality to
move data to this version. Copy data copies data and textual input.

• What If—Provides a placeholder where users can change assumptions and ana-
lyze the outcome.

For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.

After each dimension is loaded or updated in the build process, it's a best practice to refresh the application.

About the Currency Dimension

If you enabled multiple currencies for your application, you can add the currencies you use to plan and report.

You can then define exchange rates by scenario and year to be used in conversions. A calculation script is created that enables you to perform currency conversion. To enter exchange rates, click or tap Plans and open the form "Exchange Rates to Primary Reporting Currency".

Best practices:

• Limit the number of reporting currencies. Typically, customers have only one. If you have more, see Setting Up Currencies for more information.

• Enter exchange rates for each valid scenario and year combination.

• From this point on, currency conversion can be calculated by running the Calculate Currencies business rule that is associated by default with each form.

• An account's exchange rate type is modified, such as from Ending to Average.

Run the currency conversion calc script prior to:

• Reviewing any updated local data in reporting currencies

• Running certain calculations that may be dependent on reporting currency data

About Exchange Rates

Each application has a default currency that you specify when creating the application. When you set up exchange rate tables, you enter exchange rates from all source currencies to the default. Triangulation is used to convert to all other Reporting currencies.

Exchange rates are set by Scenario by year for Average and Ending Rates. For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.

About the Period Dimension

Use the Period dimension to establish the calendar's range within a given year, for example, by month.

Best practices:
Use substitution variables for this dimension to support reporting and calculations. Potential substitution variables are: ‘CurrMo’, ‘CurrQtr’, ‘PriorMo’. These variables must be updated on a monthly basis.

To use time period calculations such as Year to Date (Y-T-D) or Quarter to Date, select the dynamic time series icon in the Period dimension. You can then select which time period calculations you need to support your process.

Summary time periods such as quarter totals and a year total should be set to dynamic calculate to reduce calculation time.

For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.

After each dimension is loaded or updated, it's considered a best practice to refresh the application.

About Years

Years are incorporated into the application in many places, including forms, calculations, reports, and Smart View. Because you'll use the application for many years into the future, the best practice to referencing this dimension is by using a substitution variable.

Substitution variables act as global placeholders for information that changes regularly. The variable and value correspond to the year, and the value can be changed at any time.

The value of the substitution variable is displayed on forms and reports as a placeholder. This reduces maintenance for the application. Set substitution variables by going to Administration, then Manage, then Variables.

For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.

As a best practice, create substitution variables for each year that is included in your process. For example:

Substitution Variable, Description
CurrY, Current Year
NextYr, Budget (Plan) Year
PriorYr, Prior Year

About Custom Dimensions

You can use a custom dimension to further categorize your data. For example, custom dimensions might include Product or Markets.

Access permissions can't be granted at the dimension level, also called generation one. For example, access permissions can't be assigned directly to the Product member for all descendants. If you enable security for your Custom dimension, it's recommended that you design generation two for all Custom dimensions to which security will be applied with security access assignments in mind.

For detailed information on adding dimensions and members using a load file, see Importing and Exporting Data and Metadata.
After each dimension is loaded or updated, it's a best practice to refresh the application.

**Refresh the Application**

You must refresh the application whenever you change the application structure.

Changes made to the application are not reflected to users performing data entry and approvals tasks until the application is refreshed.

For example, when you modify properties of an Entity member, add a Scenario, or change access permissions, these changes are reflected to users after you refresh the application.

**Load Historical Data**

After you load all of your structures, such as accounts and entities, you can load historical data. This can include data from prior year actual results and current year plan and budget.

Loading historical data provides planners a way to analyze results, review trends, and make meaningful comparisons.

It also helps verify the structures you have built into your application. For example, you can verify that data ties to previously created reports. If the data doesn't reconcile, you must verify if this is caused by a true data issue or if there is an issue with the structures.

An aggregation rule will need to be created to see consolidated data in your application. See [Aggregation Options](#) to learn how to create an aggregation rule.

**About Valid Intersections**

Valid intersections let administrators define rules, called valid intersection rules, that filter dimensional intersections for planners when they enter data or select runtime prompts. For example, you can specify that certain programs are valid only for specific departments. Leverage valid intersections to control data entry only to valid intersections.

