Oracle® Cloud

Administering Planning for Oracle Planning and Budgeting Cloud Service

ORACLE®

Oracle Cloud Administering Planning for Oracle Planning and Budgeting Cloud Service

Contents

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.

Documentation Feedback

Send feedback on this documentation to: epmdoc_ww@oracle.com

Follow EPM Information Development on these social media sites:

LinkedIn - http://www.linkedin.com/groups?gid=3127051&goback=.gmp_3127051

Twitter - http://twitter.com/hyperionepminfo

Facebook - http://www.facebook.com/pages/Hyperion-EPM-Info/102682103112642

Google+ - https://plus.google.com/106915048672979407731/#106915048672979407731/posts

YouTube - http://www.youtube.com/user/OracleEPMWebcasts

1

Using Planning

About Planning

Use Oracle Hyperion Planning to drive collaborative, event-based planning processes. Decision-makers and managers collaborate with budget holders to optimize the planning process, adapting rapidly and ensuring optimal plans.

Watch this video to see a demonstration of Planning.

▶ Video

Planning Features

Planning:

- 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 implementation 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

2

Getting Started with Planning

About Planning Applications

A Planning 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 Planning application administration tasks, including:

● Creating, deleting, and managing a Planning application.

● Creating forms, task lists, and menus.

● Managing currency conversion and exchange rate tables.

● Identifying the review and approval process, requirements, and participants, and managing the budgeting process.

Understanding Implied Sharing in Planning Forms

Some members are shared even if you do not 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. Planning assumes (or implies) a shared member relationship 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 (in this case, the parent inherits the value of the first child, regardless of the aggregation settings used for the child)

In a Planning 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.

Using the Simplified Interface

Subtopics

- Logging On to the Simplified Interface
- Enabling Access to Planning Artifacts On Mobile Devices
- Considerations When Using the Simplified Interface
- About Cell Formatting When Using the Simplified Interface
- Copying and Pasting Data Between Microsoft Excel and Planning Web Forms
- About Instant Save
- About Autosave

Administrators can make a simplified interface available to planners for accessing common planning tasks. For example, using the simplified interface, planners can access their task lists, work with data, approve budgets, view reports, and control settings. Administrators can use the simplified interface to create and manage applications, schedule jobs, and make announcements.

Logging On to the Simplified Interface

Planners and administrators can launch the simplified interface in one of two ways:

- From the Planning landing page, click **Simplified Interface**. This will launch a new browser window showing the simplified interface.

- 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`.

Enabling Access to Planning Artifacts On Mobile Devices

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

➤ To enable access to Planning artifacts on mobile devices:

1 Select **Administration**, then **Manage**, and then **Tablet Access**.

2 In **Tablet Access Setup**, select the Forms, Tasks, and Rules tabs to view and select the artifacts that you want to enable.

Considerations When Using the Simplified Interface

Note the following considerations when using the simplified interface:

- The simplified 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 standards 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.
- Some administrative functions are available only if you are accessing the simplified interface from the desktop.
- The simplified interface will not allow you to create Oracle Hyperion EPM Architect applications. Nor can you use **Navigator** in the simplified interface to directly access the Oracle Hyperion EPM Architect user interface.
- For information on copying and pasting data between Microsoft Excel and Planning web forms, see "Copying and Pasting Data Between Microsoft Excel and Planning Web Forms" on page 19.

About Cell Formatting When Using the Simplified Interface

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 or tap **Actions**, and then **Apply**.

2 Click or tap:
   - **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 Planning 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 Planning 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 or tap **Paste** to paste the data into the Planning Web form.

The following features are supported in Internet Explorer browser only:

- Copying data from Planning web forms and paste it into Microsoft Excel
- Copying data from one Planning 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 or tap **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" on page 63.

3

Setting Up Access Permissions

About Planning User and Role Management

Planning implements several security layers to ensure security. Infrastructure components, which are implemented and managed by Oracle, create a secure environment for Planning.

Service level security is ensured using the following mechanisms that permit only authorized users to access Planning:

- Single Sign-On (SSO)
- Role-based access to Planning

Global roles are granted through the Oracle Cloud Identity Console. For information about managing Planning security and users and accessing the Oracle Cloud Identity Console, see *Using Oracle Planning and Budgeting Cloud Service*.

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

Watch this overview video to learn about managing users and roles in Planning.

▶ Overview Video

Planning Artifacts That Can Be Assigned Access

You can assign access permissions to:

- Dimensions, including user-defined dimensions. Assign access 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, access can be assigned to Year and Periods.

- Launch privileges to business rules

- Forms

- Form folders and business rule folders

- Task lists

Types of Access Permissions

Access permissions include read, write, and none. You can also set who can launch which Oracle Hyperion Calculation Manager business rules.

- **Launch**: Allow launch privileges

  Note: View user types have no write access to dimension members, so cannot launch business rules having runtime prompts that include members, dimensions, member ranges, or cross-dimension runtime prompt types. They can, however, launch business rules having runtime prompts of other types (for example, date type).

- **No Launch**: Disallow launch privileges.

  Note: If a user inherits Launch access permission to a business 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 individual's access permissions conflict with those of a group the user belongs to, user access permissions take precedence.

Inheriting Access 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 access permissions. Assigned access permissions take precedence over inherited access permissions. You can include or exclude the member from the access permissions setting.

Table 1   Options for Inheriting Access Permissions

Inheritance Option	Access Permission Assignment
Member	Only to the currently selected member
Children	To all children members in the level below the currently selected member

Inheritance Option	Access Permission Assignment
iChildren	To the currently selected member and all children members in the level below it
Descendant	To all descendant members below the currently selected member
iDescendant	To the currently selected member and all descendant members below it

How access permissions are evaluated

When evaluating access permissions, Planning gives precedence in this order:

1. Role-level security. Users with the Service Administrator role have access to all application elements.

2. For Power User and Planner user types, access permissions that are specifically assigned to users.

3. Access assignments that are acquired by belonging to a group.

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

Enabling Access Permissions for Dimensions

Use the **Dimensions Property** tab to set access permissions to user-defined custom dimension members.

➤ To enable access permissions for dimensions:

1 Select **Administration**, then **Manage**, and then **Dimensions**.

2 In **Dimension**, select the dimension.

3 Select **Actions**, and then select **Edit**.

4 In **Dimension Properties**, select **Apply Security**.

> **Note:** If you do not select this option, there is no security on the dimension, and users can access its members without restriction. See "Enabling Access Permissions for Dimensions" on page 23.

5 Click **Save**.

Click **Reset** to revert to previously saved values.

Assigning Access to Members and Business Rules

Before you can assign access to members of user-defined custom dimensions, you must select the **Apply Security** check box on the dimension's **Property** tab. See "Enabling Access Permissions for Dimensions" on page 23.

➤ To assign access to members or business rules:

1 Select the member or business rule:

- **For members:** Select **Administration**, then **Manage**, then **Dimensions**, and then select the dimension and member.

- **For business rules:** Select **Administration**, and then **Business Rule Security**. Select the folder containing the business rules, and then select the business rules.

2 Select **Actions**, and then select **Assign Access**.

3 Add, edit, or remove access.

See .

Adding, Editing, and Removing Access

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

➤ To assign, edit, and remove access permissions to members or business rules:

1 Select the member or business rule:

- **For members:** Select **Administration**, then **Manage**, and then **Dimensions**. Then select the dimension and member.

- **For Calculation Manager business rules:** Select **Administration**, and then **Business Rule Security**. Select the folder containing the business rules, and then select the business rules.

2 Select **Actions**, and then select **Assign Access**.

3 To add access:

   a. Click **Add Access**.

   b. Select the users and groups to access the selected member or business rule.

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

   c. **Optional for members:** Select a relationship.

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

   d. Select an option:

       - Click **Launch** to allow the selected users and groups to launch the selected business rules.

       - Click **No Launch** to prevent the selected users and groups from launching the selected business rules.

   e. For the selected users or groups, select the access type and click **Add**.

   f. Click **Close**.

4 To modify access:

   a. Click **Edit Access**.

b. For the selected member or business rule, select the access type for the displayed users or groups.

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

For business rules only:

- Click **Launch** to allow the selected users and groups to launch the selected business rules.

- Click **No Launch** to prevent the selected users and groups from launching the selected business rules.

c. **Optional for members**: Select a relationship.

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

d. Click **Set**.

e. Click **Close**.

5 **To remove access:**

a. Select the users and groups for whom to remove access to the selected member or business rule.

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

b. Click **Remove Access**.

c. Click **OK.**

d. Click **Close**.

About Effective Access Permissions to Shared Members

You cannot assign access directly to a shared member. A shared member inherits access permissions from its base member, parent, or ancestor.

Planning 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 Entity Members

Parent Entity	Child Entities
United States	
	CA (base)
	NY
West	
	CA (shared)

Parent Entity	Child Entities
	NV
Sales Region 1	
	CA (shared)

Table 2   Example of Inherited Access to Shared Members

Case	Access Permission	Effective Access for Base and Shared Member CA	Explanation
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.

Managing Access to Forms and Folders

Subtopics

- Assigning Access to Forms and Folders
- Adding, Changing, and Removing Access to Forms and Folders

Assigning Access to Forms and Folders

Administrators can assign access to forms, form folders, and Calculation Manager business rule folders. (For information on assigning access to business rules and members, see "Types of Access Permissions" on page 22 and "Assigning Access to Members and Business Rules" on page 23.)

Principles:

- **Forms:**
  - Planners and Power Users can view or enter data only into forms to which they have access (and can work only with members to which they have access).
  - Service Administrators and Power Users can design forms.
  - Power Users can access forms they created or to which a Service Administrator assigned them access.

- Service Administrators have write access to all dimension members and to all forms.
- **Business rules**: Planners and Power Users can see and launch only business rules to which they are assigned Launch access.
- **Form folders and business rule folders:**
  - Planners who are assigned access to a form folder can access the forms in that folder, unless they are assigned more specific access. Likewise, planners have Launch access to the Calculation Manager business rules in folders to which they are assigned access, unless they are assigned more specific access.
  - When you assign access to a folder, all folders under it inherit that access.
  - If you assign specific access (for example, 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 does not see Folder2.
  - If you assign specific access (for example, Launch) to a Calculation Manager folder, that access permission takes precedence over its parent folder's access permissions. For example, if a user has Launch access to RulesFolder1 that contains RulesFolder2 to which the user has No Launch access, the user can open RulesFolder1, but does not see RulesFolder2.
  - 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.
  - If a user has No Launch access to a Calculation Manager folder called RulesFolder1 that contains a business rule called Rule1 to which the user has Launch access, the user can see RulesFolder1 and Rule1.

For procedures, see "Adding, Changing, and Removing Access to Forms and Folders" on page 27.

Adding, Changing, and Removing Access to Forms and Folders

➤ To assign access to forms, form folders, and Calculation Manager business rule folders:

1 Select the form or folder.

- For forms and folders, see "Selecting and Opening Forms and Folders" on page 86.
- For business rule folders, select **Administration**, then **Manage**, and then **Business Rule Security**.
- For business rules, select **Administration**, then **Manage**, and then **Business Rule Security**. Open the business rule folder containing the business rules and select the rules.

You can assign access to only one form, business rule, or folder at a time.

2 Select **Actions**, and then select **Assign Access**.

3   **To add access to forms and folders:**

    a.   Click **Add Access**, and select the users or groups to access the form or folder.

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

    b.   For **Type of Access**, select the kind of access users or groups have to the form or folder.

        For business rules or their folders only:

        ●   Click **Launch** to allow the selected users and groups to launch the selected business rules.

        ●   Click **No Launch** to prevent the selected users and groups from launching the selected business rules.

    c.   Click **Add**.

    d.   Click **Close**.

4   **To change which users can use or change forms or folders:**

    a.   Select the users or groups for which to change access, and click **Edit Access**.

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

    b.   For **Type of Access**, select the kind of access users or groups have to the form or folder.

    c.   Click **Set**.

    d.   Click **Close**.

5   **To remove access from forms or folders:**

    a.   Select the users or groups for which to remove access, and click **Remove Access**.

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

    b.   Click **OK**.

Reporting on Access Permissions

Subtopics

- Selecting Reporting Objects
- Selecting Reporting Options
- Working with Access Permissions Reports
- Examples of Tracked Actions

You can view current access permissions and print reports.

➤ To report on current access permissions for users and groups in Planning:

1   In Planning, select **Tools**, then **Reports**, and then **Access Control**.

2   On **Select User or Group**, select from the available options.

3   From the left **Available** panel, select and move users or groups to report to the **Selected** panel:

4   Click **Next**.

Selecting Reporting Objects

You can report on these objects: Accounts, Scenarios, Versions, Entities, user-defined custom dimensions, and forms.

➤ To select reporting objects:

1   Start the Access Control Report.

See "Reporting on Access Permissions" on page 28.

2   On **Select Objects**, select the Planning objects on which to report.

3   Click **Next**.

Selecting Reporting Options

➤ To specify options for access reports:

1   Start the Access Control Report.

See "Reporting on Access Permissions" on page 28.

2   On **Report Options**, for **Show Matching Access of Type**, select the access to view: **Read**, **Write**, or **None**.

3   For **Group the Results By**, select how to view the report: **Users** or **Objects**.

4   From the **Report Type** sections, select **Assigned Access** or **Effective Access**:

Table 3   Access Report Types

Report Type	Description	Options
Assigned Access	Summarizes access permissions that administrators assign	Specify whether access is assigned by member selection relation or group membership: ● **Show Matching Access of Relation**: **Member**, **Children**, **Children (inclusive)**, **Descendants**, or **Descendants (inclusive)**. ● **Show Inherited From Group**: Show access permissions inherited by users in a group.
Effective Access **Note:** Effective Access reports cannot be generated for groups.	Summarizes access assignments as Planning evaluates them (for example, by member selection relation, such as children, or group membership). This is useful if there are conflicts in access permissions.	Describe the origin of the effective access by selecting **Show Effective Access Origin**. For example, a user named JSomebody may be assigned Write access to Entity1 and belong to a group named Sales that is assigned Read access to Entity1. This setting would show that JSomebody has Write access to Entity1 because individual assigned access supersedes access inherited by group membership.

5   Click **Finish**.

Adobe Acrobat launches, displaying the report online.

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.

Examples of Tracked Actions

For each recorded action, Planning tracks:

Tracked Changes	Examples
The type of change	Metadata, data, form, access permissions, planning units
The affected object (The columns ID_1 and ID_2 in the audit report help define the object that changed.)	Form: Expenses 04 Group: Marketing
User	VHennings
Time Posted	08/22/2013 8:17
Action	Add
Property	Currency
Old value	Default
New value	USD

4

Managing Planning Databases

Using Broadcast Messaging

Use broadcast messaging to communicate a text message to Planning users logged on to Planning. You should send broadcast messages about system availability or periodic maintenance and you can send broadcast messages to request that users log out before migrating applications.

The messages are displayed in logged-on users' browsers upon page refresh or when they visit another page.

➤ To send broadcast messages:

1   Select **Administration**, then **Application**, then **Broadcast Messages**.

2   In **Create Message**, enter the message to broadcast.

3   Click **Send**.

Viewing Usage Statistics

You can determine which Planning users are logged on to Planning, and how long since they accessed Planning.