For form design, if the dimensions that are set in the valid intersection are found on the Page, the user will be presented only with valid combinations in the member selector. If the dimensions that are set with valid intersections are found on the column or row, the form designer can suppress invalid intersections completely. When the suppression option isn't selected, invalid intersections are set to read only.

To learn more, see [Defining Valid Intersections](#).

**About Forms**

You'll build a number of forms to support data entry and summary-level reports. The form content is similar to the templates you use to collect and calculate data. The layout may differ from what you're currently accustomed to in spreadsheets.

You group forms within major categories such as Revenue, Compensation Expense, Other Expenses, and so on. You can create some forms to support data entry, and others for summary and review. You can also include charts to help planners analyze results.
Planners can enter text and data. They can also enter supporting detail by selecting an appropriate intersection in the form, and then clicking “Supporting Detail” to open a new input form that allows entry of additional detail for that intersection.

Form performance is based on several factors, including network and environmental factors, structure, layout, and so on.

Best practices:

- Put dense dimensions such as Account and Period on the row and column of a form. Put sparse dimensions such as Entity on the Page axis.
- Dimensions such as Scenario or Version and Year can reside on the POV, column, or row. It's important to properly gauge how columns or rows will be returned when a planner opens the form.

**Build Entry Forms**

Create forms to enable planners to enter information such as revenue, expenses, and assumptions.

Best practices:

- Group accounts logically, but don't include too many accounts on a single form.
- Limit the number of entry forms to an amount comfortable for end users. A delicate balance needs to be achieved between the number of accounts on a single form and the number of forms required to support your process.
- Consider using composite forms as a way to limit the number of forms while still controlling the number of rows.
- Use detail forms to enable planners to enter all related information. All accounts that require input should be found on a form. The accounts can be broken down into several different forms.
- While building forms, ensure that you select all appropriate options to enhance the design of your form. For example, use settings to control precision, display, and menus, and to associate the proper rules with your form.
- Use Substitution Variables to reference dimensions such as Years.
- Suppress invalid Scenario/Time Period option, set Periods in row or column on the form to the Start and End Period set for the Scenario. Leveraging this functionality can be used instead of substitution variables for Years.
- Consider setting valid intersections to set relationships between different dimensions. Suppressing invalid combinations can be set in row or column to make only valid intersections available to end users. By default, only valid intersections will be available to the end users when the dimensions are set in the Page selection.
- Use relationships to incorporate members onto the forms instead of picking members individually.
- Consider using User Variables for dimensions such as Entity and Scenario to help reduce the dimension selection for end users.
- If your application supports multiple currencies, consider setting a User variable so users can define their base currency.
- Organize forms into folders.
- Substitution Variables reduce the maintenance on forms.
Build Detailed Revenue and Expense Forms

Detail forms should allow planners to enter all revenue- and expense-related information. All accounts that require input should be found on a form.

Best practices:

- Group accounts logically, but don't include too many accounts on a single form.
- Limit the number of entry forms to an amount comfortable for end users. A delicate balance needs to be achieved between the number of accounts on a single form and the number of forms required to support your process.
- Consider using composite forms as a way to limit the number of forms while still controlling the number of rows.
- Use detail forms to enable planners to enter all revenue related information. All accounts that require input should be found on a form. The accounts can be broken down into several different forms.
- While building forms, ensure that you select all appropriate options to enhance the design of your form. For example, use settings to control precision, display, and menus, and to associate the proper rules with your form.
- Building forms can be iterative to support planners and processes.

Associate Rules with Forms

Associating rules with forms enables users with appropriate access to launch associated business rules from the form to calculate and derive values.

You can associate multiple business rules with a form by cube.

Business rules associated with a form can be set to launch automatically when the form is opened or saved. You can select Use Members on Form to populate runtime prompts from the current form instead of prompting users for input when rules are launched.

Best practices:

- For rules that take longer to run, set them to launch from an Action menu or simply through association with the form.
- If a business rule has runtime prompts, limit the number of prompts to keep the planner's job simple.

Add Menus to Forms

You can associate menus with forms. Action menus allow planners to click rows or columns in forms and select menu items to:

- Launch a business rule, with or without runtime prompts
- Move to another form
- Move to Manage Approvals with a predefined scenario and version

Menus are context-sensitive. The menus that display depend on the form settings and where planners right-click in the form.