➤ To view usage statistics:

1   Log on to Planning.

2   Select **Administration**, then **Application**, then **Statistics**.

Creating and Refreshing Application Databases

On the **Manage Database** page, you can create and refresh application databases, which are used to store data in Planning for each plan type in the application. Databases are structured according to dimensions, hierarchical members, attributes, and other data specified in an application.

When you create an application, select **Create** to update the Planning multidimensional databases that store application data. While building your outline, you can transfer database changes and access permissions separately to improve performance and make changes quickly available to users. When the outline is complete, Oracle recommends that you include access permissions when refreshing database information.

You must refresh the application database whenever you change the application structure. Changes made to an application are not reflected to users performing data entry and approvals tasks until you refresh the Planning databases for the application. For example, when you modify properties of an Entity member, add a Scenario, or change access permissions, these changes are stored in the Planning relational database until you refresh the application database.

During refresh:

- Currency conversion calculation scripts are updated.
- Members and associated properties are propagated from the relational database to the multidimensional database.
- Custom attributes are added, modified, or deleted in the multidimensional database.
- Exchange rate values are repopulated in the Planning outline.
- Member formulas for certain accounts are dynamically generated or updated.
- Additions or changes to alias tables and their associations to dimensions or members are updated.
- The Planning application is restructured.
- UDAs are added to the Planning application.

When an administrator uses **Create Database** or **Refresh Database**, all other tasks are unavailable to other users.

All users must be logged off from the Planning application before Planning databases can be refreshed. Oracle recommends that administrators send a broadcast message to all users, asking them to save their work and close the application before the Planning application is refreshed. Planning does not log out users during refresh. See "Limiting Use of an Application During Maintenance" on page 175.

Application refresh time depends on factors such as the number of entities and users in the application, and many users lengthens refresh time. To maximize system availability, you can transfer access permissions information during nonpeak hours.

Watch this video to learn about updating the database in Planning.

▶ Video

➤ To create or refresh the application database:

1 Select **Administration**, and then **Application**.

2 Select **Create Database** or **Refresh Database**.

3 Select options:

- **Database**: Creates or refreshes a Planning database for the application.

- **Update custom-defined functions**: Updates Planning custom-defined functions for the application when creating or refreshing the application database.

- **Shared Members**: Evaluates access permissions that are set for all instances of the member (base and shared) and applies the least restrictive access permission to them all. For example, if a shared member's parent has Write access assigned to all its children, and another shared member under another parent has Read access assigned to all its children, the base member and all its shared members are given Write access (for examples, see "About Effective Access Permissions to Shared Members" on page 25).

  Note: If your application does not depend on shared member security, consider clearing this option to improve performance.

  If this option is cleared, shared members inherit the security assigned to the base member.

4 To create or refresh data in the Planning database, click **Create** or **Refresh**.

5 Review the confirmation message. To continue, click **Create** or **Refresh**. After the update completes, click **Finish**.

  Note: If the create or refresh process takes some time, click **Run in Background** to run the process in the background without displaying the status.

Managing Exchange Rates

Exchange rate tables enable budget preparers to create plans in different currencies. For example, you can specify Yen as the base currency for the Japan entity and US dollars for the United States entity. When you display a form with values for the Japan entity and display currency set to US dollars, the Yen exchange rate is used to convert values for Japan to US dollars. If the display currency is set to Yen, the exchange rate for US dollars converts values for the United States entity to Yen.

To use exchange rates, you must select Multiple Currencies when creating an application. You can set up exchange rates using the procedure in "Creating Exchange Rate Tables" on page 34 and "Editing Exchange Rate Tables" on page 34.

A multiple currency application stores exchange rates with the HSP_Rates dimension, which 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, you must 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.

Creating Exchange Rate Tables

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

➤ To create exchange rate tables:

1 Select **Administration**, then **Manage**, and then **Exchange Rates**.

2 Click **Create**.

3 Specify information for the Exchange Rate table.

4 Click **Save**, and then define settings for the table as specified in .

Editing Exchange Rate Tables

The default currency and triangulation currencies are available as destination currencies. You can enter exchange rates from source currencies to the default or triangulation currencies. You enter conversion values between the default currency and all the 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 plan types.

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 cannot select the application's default currency as a triangulation currency.

When you input exchange rates for converting from one currency to another, you can select Multiply or Divide as the calculation method.

➤ To edit exchange rate tables:

1 Select **Administration**, then **Manage**, then **Exchange Rates**, select the table to edit, then click **Edit**.

2 In the **Rate Table** tab, select options.

3 Click **Next**.

4 On the **Exchange Rate** tab, set options:

Table 4   Exchange Rate Table Options

Option	Description
To Currency	The currency for which to enter conversion rates (the default currency or a triangulation currency).
Show Years	The time periods displayed (by default, the current application year).
Rate Table Name	The name of the exchange rate table (display only).
Method	Multiply or Divide, the mathematical operator that determines how values are calculated between the source and destination currencies.
Historical	For all time periods, the exchange rate for accounts whose Exchange Rate Type is set to Historical. The account's Data Type must be set to Currency. Historical is typically used for balance sheet account types. A historical exchange rate may reflect a calculated rate over time, a rate for a point in time before the application's calendar, or a rate that was in effect when an event occurred.
BegBalance	The value of balance sheet accounts. There is one beginning balance time period, the first time period in the application. Rates for the Beginning Balance time period are populated for each year in the application. Scenarios that do not include the first year of the application can include a Beginning Balance time period.
Avg	For time periods, the exchange rate for accounts whose Exchange Rate Type is set to Avg, or Average. Avg is typically used for Revenue and Expense account types, or for Saved Assumption account types whose Time Balance is set to Flow. The account's Data Type must be Currency.
End	For time periods, the exchange rate for accounts whose Exchange Rate Type is set to Ending. Ending is typically used for Asset and Liability account types, or for Saved Assumption account types whose Time Balance is set to Balance. The account's Data Type must be Currency.

Tip: After entering values, fill in the value throughout the current year or all years in the table. For example, if you enter a value for Avg in the Jan11 cell and select **Fill Year**, the value is spread to all the months in 2011. If you select **Fill Table**, the value is spread to the months for all the years included in this exchange rate table. To fill in values, enter a value for Avg or End, right-click the cell, and select **Fill Year** or **Fill Table**.

5   Click **Save**.

Deleting Exchange Rate Tables

➤ To delete exchange rate tables:

1   Select **Administration**, then **Manage**, and then **Currency Conversion**.

2   Select the exchange rate table to delete.

3   Click **Delete**.

4   At the prompt, click **OK**.

Managing Currency Conversions

➤ To manage currency conversions:

1 Select **Administration**, then **Manage**, and then **Currency Conversions**.

2 Click **Create**.

3 In the **Create File** tab, specify information for the Currency Conversion Script file, and then click **Next**.

4 In the **Details** tab, select information for the Currency Conversion Script details.

   Click **Select Member** to select members for fields.

Working with Currency Conversion Calculation Scripts

If multiple currencies are enabled for the Planning application when you create a currency conversion, a currency conversion calculation script is created, based on selected scenarios, versions, and currencies. A second calculation script is generated by Planning, which copies appropriate exchange rates to the account, based on account rate types. For currency conversion, the Account type always takes precedence. Data type evaluation order is not considered. The copy calculation script is named `HspCrtB.csc` for bottom-up versions and `HspCrtT.csc` for target versions. Running the copy calculation script enables the currency conversion calculation script to run in BLOCK mode, which is more efficient than CELL mode.

The selected scenarios, versions, and currencies must be able to store data in the database outline. Dynamic Calc, Dynamic Calc and Store, and Label Only are virtual members that do not store data. There is no benefit to running the copy of the currency conversion calculation script if the target version has virtual members because Planning discards the results of the calculation for these members.

To convert currencies correctly, the first time a currency conversion is launched, administrators must run the copy currency rates calculation script and the currency conversion calculation script. After launching the `HSPCrtB.csc` or `HspCrtT.csc` copy calculation script, you must launch them again if you change the database outline (for example, by adding or changing exchange rates, account rate types, versions, scenarios, accounts, or user-defined dimension members).

To create the copy currency calculation script for calculation scripts, you must regenerate currency conversion calculation scripts.

Optimizing Performance

Subtopics

● About Reordering Dimensions
● Other Performance Optimization Tips

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" on page 206, "About Reordering Dimensions" on page 37, and "Setting Dimension Density and Order" on page 219).

- Set the **Suppress Missing Blocks** option (see "Setting Form Grid Properties" on page 63).

About Reordering Dimensions

The order of dimensions is critical for the structure and performance of a Planning 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, do not consolidate children to create data.

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 plan type to reduce the database size in your current plan type.

- 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 Planning is launched and forms are opened, caches are loaded, which takes more time than in subsequent sessions. So, before your users start using Planning, launch Planning and open the most commonly used forms to reduce the loading time for your users.

5

Importing and Exporting into a Planning Application

About Importing and Exporting

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

- Import metadata using Planning import feature. See "Importing Metadata" on page 39.

- Populate Planning data cells by importing data from your source system. You can import data in two ways:

  ❍ Using Data Management. See "Importing Using Data Management" on page 47 and the *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

  ❍ Using Planning's Import feature. See "Importing Data" on page 42.

To import data, you must specify into which cell Planning stores each unit of data by defining their dimensional intersection.

Importing and Exporting Metadata and Data

Subtopics

- Importing Metadata
- Importing Data
- Exporting Metadata to a File
- Exporting Data to a File
- Viewing Import and Export Status
- Administering Data Load Settings

Importing Metadata

To import metadata, perform these tasks:

- Define a metadata flat file. See .

- Load the metadata flat file. See .

Watch this video to learn about importing metadata in Planning.

▶ Video

Creating the Metadata Flat File

Use the Import and Export functionality to import more metadata or to perform incremental updates from the source system. In order to import, you must create a metadata flat file in CSV format. For each dimension for which you want to import members and their properties, create a metadata flat file. The metadata flat file must contain:

1. A header record, the first line in the file, that:

   - Lists the dimension and any member properties used by subsequent metadata records; the header record and subsequent records do not 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

2. After the header record, a list of metadata records that you want to import or update. Each metadata record contains a comma-separated list 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.

```
Entity,Parent,Data Storage,Plan Type (Plan1),Plan Type (Plan2),Plan Type (Plan3),Data Type,Base Currency,
Alias: Default NoEntity,,Never Share,TRUE,TRUE,TRUE,Unspecified,USD,Unspecified Entity
AllEntities,,Never Share,TRUE,TRUE,TRUE,Unspecified,USD,Total Entity
1000,AllEntities,Never Share,TRUE,TRUE,TRUE,Unspecified,USD,Agencies
1001,1000,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,Appellate 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,Dept. of Revenue
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
```

In the following example, the import file loads an Entity dimension with the required header record and three data records. 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 is not 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**.

Importing the Metadata Flat File

➤ To import metadata into the Planning application:

1 **Create a metadata flat file for each dimension you want to import and ensure that you understand the process.**

2 **Select Administration, then Import and Export, and then Import Metadata from File.**

3 **In Dimension to Import, select the dimension to load, whose member fields correspond to the header record in the metadata file. If needed, create a new dimension on import.**

- To load metadata for a user defined dimension, select **User Defined Dimension** from **Dimension Type**, and then select the dimension.

- If the file contains metadata for a user defined dimension which does not yet exist, select **New Dimension**, then **User Defined**, and then enter then name of the dimension in **Dimension Name**. Planning will create the dimension and load it's metadata.

4 **In Metadata File to Import, select the import file that contains a header record and metadata records.**

5 **Optional:** Display and set import options by expanding **Options:**

Table 5 Metadata Import Options

Import Option	Description
Database Refresh	Refresh the database after importing the metadata.
Delete All Planning Units	Select this option and **Before Importing, Delete All Members from the Dimension** to be able to delete started planning units and all members in the dimension identified in the import file.

Import Option	Description
Before Importing, Delete All Members from the Dimension	Select to remove attribute associations to members and approvals states, and delete all members of the import dimension before performing the import. **Caution:** Run **Validate File Format** before executing the import using this option. If there is a problem with the import file, all members in the import dimension are deleted.
Assume Parent's Plan Type if Unspecified in File	● Select to apply unspecified plan type settings from the parent to newly added members ● Clear to specify plan types for members
Lock Load Dimension before Load	● Select to prevent other users from making changes to the dimension until the load is completed ● Clear if you do not want to lock the dimension (not recommended unless you are performing a dry run)
Use the Member Order in the Import File	Select to maintain the order of members (except UDAs) in the import file
Sort Import File's Parent-Child Order	● Select to arrange child members below their parents ● Clear if parent-child ordering isn't required, and for better performance.

6   Click **Run** or click **Validate** to ensure the file is correctly set up.

When importing metadata, you have the option of running the import in the background. Click **Run In Background** on the progress bar to enable background processing.

Tip: For the status of multiple import operations, select **Tools**, and then **Import and Export Status**. From the resulting page, click the import operation for which you want to see details.

You can now set up the application and import data.

Importing Data

Subtopics

● Importing Data from a File
● Driver Member Casting Errors

Watch this overview video to learn about loading data in Planning.

▶ Overview Video

Importing Data from a File

➤ To import data from a file:

1 Select **Administration**, then **Import and Export**, and then **Import Data From File**.

2 Select the format of the data import file:

- **Planning**—Planning file format (see). If you select this file format:

  a. From **Dimension to Import**, select the dimension into which to load data, whose member fields correspond to the header record in the data file. This dimension must exist in the application.

  b. In **Data File to Import**, navigate to the data import file that contains a header record and data records in CSV format.

     Click **Update** to navigate to a different file.

  c. For **Driver Member Specification**, enter the driver members specified for the import dimension. If you leave this field empty and you specified import and driver members on the Administration **Data Load Settings** window, those settings are used for the import. Conversely, values that you enter here override those on the **Data Load Settings** window. See .

  d. Click **Options**, and specify the following:

     ❍ **Date Format**—Date format to use in the import, such as MM-DD-YYYY.

     ❍ **Clear Essbase Data**—Clear an Essbase block before performing an import.

- **Native Essbase**—Oracle Essbase file format (exported by an on-premises product). These files can be formatted in either Cell list or Column list format. Cell list format specifies a single cell value for each record, whereas Column list format specifies multiple cell values for each record. For example, if you specify Time on the column, you might have the values Jan, Feb, Mar, and so forth on the column, with 12 values for each record (that is, one for each month). Conversely, the Cell list equivalent would contain 12 records, one for each month. For information on exporting files from Essbase, see the on-premises *Oracle Essbase Administration Services Online Help*.

  Note: After looking at the format of data records in the exported file, you can create a similar file using whichever tool you want. For example, you can generate this file format using an ETL (extract, transform, and load) tool and source these files from Enterprise Resource Planning systems.

  If you select the **Native Essbase** file format:

  a. For **Plan Type**, select the plan type into which to import the data.

  b. In **Data File to Import**, navigate to the data import file.

     Click **Update** to navigate to a different file.

  c. Click **Options**, and provide information:

❍ **Date Format**—Date format to use in the import, such as MM-DD-YYYY.