Best practices:
• When designing forms, use Other Options to select menus available for Form menu item types.
• As you update applications, update the appropriate menus. For example, if you delete a business rule referenced by a menu, remove it from the menu.

Build Data Validation Forms

Data validation can serve as a visual clue to planners that business policies have been met. You can add conditional color coding to forms, and validation messages can be generated if entered data violates validation rules or if a condition is met.

Defining data validation rules involves these main tasks:
• Identify the data cells or location that you want to display with validation messages or in different colors when conditions are met.
• Identify the cell, column, or row that needs to participate during rule evaluation, and define the rule accordingly.
• Create the data validation rule at the location identified.

Create Composite Forms

Composite forms allow you to display several forms simultaneously. You can display forms as charts. You can also display multiple forms in a tabular format, similar to what your users are accustomed to seeing in spreadsheets.

These features help you create whatever composite form layout is best for your application.
• Each area in the composite form is called a section. Initially, you specify whether to divide the composite form layout into two side-by-side sections, or into two sections that are stacked one above the other.
• There is also a custom layout option. Each section in a composite form is associated with properties set during creation. You can edit these properties after creating a composite form.
• The composite form point of view and page dimensions can be specified within a composite form. You can set the composite setting to combine dimensions.
• You can design composite forms that have one master composite form and multiple simple forms. When you do so, the selection of members in the master form automatically filters to the members in the simple forms. The simple forms show only the details that are relevant to the members highlighted in the master form. Leveraging master composite forms is a useful way to view both summary and detail level data within a single form.

Organize Forms into Folders

Use folders as a way to organize the forms in your application. Forms can be grouped in folders by process or user type, or simply to help users readily find forms. You can move forms into folders, and you can create a folder hierarchy. Creating folders also simplifies assigning access because all forms in the folder will inherit the access permissions assigned.

Create Dashboards

Dashboards allow you to display information graphically or to display several forms simultaneously. You can also design interactive multi-chart dashboards to enable users
to analyze their plan or forecast data. As another option, you can display a grid and a
graph together, or you can combine multiple grids.

To create a dashboard:

- Drag and drop forms into the dashboard. Using the settings wheel to select the
  chart type desired for each grid.
- You can drag and drop as many forms as you would like, setting the size of the
display by setting either the height or width for the component.
- Set dashboard settings to combine dimensions into a common POV.
- As a best practice, balance the number of components on the dashboard to en‐
sure that it's visually pleasing to the user.

Build Summary Level Forms

Summary level forms typically bring together all of the pieces of a user's plan or fore‐
cast. They enable planners to review and analyze their results.

Leveraging composite forms is a useful way to pull together summary level results and
still have the ability to drill into detail. Using charts can also be an effective way to help
planners analyze their results.

Using dashboards can also be an effective way to help planners analyze their results.

Build Financial Statements

Financial statements allow users to analyze performance and verify their assumptions.
Financial statements could include Income Statement, Balance Sheet, and Cash Flow.

Typically, financial statements include comparative information so users can analyze
their variances. Summary level information is typically built into financial statements
with the ability to view detailed data either through master detail composite forms or by
linking detailed forms using menus.

Incorporate Business Logic

To incorporate your business logic into your application, calculations can be built using
Calculation Manager. This enables you to create, validate, deploy, and administer so‐
plicated calculations that solve business problems.

You typically create business rules and rulesets to:

- Perform revenue modeling
- Perform expense modeling
- Calculate KPIs
- Perform allocations

Calculation Manager includes these objects:

- Rules—Contain components and templates
- Components—Assist you in building rules
- Rulesets—Contain rules that can be calculated simultaneously or sequentially

To learn more about creating calculations, see the Calculation Manager documenta‐
Build Aggregations

Aggregations roll up your application to summary-level members in the dimension, such as Entity or any other sparse dimension.

Calculation Manager includes templates to help you build aggregations. The System Template Aggregation has several tabs. Here are some suggestions on how to use templates.