❍ **Clear Essbase Data**—Clear an Essbase block before performing an import.

3   Click **Run**.

When importing data, you have the option of running the import in the background. Click **Run In Background** on the progress bar to enable background processing.

If a value in a data load input file does not match the evaluated driver member type; for example, if the evaluated type of a driver member is "date" and it's corresponding input value is not a valid date format, a driver member casting error occurs. See "Driver Member Casting Errors" on page 44.

Driver Member Casting Errors

A driver member casting error occurs when a value in a data load input file does not 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 is not 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, Planning will not reject the entire input record, but the error will be noted in the import and export status. See "Viewing Import and Export Status" on page 46.

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,u1,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: aaaaaaasl1), (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 Metadata to a File

➤ To export data to a file:

1   Select **Administration**, then **Import and Export**, and then **Export Metadata to File**.

2   From **Dimension**, select the dimension to export

3   Click **Run**, and then run or save the export file.

Exporting Data to a File

➤ To export data to a file:

1   Select **Administration**, then **Import and Export**, and then **Export Data to File**.

2   Choose the options that were set during form design:

Table 6   Export Data to File Options

Option	Description
Plan Type	The plan type
Row Members	The row information Example: `Idescendants(Ent_Total)`
Column Members	The column information Example: `Idescendants(YearTotal)`
POV Members	The Page dimensions and member information Example: `FY13,Current, BU Version_1,account1,Local`

3   Click **Run**, and then save or run the resulting export file.

If the data export exceeds three minutes in duration, the following message will display and the export will automatically run in the background:

```
"The export has exceeded the maximum configured wait time and will now be performed in the
background. To check export status, open the Import and Export Status page. Note that
exported data files should be downloaded and saved because they may be periodically
removed from temporary locations on the server."
```

Once this message displays, clicking **View Status** or selecting **Tools**, and then **Import and Export Status** will display a **Processing** status in the **Status** column. When the data export is complete,

the **Status** column will display a **Complete** status. Selecting the **Complete** status will launch the **View Status** window and display a green arrow. Click the green arrow to download the exported file from a temporary location and save it to a permanent location. If the file in the temporary location no longer exists, the following message is displayed:

```
"The file that you tried to download no longer exists on the server. Export your data
again."
```

Viewing Import and Export Status

To determine the progress of an import or export:

1. Select **Tools** and then **Import and Export Status**.

2. Perform any task:

   - Query the import and export processes run by criteria such as load ID, process type,
     and status. To query, select **View**, then **Query by Example** or click ⧉ , and then enter the
     criteria to use above each column.

   - Change the data that is displayed by selecting **View**, then **Columns**, and then enable or
     disable individual columns.

   - Change the order in which the columns are displayed by selecting **View**, then **Reorder Columns**, and then move columns up or down.

   - Detach the import or export status page.

   - Click ✖ to delete the selected import or export process from the status page.

Administering Data Load Settings

Specify parameters to enable data to be loaded directly into a Planning 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 Select **Administration**, and then **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 are 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 above **Data Load Dimension Parent**,

   b. To the right of the new field, click and select a parent member.

   For information about selecting members, see Chapter 7, "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 step 5.a through step 5.c.

   e. To duplicate or delete a row, select the check box to the left of the row, and click or . You can duplicate only one selected row at a time.

6 Click **Save**.

Importing Using Data Management

Subtopics

- Overview
- Importing Data
- Scheduling Jobs in Data Management
- Drilling Through to Source Data
- For More Information

Overview

The basic process for importing data includes:

- Exporting data from the source system to a CSV file

- Defining the relationship between the source system data to Planning members using data load mappings

- 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

- Importing the data into Planning by executing the data load rules

Importing Data

➤ To import data using Data Management:

1   Select **Navigate**, then **Administer**, and then **Data Management**.

2   Import data into the Planning application, using the information in the following sections and the *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

See these examples:

-

-

Import Scenario Case 1: Importing Data when Member Names Match

This example walks you through the steps to importing data into a Planning application when the member names in the source system match the member names in Planning.

See .

Review the Import Format

When you create a Planning 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 Planning 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, you adjust the format.

➤ Review the default Import Format:

1   Select **Navigate**, then **Administer**, and then **Data Management**.

2   Select the **Setup** tab, and then click **Import Format**.

3   On **Import Format**, confirm the settings.

Note:

- The **File Delimiter** drop-down list specifies the character that separates values in the file.

- The **Field Number** column 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" on page 49.

Review the Default Location

The Location is used as a container to load data from Data Management. When you create a Planning application, a default Location is automatically created with the name *appname*_1.

➤ To review the default location:

1   On the **Setup** tab, click **Location**.

2   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 Planning and Budgeting Cloud Service*.

3   Click **Save**.

See "Create the Data Load Mapping" on page 49.

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. The **Data Management** component 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 Select the **Workflow** tab, and then click **Data Load Mapping**.

2 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:
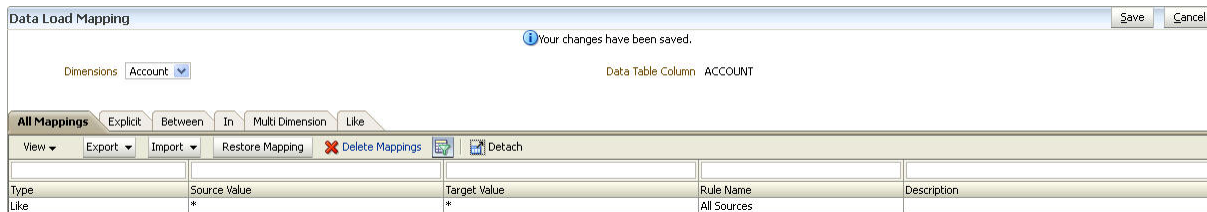
a. Because the source and target member names match in this example, on the **Like** tab, enter * in the **Source** and **Target** text boxes.

b. Under **Rule Name**, enter the name of the rule.

c. Click **Save**.

See .

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.

➤ On the **Workflow** tab, click **Data Load Rule**.

1 Click **Add**.

2 Under **Details** on the **Data Load Rule** window, you define the rule.

3 In this example, you will browse to a file:

a. Name and describe the rule.

b. For **Category**, select the Planning scenario.

When you create a Planning application, a default Category mapping is created to map to the Planning Scenario dimension and a default Period mapping is created based on the Planning Period dimension.

c. Select the **Plan Type** for which you want to load data.

d. Click **Select** to browse to the data load file.

e. Click **Save**.

4   To run the data load rule, using the mapping that you created, click **Execute**.

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 ERP data and Planning data are in different formats:

● The ERP Chart of Account structure consists of Company, Account, Sub Account, Department, and Product whereas the Planning application uses only Department, Account, and Product.

● The ERP Chart of Accounts has multiple levels in a Division, whereas the Planning application uses only the Division level.

● The ERP account numbers use eight characters, whereas the Planning application uses only the first six characters of the account.

This section gives an example of how to import data into a Planning application when the member names in the source system do not match the member names in Planning.

Refer to these steps and then perform the following procedure:

●

●

The following example assumes the source system has multiple products in the Electronics product line (for example, Televisions, Smart Phones, Computers), whereas the Planning 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 Planning into the Electronics level in the Product dimension.

Table 7   Electronics and Appliances Hierarchies

In Source System		In Planning Target Application	
Electronics		Electronics (includes all TVs, Phones, and Computers)	
	110 - TVs		na
	120 - Phones		na
	130 - Computers		na
Appliances	210 - Dishwashers	Appliances (includes all Dishwashers and Ovens)	
	220 - Ovens		na

➤   To set up the data load mapping for this example:

1   Select the **Workflow** tab, and then click **Data Load Mapping**.

2   From the **Dimensions** drop-down list, select **Product**.

3 On the **Like** tab, enter "1* " in the **Source** field to import data for all Electronics (product codes that begin with "1").

4 In the **Target** 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** field to import all Appliances (product codes that begin with "2"), and enter "Appliances" in the **Target** field.

5 Click **Save**.

When the data load rule is executed (see), data for all electronic products that begin with the code "1 " are imported and aggregated at the Electronics level in the Planning application.

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 Planning application on Friday of each week.

➤ To schedule a job:

1 Select **Navigate**, then **Administer**, and then **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 8   Schedule Details and Parameters

Schedule Details	Data and Time Parameters
Simple	Date
	Hours
	Minutes
	Seconds
	Select (AM/PM)
Hourly	Minutes
	Seconds
Daily	Hours
	Minutes
	Seconds
	Select (AM/PM)

Schedule Details	Data and Time Parameters
Weekly	Monday-Sunday
	Hours
	Minutes
	Seconds
	Select (AM/PM)
Monthly (day of month)	Monthly Date
	Hours
	Minutes
	Seconds
	Select (AM/PM)
Monthly (week day)	Day of Month
	Day
	Hours
	Minutes
	Seconds
	Select (AM/PM)

6   Click **OK**.

Drilling Through to Source Data

You can drill through on values from Planning 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 Planning. If a URL has been specified in the **Import Format** option for the Planning 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

Subtopics

- Defining Category Mappings
- Defining Periods
- Creating Data Load Mappings
- Defining Data Load Rules
- Managing Data Load Rules
- Using the Data Load Workbench
- Integrating Fusion Data in Planning

For more information, see the following sections in *Administering Data Management for Oracle Planning and Budgeting Cloud Service.*

Defining Category Mappings

When you create a Planning application, a default Category mapping is created to map the Planning Scenario dimension to an Data Management Category. To update the default mapping, see *Administering Data Management for Oracle Planning and Budgeting Cloud Service.*

Defining Periods

When you create a Planning application, a default Period mapping is created based on the Period for the Planning application. To update the default mapping, see *Administering Data Management for Oracle Planning and Budgeting Cloud Service.*

Creating Data Load Mappings

See *Administering Data Management for Oracle Planning and Budgeting Cloud Service* 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 Planning application. Data load rules are defined for locations.

See *Administering Data Management for Oracle Planning and Budgeting Cloud Service* for information on the following topics.

- Creating the Data Load Rule
- Defining the Data Load Rule Details
- Defining Data Load Rules Details for File Based Data Load Rules

Managing Data Load Rules

After data load rules are created, you can edit, run, or delete them.

See *Administering Data Management for Oracle Planning and Budgeting Cloud Service* 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 Planning and Budgeting Cloud Service.*

Integrating Fusion Data in Planning

Data Management facilitates data loading and write-back between a Fusion Financials Cloud application and a Planning application. Fusion Financials Cloud Service 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 Fusion GL Cloud 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 Fusion Financials Cloud Service. Both processes are supported using the Fusion WebCenter Content Management for the Cloud service with data provided by both the Fusion GL Cloud service and Planning.

For more information, see *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

Process Description

At a high level, the steps for integrating Fusion data in Planning:

1. Create a Planning application that includes the dimensions to which to map General Ledger balances. See .

2. Extract the Fusion General Ledger file from the Fusion Financials Cloud Service and move it to Planning. See "Extracting the Fusion General Ledger File" in *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

3. Register and configure a source connection for the Fusion General Ledger file. See "Configuring a Source Connection in Data Management" in *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

4. Load data from the Fusion General Ledger file to Planning. Then after mapping data to the Planning dimensions, write back the data to the Fusion Financials Cloud Service. See "Writing Back to the Fusion General Ledger for the First Time" in *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

5. Optionally, you can extract the data from the integrated Planning application to create a custom target application. See "Extracting the Integrated Planning and Fusion Data to a Flat File" in *Administering Data Management for Oracle Planning and Budgeting Cloud Service*.

6

Managing Forms

About Forms

Subtopics

- Form Components
- Form Design Considerations

Forms are grids for entering data. You can create simple forms or composite forms to meet your needs. Because composite forms consist of simple forms, you must create simple forms before creating composite forms.

Watch this video to learn about entering and working with data in Planning.

Video

Form Components

Subtopics

- Point of View
- Page Axis
- Rows and Columns

Point of View

Select members for the point of view to determine the context for pages, rows, and columns. For example, if the Scenario dimension is set to Budget in the point of view, all data entered in pages, rows, and columns goes into the Budget scenario. The point of view is set to one member, which a user cannot change, for each point of view dimension.

To simplify the form, in the point of view you can specify only relevant members or include user variables. See .

Page Axis

Use the page axis to specify combinations of members that may span dimensions so users can work with data in smaller, logical views. Each item on the page axis can have members selected from one or more 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 .

Rows and Columns

Rows and columns define the grid into which users enter data. For example, you can assign Unit Sales to the row axis and January to the column axis. When users 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 .

Form Design Considerations

Subtopics

- Forms and Plan Types
- Forms and Access Permissions
- Forms and Currencies
- Forms and Versions
- Forms and Attributes
- Forms and Shared Members
- Forms and Calculations

Forms and Plan Types

When you create a form, you associate it with a plan type, which determines the form's valid members. For example, if you assign a form to the Revenue plan type, you can add only accounts that are valid for the Revenue plan type. Entered data is saved to the selected plan type's database. You cannot change the plan type for a form after assigning it.

You can edit form accounts if their source plan type matches the form's plan type. If you add an account to a form for a plan type other than the account's source plan type, the account is read-only on that form.

Forms and Access Permissions

By assigning access to a form, you control which users can change its design (for example, its layout and instructions) and input data. Users can select only members to which they have read or write access. Users can edit forms only if they have access to at least one member of each secured dimension. For example, if users have read-only access to the Europe entity, the rows and columns on forms that include the Europe entity are displayed as read-only. Users can change data only for members to which they have write access.

Forms and Currencies

For a single-currency application, all entities use the currency selected when the application was created. For a multicurrency application, the selected Currency member on forms determines the currency in which values display. When the Currency member Local is selected for rows or columns, no currency conversion occurs for those rows or columns, and users can enter data for the entities in their native currency. If a currency member other than Local is selected, data values are converted to the selected currency 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" on page 81.

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.

Forms and Attributes

You can select members by selecting a shared attribute. For example, select the South attribute to include members having the South attribute. Values can be entered and saved into rows and columns that use attributes.

Forms and Shared Members

You cannot select shared members individually; instead, select them using a relationship function. For example, you could select an alternate functional rollup to include all members under that rollup. Values can be entered into rows or columns that display shared members and are saved to the base members in the database. Shared members display the same as base members in forms.

Forms and Calculations

To optimize calculations, select row members using relationships (such as Descendants or Children) instead of selecting children individually. Calculating totals for the parent of individually selected children could take several passes, depending on the number of hierarchy levels.

Creating Simple Forms

Subtopics

- Setting Form Layout
- Setting Form Precision and Other Options
- Creating Asymmetric Rows and Columns
- Adding Formula Rows and Columns
- Defining Simple Form Page and Point of View

Watch this video to learn about creating simple forms in Planning.

▶ Video

Table 9   Simple Form Creation Checklist

Task	Want to Know More?
Set form layout, including: - Adding form rows and columns - Assigning dimensions to columns and rows - Selecting dimension members for users to work with - Setting grid properties for the form - Setting dimension properties - Adding formula rows and columns - Setting display properties for the form - Setting printing options for the form - Adding and updating validation rules in forms	See "Setting Form Layout" on page 61.
Define page axis and point of view	See "Defining Simple Form Page and Point of View" on page 71.
Select members	See Chapter 7, "Using the Member Selector."
Set form precision, context menus associations, and whether to enable dynamic user variables	See "Setting Form Precision and Other Options" on page 68.
Select business rules and set properties	See "Using Business Rules" on page 91.
Define access permissions	See Chapter 3, "Setting Up Access Permissions."
Design formula rows and columns	See "Designing Forms with Formula Rows and Columns" on page 82.