Set the Point of View

When the point of view is set, the rule will run only for the members chosen. Using a runtime prompt for the dimensions will allow users to specify member values for these dimensions when launching the rule. This way, users can launch the rule several times for different years, scenarios and versions without having to modify the rule in Calculation Manager.

Full dense aggregation

Complete this section if parent values in your dense dimensions are not set to dynamic calc. Typically this tab is left empty.

Full sparse aggregation

Select the sparse dimension that needs to be aggregated. The order of the selected dimensions isn't relevant.

Partial dimension aggregation—dense

Complete this section if parent values in your dense dimensions are not set to dynamic calc. Typically this tab is left empty.

Recommended settings:

Aggregate the data up to the local currency—No
Aggregate the missing values in the Database—Yes
Optimize the Calculation on Sparse dimension—Off
Select a value for the calculator cache—Default

Do you want to activate the debug mode for this wizard?—Debug Wizard On or Debug Wizard Off. Select Debug Wizard On if you want to see a script generated to display selections for some of the Design Time Prompts in this template.

Best practices:

• Leverage runtime prompts for members such as Entity, Scenario, and Version. This allows your rule to be dynamic and run based on user input.
• Typically, dense dimensions such as Account and Period don't need to be aggregated. If this is the case, you can set parent members to dynamic calc. However, if you have member formulas on dense dimensions and they are not set to dynamic calc, a Calc Dim rule is required.

Build Detailed Calculations

You use Calculation Manager to create, validate, deploy, and administer sophisticated calculations that solve business problems.
There are three types of objects that can be calculated in Calculation Manager:

- **Rulesets**—Contain rules that can be calculated simultaneously or sequentially (See Administering Rules.)
- **Rules**—Contain components and templates (See Administering Rules.)
- **Components**—Contain formula components, script components, condition components, range components, and fixed loop components (See Administering Rules.)

Best practices:

- As a first step in building your rules, ensure that you understand the business logic and which entities or departments the rule applies to. For example, know the accounts that are involved in the rule.
- Be sure you know the source and destination accounts.
- After you fully understand the drivers of the calculation, use the proper object component or template to build the rule. The components and templates facilitate member selection to help deploy the rules.

Leveraging runtime prompts for members such as Entity, Scenario, and Version allows your rule to be dynamic and run based on user input.

**Build Reports**

Building reports allows you to report on your financials for management. In this step, you build your Income Statement and other detailed reporting with the proper formatting that your management team is accustomed to reviewing.

Report formats specify the layout of the report, such as which elements are on the rows and columns. Report formats can be used to create many different reports, such as by Cost Center or by Division.

Best practices:

- Before building reports, determine how many different report formats are required.
- To simplify building reports, specify a report format for each type of report you need.
- Begin building reports by arranging dimensions properly. Next, get the report to capture the data. Finally, apply formatting.

**Build Task Lists**

Task lists guide users through the planning process by listing tasks, instructions, and due dates. Task lists help flow users through the application to ensure that the process is followed and that all of the proper data has been collected.

Task lists should be developed to support the different types of users and process flows. Tasks can help users perform many types of tasks, such as:

- Open a form
- Launch a business rule that you specify
- Start the review process with a specified scenario and version
- Copy a version of the current form's data
- Open a specified URL
Tasks guide users through the planning process. Tasks can help users perform many types of tasks. They help flow users through the application to ensure that the process is followed and that all of the proper data has been collected.

Build tasks to support the different types of users and process flows.

Set Up the Navigation Flow

Navigation Flows set the clusters or cards available at the top of the user screen. The cards are typically associated with actions in your business process, such as Plan Revenue and Plan Expenses. Within each card, vertical tabs can be created to lead the user through the process for that business area. Forms can be linked to a vertical tab to guide the user in the process. Vertical tabs can have one or many horizontal tabs linking to either forms or dashboards.

Your application comes with a default navigation flow. To customize the cards and flow for your organization, copy the default, and then you can use it to make your own.

Tap or click Settings, then Navigation Flow. Tap or click Action, then Create Copy.

You can create a cluster to represent an entire business process that can contain cards for the actions, or you can simply create new cards. Cards can be designed to be single page or can have multiple tabs. For a card that is set up to be tabular, you can have multiple tabs that allow you to have content visible to the end user as horizontal tabs. You specify the content type for each of the tabs and link to an artifact.

For example, you can associate cards with:

- Dashboards
- Forms
- Rules
- Approvals

Set Up Access Permissions

Access permissions determine a user's privileges after the product launches. Most often, groups are established to help organize users. By definition, a user group is a set of users with similar access permissions.

Access permissions for groups and individual users can be assigned to these application elements:

- Scenarios
- Versions
- Accounts
- Entities
- Custom dimension members
- Forms
- Business rules

Users can be in a group:

- Service Administrator
- Power User
• Planner
• Viewer

Best practices:

• For dimensions secured by default, modify access permissions as needed.
• Assign access permissions to application elements such as dimension members, forms, and rules. Users can view or use only those application elements to which they have access.

About Users and Groups

Your company’s users must be added to the Oracle Identity Management System prior to gaining access permissions to any of the elements in your application. Access permissions determine a user’s privileges after the product launches.

By definition, a user group is a set of users with similar access permissions. The use of groups as a way to organize users and assign access permissions is a best practice.

Add Users

Users must be added to your environment, assigned privileges, and granted access to the application.

Users’ roles will be defined as one of these types:

• Service Administrator—Creates and manages applications, including dimensions, forms, Calculations, and so on. The Service Administrator manages access permissions and initiates the budget process
• Power User—Creates and maintains forms, Smart View worksheets, and Financial Reporting reports. Can perform all Planner tasks.
• Planner—Enters and submits plans for approval, runs business rules, uses reports that others have created, and views and uses task lists. Leverages Smart View to enter data and do ad hoc analysis.
• Viewer—Views and analyzes data through data forms and any data access tools for which they are licensed. Viewers can’t modify any data in the application. Typical View users are executives who want to see business plans during and at the end of the budget process.

Planners, Power Users, and Viewers can access forms, task lists, and business rules based on permissions assigned by the Service Administrator.

Create Groups

It’s highly recommended to leverage groups when assigning access permissions to users. Having groups of similar users eases security maintenance on an on-going basis. As users are added to groups, they inherit the access permissions of the group. Assigning group access permissions to elements such as dimension members, forms, and task lists means that you don’t need to assign those access permissions individually for each user.

Best practices:

• If an individual user is assigned to a group, and the access permissions of the individual user conflict with those of the group, the individual user’s access permissions take precedence.
The use of groups for sets of users with similar access permissions should be well defined prior to implementing user access.

Individual permissions override group permissions.

If an individual is assigned to multiple groups, the group with the highest access permission takes precedence.

Access permissions assigned to a user directly override access permissions inherited from groups that the user belongs to. For example, if you have inherited read access to Plan from a group but are assigned write access to Plan directly, you get write access to Plan.

Assign Users to Groups
As a best practice, leverage groups as a way to reduce maintenance and assign similar access to users. Give users access to the appropriate groups.

Assign Access to Dimensions
In order for users to read or write data, access permissions must be assigned to the following dimensions:

- Account
- Entity
- Scenario
- Version

If security is enabled on custom dimensions, you must assign security to users to those dimensions as well. For dimensions secured by default, modify security access as needed.

Assign Access to the Account Dimension
Give users read or write access only to those accounts they are allowed to see. You can assign access privileges as Read, Write, or None.

Best practices:

- Relationship functions should also be leveraged whenever possible to reduce ongoing security maintenance. The relationship functions are: Member, Children, iChildren, Descendant, and iDescendant. For example, assigning Write access to Descendants of Net Income for a group allows all users of that group to have Write access to all accounts that are descendants of Net Income. This way, you don't need to assign access individually to each account.

- To take full advantage of the rules of precedence and inheritance, use an exception-based method for managing security. Primary assignment of security should be by group and relationship. Assign group rights to parent level members, and use relationships to push the assignments down to the children or descendants. Assign individual user rights to children on an exception basis.

Assign Access to the Entity Dimension
Give users read or write access only to those entities they are allowed to see. You can assign access privileges as Read, Write, or None.
Assign Access to the Scenario Dimension

Access to Scenario is typically set to Read or Write. For example, you may want to assign access to Actual and Variance scenarios as Read, and to Plan and Forecast as Write.

Assign Access to the Version Dimension

Access to Version is typically set to Read or Write. For example, you may want to assign access to Final Version as Read, and to Working as Write.