Task	Want to Know More?
Design data validation rules	See "Designing Forms with Data Validation" on page 82.

➤ To create simple forms:

1 Select **Administration**, then **Manage**, and then **Forms and Ad Hoc Grids**.

2 Click **Actions**, and then select **Create simple form**.

3 On the **Properties** tab, provide a form name of up to 80 characters, and an optional description of up to 255 characters.

4 Select the **Plan Type** associated with the form. See "Forms and Plan Types" on page 58.

5 **Optional**: Provide instructions for working with the form.

6 Click **Next** to specify the form layout. See "Setting Form Layout" on page 61.

You can also create composite forms, which are forms that display several simple forms simultaneously. See "Creating Composite Forms" on page 72.

Setting Form Layout

Subtopics

- Setting Form Grid Properties
- Setting Dimension Properties
- Setting Display Properties
- Setting Printing Options
- Including Data Validation Rules in Forms

When you create forms, the **Layout** tab initially contains one row and one column, and all dimensions are in Point of View. When creating or editing forms, you can add rows and columns to a form, as necessary.

When setting row and column layout:

- Assign at least one dimension to the row and column axis.

- You cannot select the same dimension for multiple axes. (You can have dimensions on multiple axes if you set user variables in the point of view.)

- Select a dimension from any axis and drag it to the destination axis to move a dimension from one axis to another.

- Select display properties

➤ To set or update the form layout:

1 Open the form, and then click **Layout**.

See "Selecting and Opening Forms and Folders" on page 86.

2 **Optional**: Click ⬚ to select a dimension, and then drag it to **Rows** or **Columns**, or within a row or column.

> **Note:** Initially, all dimensions are in the form point of view. You can drag dimensions from point of view to rows, columns, or to page. You can also drag dimensions from any area in the grid (row, column, point of view, or page) to any other area.

3 **Optional**: Select another dimension, and then drag it to **Rows** or **Columns**, or within a row or column.

4 Select each dimension's members.

See Chapter 7, "Using the Member Selector."

5 **Optional**: To rearrange the order of dimensions in rows or columns, click ⬚, and then select **Move Dimension Up** or **Move Dimension Down**.

6 Select a row header (such as 1 or 2) to set row properties, or a column header (such as A or B) to set column properties using the information in this table (options are listed on the right under **Segment Properties**).

Table 10   Segment Properties

Option	Description
Apply to all rows	Apply settings to all rows; available when there are two or more rows. Clear this option to set different properties for each row.
Apply to all columns	Apply settings to all columns; available when there are two or more columns. Clear this option to set different properties for each column.
Hide	Hides the column or row on the form
Read-only	Creates a read-only row or column, enabling comparison of old, read-only data with new, editable data
Show separator	Creates a bold border before the segment to visually distinguish it
Suppress hierarchy	Suppresses indentation.
Suppress missing data	Hides rows or columns without data. Clear to display rows or columns with "#MISSING" in cells when data is missing.
Column width	● Default: Use the column width defined at the grid level (under **Grid Properties**) ● **Small**: Display seven decimal places. ● **Medium**: Display 10 decimal places. ● **Large**: Display 13 decimal places. ● **Size-to-Fit**: Force all columns to fit in the displayed space based on the top data cell value. ● **Custom**: Select a custom size to display more than 13 decimal places, up to 999 places.
Row height	● Default: Use the row height defined at the grid level (under **Grid Properties**). ● Medium: Display standard row height. ● **Size-to-Fit**: Force all rows to fit in the displayed space. ● **Custom**: Select a custom size in pixels for the row height.

To enable transferring global assumptions from a test to a production environment for a simple form, select **Global Assumptions Form**. Then update the form to store global assumptions such as a tax rate. See "Designing Forms with Global Assumptions" on page 82.

7   **Optional**: Add formula rows or columns. See "Adding Formula Rows and Columns" on page 70.

8   **Optional**: Add or update data validation rules. See "Including Data Validation Rules in Forms" on page 67.

Setting Form Grid Properties

Form grid properties set the general form row and column display.

Watch this video to learn about setting form grid properties in Planning.

▶️ Video

➤ To set form grid properties:

1   Open the form, and then click **Layout**.

See "Selecting and Opening Forms and Folders" on page 86.

2   In **Grid Properties**, set general row and column properties using the information in this table:

Table 11   Form Grid Properties

Option	Description
Suppress missing blocks	(Rows only) Improves the performance of the **Suppress missing data** setting when suppressing many rows, for example, 90% or more. The **Suppress missing blocks** setting can degrade performance if few or no rows are suppressed. Test forms before and after using this setting to determine whether performance is improved. Also test forms whenever you make significant changes to your application. With this setting selected, attributes may not display in forms, certain suppressed blocks may ignore Dynamic Calc members, and row members do not display as indented.
Suppress missing data	Hides rows or columns without data. Clear to display rows or columns with "#MISSING" in cells when data is missing.
Suppress invalid data	Hides rows or columns with invalid data. Clear to display rows or columns that contain cells with data that is invalid. Cells with invalid data are read-only.
Default row height	● **Medium** ● **Size-to-Fit**: Force all rows to fit in the displayed space ● **Custom**: Select a custom size in pixels for the row height
Default column width	● Small: Display seven decimal places ● Medium: Display 10 decimal places ● Large: Display 13 decimal places ● Size-to-Fit: Force all columns to fit in the displayed space based on the top data cell value. ● Custom: Select a custom size to display more than 13 decimal places, up to 999 places

Option	Description
Global Assumptions Form	To enable transferring global assumptions from a test to a production environment for a simple form, select **Global Assumptions Form**. Then update the form to store global assumptions such as a tax rate. See "Designing Forms with Global Assumptions" on page 82.
Enable Autosave	Selecting this option has these effects in a simple form in the simplified interface: ● 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 are displayed with a green background. ● Planners can successively undo actions with Ctrl+Z. **Note:** For optimal Autosave performance, you must have only dense dimensions on rows and columns. However, if you must put a sparse dimension on either a row or a column, for improved Autosave performance on block storage databases, enable hybrid aggregation with the Essbase configuration setting `ASODYNAMICAGGINBSO`. See also "About Autosave" on page 20.
Run Form Rules on Autosave	If **Enable Autosave** is selected, this option becomes available. If **Run Form Rules on Autosave** is selected, dynamically calculated cells that depend on the changed and saved cell values (for example, a row that has a member formula that calculates a percentage of an aggregated parent value) are also updated and are displayed with a green background .

3   Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Setting Dimension Properties

You can set and edit form dimension display properties, including whether to display the member name or alias in the form, hide the row or column, and permit users to view the member formula. 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" on page 86.

2   Click in a point of view, page, row, or column to set dimension properties.

3   Select **Dimension Properties**:

Table 12   Dimension Properties

Property	Description
Apply to all row dimensions	Applies properties to all row dimensions
Apply to all column dimensions	Applies properties to all column dimensions
Apply to all page dimensions	Applies properties to all page dimensions
Apply to all POV dimensions	Applies properties to all point of view dimensions
Member Name	Displays the member name
Alias	Displays the member alias

Property	Description
Member Formula	Displays member formulas
Hide dimension	Hides the dimension
Show consolidation operators	Displays consolidation operators
Start expanded	Available only for dimensions on row or columns, choosing this option initially displays the dimension member list expanded.
Enable custom attributes	Available only for dimensions on row or columns, enables custom attributes.