Assign Access to Custom Dimensions

If security is enabled on any custom dimension, you must assign security to the dimension in order for users to have access.

Assign Access to Forms

Before users can open forms, they must be assigned access permissions.

Planners who are assigned access to a form folder can access the forms in that folder unless they are assigned more specific access.

Planners and interactive users can view or enter data only into forms to which they have access. They can work only with members to which they have access.

Tips:

• To simplify assigning access to forms, organize forms into folders and assign access at the folder level instead of the individual form level. Access permissions can be set to Read, Write or None.

• When you assign access to a folder, all folders under it inherit that access.

• If you assign specific access (such as None or Write) to a form folder, that access permission takes precedence over its parent folder’s access permissions. For example, if a user has Write access to Folder1 that contains Folder2 to which the user has None access, the user can open Folder1, but doesn't see Folder2.

• If a user has None access to a form folder called Folder1 that contains a form called Form1 to which the user has Write access, the user can see Folder1 and Form1.

Assign Access to Business Rules

Before users can launch business rules, they must be given access permissions to the rules.

As a best practice, organize business rules into folders that have similar user access, and apply security to the folders. You can also give access permissions to individual business rules, although this is a little more time-consuming.

Planners have Launch access to Calculation Manager business rules in folders to which they are assigned access, unless they are assigned more specific access.

Assign Access to Task Lists

In order to navigate through the application, users must be assigned access to individual task lists.
As a best practice, assign access using groups. This is more efficient than applying access to individual task lists.

Assign Access to Reports

Users must be assigned access to a report before they can use it.

As with other artifacts, it's recommended that you organize reports into folders and assign access at the folder level. This limits the amount of maintenance required for security. As reports are added to the folder, access is inherited from the folder.

Build Approvals

Use approvals to track budgets and review status, planning unit ownership, and process issues. This reduces the time required for the planning cycle.

Set up the approval path independent of organizational structure to reflect the path a plan or forecast must follow for approval.

Users can provide annotations and comments for their submissions.

Set up the Planning Unit Hierarchy

Setting the planning unit hierarchy defines the promotional path used in approvals. The basis of the planning unit hierarchy is the Entity or any part of the Entity dimension in combination with any secondary dimension.

The secondary dimension can be a mix between several dimensions, depending on where you are in the workflow. For example, you can combine the Entity dimension with the Products dimension in the promotional path for certain entities, while using the Channels dimension in the promotional path for other entities.

Owners and reviewers can be directly assigned the planning unit. Validation rules can be created to handle conditional promotion path dependent on data conditionals. You create different planning unit hierarchies to support review processes within your organization.

The planning unit hierarchy is then assigned to the appropriate Scenario and Version combination.

Planning units are combinations of scenario, version, and entity or part of an entity. Scenarios and versions are the basis of the review cycle. A planning unit hierarchy contains planning units and any other dimensions that are part of the review process.

Things to know about approvals:

- The review process follows the promotional path you set up when you select the owner and reviewers for a planning unit, unless an event triggers a change in the promotional path.
- Parent/child relationships between planning unit hierarchy members affect the review process.
- When users promote or reject a parent, the parent's children are promoted or rejected unless they are approved. The owner for the parent becomes the owner of the children.
- When users approve a parent, its children are approved.
- After all children are promoted to the same owner, the parent is promoted to the owner.
• When the status of all children changes to one status, for example Signed Off, parent status changes to the same status.

• Users can't change the status of a parent if its children have different owners.

• If the children are promoted to, submitted to, or signed off by different users, the parent has no owner and only administrators can change its status.

• The planning unit moves from one reviewer to another until the budget process is complete.

Test

Testing is a critical step in application development. All of the calculations, access permissions, and reports must be tested to ensure that they work appropriately.

About Unit Testing

Unit testing is the first step of formalized testing, and is the main building block of the test environment. Unit testing involves testing each functional area of the application as a separate unit to ensure that it performs as expected.

For example, a test could confirm that a data load executes to completion without errors. Other tests could confirm that forms and reports are accessible, calculations complete, and so on.

The person that builds or configures the application usually conducts unit testing.

About System Testing

System testing validates that the system operates without error and provides the required functionality.

The main emphasis is to test the way in which the application has been configured and to look at how the team has constructed the business processes and reports. System testing focuses on testing the entire system, including unique parameter configuration, all functions that will be used, and any enhancements.

System testing also looks beyond the software, and validates the effectiveness of manual procedures, forms and controls. It’s a complete set of formal functional tests covering all aspects of functionality within the system being built.

This type of test is often combined with:

• Security Tests—Tests that the system security and database security is appropriate for the overall system and each specific user.

• Integration Tests—Tests the overall business solution, including the passage of data to and from other integrated systems. This confirms that the functionality remains valid when all aspects of the system have been combined.

• User Acceptance Tests—Users validate that the system operates correctly and meets requirements. If users are not involved in formal system testing or they request specific tests, there may be a need for further acceptance tests. However, in most cases, this type of testing is done as part of System and Integration Tests, provided that users recognize these tests as adequate for acceptance purposes.
Rollout

During rollout, you can train end users on the system, and show them how to navigate and use functionality. As a best practice, document your system to enable someone else to take over administration if necessary.

Training

All users of the system should be trained on the application. Users need to learn how to navigate comfortably around the application and understand the tasks assigned to them. Training should include logging into the application, navigating through task lists, entering data, running rules, using Smart View, and using tools within the application. Training is typically the user’s first exposure to the application, and a well planned and executed training session helps make a good first impression.

Document System and Administrative Information

After building your application, it’s recommended that you create system and administrative documentation for the application.

Best practices:

- Create this documentation at the end of the Build Process when the information is fresh.
- Include information such as the sources of data, the application structure, how calculations work, and what maintenance is required for the application.

List maintenance tasks broken down into timeframes, such as monthly and annual maintenance. This makes it possible for someone else to take over the system later if necessary.

Enable the Application for Users

To enable the application for end users, you must open up the system enabling. In addition, start planning units to enable approvals.

Start Planning Units

The planning unit must be started before users can access the system to begin the review process. After it's started, the planning unit moves from one reviewer to another until the process is complete.
Using Direct Page Links for Forms and Dashboards in EPM Cloud

Planning supports direct URL links that allow you to access Planning forms and dashboards and to integrate with Oracle Sales Planning Cloud. You can use direct page links to create a URL that points to a specific form or dashboard. If users have appropriate access permissions, they can use the URL to go directly to the form or dashboard to view or update it within Planning. Note that this feature is intended for links from external services only; it is not intended for internal action menu links within the application.

To use this feature, first create a form or dashboard, and then assign access and manage users and roles. You can then set up the URL for a direct page link and communicate the URL to users.

Note:

Composite forms and internal action menu links within the application are not supported for direct page links.

URL Parameters

The URL includes the following parameters. For ObjectType, specify either FORM or DASHBOARD. For ObjectName, substitute the name of your form or dashboard for FormName or DashboardName. Specify a point of view using POV and a comma-separated list of members; substitute your own member names for M1 and M2.

- virtualhost
- Direct=True
- ObjectType=FORM or DASHBOARD
- ObjectName=FormName or DashboardName
- POV=M1,M2

URL Pattern

Use the following syntax to create a direct page link with the URL parameters for a form or dashboard. In the URL, include one of the supported artifacts, FORM or DASHBOARD. Substitute the name of your own form or dashboard for FormName or DashboardName. If parameters like the form name or POV members have a space, replace the space with the URL encoding for a space (%20).

Include a POV with a comma-separated list of member names and substitute your member names for M1 and M2 as shown in the examples. If a member name in the POV contains a comma, enclose it in URL-encoded double quotation marks (%22).
• **Forms** – use this direct page link URL pattern:

https://virtualhost/HyperionPlanning/faces/LogOn?Direct=True&ObjectType=FORM&ObjectName=FormName&POV=M1,M2

For example, if your URL points to a form named Form01, and your POV includes members called Mem1 and Mem2, use a URL like the following:

https://virtualhost/HyperionPlanning/faces/LogOn?Direct=True& ObjectType=FORM&ObjectName=Form01&POV=Mem1,Mem2

If the URL points to a parameter with a name that contains a space, replace the space with %20 as shown in this example for a form named Form 01. If other parameters have spaces, follow the same approach.

https://virtualhost/HyperionPlanning/faces/LogOn?Direct=True&ObjectType=FORM&ObjectName=Form%2001&POV=Mem1,Mem2

• **Dashboards** – use this direct page link URL pattern:

https://virtualhost/HyperionPlanning/faces/LogOn?Direct=True&ObjectType=DASHBOARD&ObjectName=DashboardName&POV=M1,M2

For example, if your URL points to a dashboard named Dashboard01, and your POV includes members called Mem1 and Mem2, use a URL like the following:

https://virtualhost/HyperionPlanning/faces/LogOn?Direct=True&ObjectType=DASHBOARD&ObjectName=Dashboard01&POV=Mem1,Mem2

### User Authentication for Secured Access

The direct page link servlet requires authentication. Users must log on to gain access to the target page. After logon, users are redirected to the target page. If a user has already logged on, the target page is immediately displayed without requiring logon.
Frequently Asked Questions

Related Links

• What is the PBCS Plus One Business Process Option?
• How Can I Brand My Environment for Easy Recognition?
• Can I Change the Logo in My Business Process? What Graphical Elements can I Customize?
• Can I Customize the Look of My EPM Business Processes and Change the Background?
• Where Can I Find Documentation and Examples for Groovy Business Rules?

What is the PBCS Plus One Business Process Option?

If you purchased EPM Cloud before June 4, 2019, in addition to the Standard PBCS and Enterprise PBCS licenses, Oracle offers a PBCS Plus One Business Process option. When you purchase the Enterprise PBCS license or PBCS Plus One Business Process option license, depending on the specific option you license, you can create an application of Enterprise type or convert a Standard application to an Enterprise application type and choose to configure Enterprise business processes: Financials, Strategic Modeling, Workforce, Projects, or Capital.

Some features in the product are enabled only when your application is an Enterprise application type. One such feature is using Groovy rules (see Using Groovy Rules).

How Can I Brand My Environment for Easy Recognition?

See Customizing Your Display.

Can I Change the Logo in My Business Process? What Graphical Elements can I Customize?

Yes, you can change the logo. See Customizing Your Display.

Can I Customize the Look of My EPM Business Processes and Change the Background?

Yes, you can customize the look and change the background. See Customizing Your Display.

How Can I Display Member Names in Multiple Languages?

Alias tables enable you to display member names in multiple languages. First, create an alias table for each language. Load member names for each language into the various alias tables. Then advise users of the specific regions to select their corresponding alias table as a user preference. See Administering Alias Tables and Setting the Display of Member Names or Aliases.
Where Can I Find Documentation and Examples for Groovy Business Rules?

Note:

- If you purchased EPM Cloud before June 4, 2019, you can use Groovy rules only for applications of type “Enterprise” (available with Enterprise PBCS or PBCS Plus One licenses), Oracle Strategic Workforce Planning Cloud, or Oracle Sales Planning Cloud.
- If you purchased EPM Cloud Standard Edition after June 4, 2019, you can use the Groovy rules included with the modules.
- If you purchased EPM Cloud Enterprise Edition after June 4, 2019, you can use the Groovy rules included with the modules. Additionally, you can edit and create Groovy rules in Custom, Module, and Free Form business processes.

For basic information on working with Groovy rules, see Using Groovy Rules.

For details about creating business rules with Groovy, see Creating a Groovy Business Rule in Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud.

To view a technical reference and sample scripts for designing Groovy rules, see the Java API Reference for Oracle Enterprise Performance Management Cloud Groovy Rules on the cloud help center. You can also access this reference from the Oracle Enterprise Planning and Budgeting Cloud Academy. To access the Academy, sign into a business process instance, and then click Academy.

The API reference includes examples that demonstrate the syntax and power of the EPM Groovy object model.

To see example Groovy scripts:

2. Do one of the following:
   - On the main page, scroll down to Example Groovy Scripts. To view the examples, click the word “here” in the sentence about example Groovy scripts.
   - In the left pane, under All Classes, click a topic to view detailed information and examples of Groovy scripts. For example, click StrategicModel to see example scripts to load data from Planning and push it into Strategic Modeling, and to load data from Strategic Modeling and push it into Planning.