4   Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Setting Display Properties

You can set and edit options for form display, such as hiding forms or displaying missing values as blank, in the **Layout** tab.

You can also enable account-level annotations. Users can add annotations to accounts in forms if 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.

Notes:

- The Account dimension must be assigned to a row axis.

- Account, Entity, Versions, and Scenario dimensions cannot be assigned to the column axis.

- The Entity dimension can be assigned to the row, page, or Point of View axis.

- Version and Scenario dimensions must be assigned to the page or Point of View axis.

➤ To set display properties:

1   Open the form, and then click **Layout**.

See .

2   Select **Display Properties**, and then select form options:

Table 13   Display Properties

Option	Description
Make form read-only	You cannot set this option for composite forms.
Hide form	For example, hide forms that are part of composite forms or are accessed from menus or task lists.
Display missing values as blank	Leave form cells empty where data does not exist. If this option is not selected, empty cells display the text "#MISSING."
Enable account annotations	This option is available only if the Account dimension is on the row. For information about using account annotations, see the *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

Option	Description
Allow multiple currencies per entity	If the application supports multiple currencies, allow entities to support multiple currencies, regardless of base currency. Users can select currency for displayed cell values in forms.
Enable Mass Allocate	Users must have the Mass Allocate role to use this option. For information about using Mass Allocate, see the *Working with Planning for Oracle Planning and Budgeting Cloud Service*.
Enable Grid Spread	For information about using Grid Spread, see *Working with Planning for Oracle Planning and Budgeting Cloud Service*.
Enable cell-level document	(Default) Enable users to add, edit, and view documents in cells in the form, depending on access permissions. To prevent users from using documents in a form, clear this option. To use cell documents, see *Working with Planning for Oracle Planning and Budgeting Cloud Service*, Chapter 3, "Setting Up Access Permissions."
Message for forms with no data	Enter text to display in form rows for queries without valid rows. Leave blank to display the default text, `There are no valid rows of data for this form`.

3 Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Setting Printing Options

You can set and edit preferences for printing form information in the **Layout** tab.

➤ To set printing options:

1 Open the form, and then click **Layout**.

See "Selecting and Opening Forms and Folders" on page 86.

2 Select **Printing Options**, and then set preferences for printing form information:

Table 14 Printing Options

Option	Description
Include supporting detail	Include supporting detail as extra rows in PDF files. Specify display format: **Normal Order**: Prints supporting detail in the same order as on the Supporting Detail page, after the member it is associated with **Reverse Order**: Prints supporting detail in 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 text notes associated with cells
Format data	Apply number format settings from the form to the displayed data
Show attribute members	If attribute members are selected in the form, display them in PDF files
Apply precision	Apply form precision settings (desired number of decimal points) to the displayed data in PDF files

Option	Description
Show currency codes	If the form supports multiple currencies, display currency codes in the form and in PDF files. Whether currency codes display depends on whether currency codes are present on any member in the form. If a currency code is present on any member contained in the form, currency codes display in the form regardless of the selection for this check box. If currency codes are not present on members in the form, currency codes are not displayed.
Show account annotations	If account annotations are enabled for the form, select to display account annotations in PDF files

3   Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Including Data Validation Rules in Forms

In the **Layout** tab, you can add and update validation rules to the grid, column, row, or cell. When rules are processed, they can change the color of cells, provide validation messages to users during data entry, and can change the promotional path for planning units. Validation rules are saved with the form.

Before adding data validation rules, it is important to consider the function the rule will perform, and to plan the rule scope. For detailed information and best practices for planning and implementing validation rules, see Chapter 8, "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" on page 86.

2   Select **Validation Rules**, and then select an option:

Note: The menu options that display are context-sensitive, and depend on whether rules have already been added, and if you have previously selected a menu option. For example, if you right-click a cell that contains a rule and select **Copy Validation Rules**, the **Paste Validation Rules** menu option is displayed when you right-click another cell.

Table 15   Validation Rules Options

Option	Description
Add/Edit Validation Rules	Add or edit existing rules in the condition builder area of the **Data Validation Rule Builder** dialog box.
Copy Validation Rules	Copy the selected rules to be pasted to a new location.
Paste Validation Rules	Paste the previously copied rules to a new location.
Validate only for users with access to this form	If the currently logged in user does not have access to the form, do not execute validations associated with the form when validating the planning unit.

Option	Description
Validate only for pages with existing blocks	When enabled, Planning intelligently 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.
Validate only for cells and pages the user has access to	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.

3   Build and validate the rules as described in "Creating and Updating Data Validation Rules" on page 113.

4   In the form, click **Next** to continue building the form, and then validate and save the form.

Setting Form Precision and Other Options

In **Other Options**, you set data precision, associate context menus with the form, and enable dynamic user variables.

You 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.

➤   To set form precision and other options:

1   Open the form, and then click **Other Options**.

See "Selecting and Opening Forms and Folders" on page 86.

2   In **Precision**, select options to set 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 longer numbers. For example:

Table 16   Data Precision Examples

Value	Minimum Precision	Maximum Precision	Displayed Value
100	0	Any	100
100	3	Any number greater than or equal to 3 or None	100.000
100.12345	Any number less than or equal to 5	None	100.12345
100.12345	7	None	100.1234500
100.12345	Any number less than or equal to 3	3	100.123
100.12345	0	0	100

Value	Minimum Precision	Maximum Precision	Displayed Value
100.12345	2	4	100.1234
100	2	4	100.00

Notes:

- By default, the precisions settings you select here override the precision set for the currency member (see "Creating Currencies" on page 237). If instead you want the currency member's precision setting to prevail for the form, select **Use Currency member precision setting**.

- Precision settings affect only the display of values, not their stored values, which are more accurate. For example, if **Minimum Precision** is set to 2, and if Planning spreads the value 100 from Q1 into the months Jan, Feb, and Mar, the month cells display 33.33 when they are not selected. When they are selected, they display their more accurate 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 back to Q1, 33.33333333333333 is multiplied by 3 and Q1's value displays 99.99999999999998 when you click into Q1.

3 In **Context Menus**, associate menus with the form by selecting them from **Available Menus** and moving them to **Selected Menus**:

- ❯ moves selections
- ❯❯ moves all
- ❮ removes selections
- ❮❮ removes all

4 If you select multiple menus, click one of the following options to set the order in which they display:

- ⤒ moves the selection to the top of the order
- ▲ moves the selection up one spot in the order
- ▼ moves the selection down one spot in the order
- ⤓ moves the selection to the bottom of the order

Multiple menus display sequentially, with separators between them.

5 Select **Enable Dynamic User Variables** to allow dynamic user variables in the form (see *Working with Planning for Oracle Planning and Budgeting Cloud Service*).

6 Click **Save**.

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

Watch this video to learn about creating asymmetric rows and columns in Planning.

▶ Video

➤ To create asymmetric rows or columns:

1   Open the form, and then click **Layout**.

    See "Selecting and Opening Forms and Folders" on page 86.

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 Chapter 7, "Using the Member Selector."

4   Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Adding Formula Rows and Columns

Formula rows contain formulas that apply to form rows. Formula columns contain formulas that apply to form columns. For example, you can create a formula column (column D) that computes the percentage variance between the January sales (column A) and February sales (column B). The formula defined for a formula row or column applies to all row or column dimensions. To define or assign existing formulas to forms, select the appropriate row or column on the **Layout** tab and then display formula building options under **Segment Properties** (see "Creating Formulas " on page 269).

Tip: Consider adding a formula row between two other rows to create a blank row. Blank rows are useful, for example, for visually separating subtotals and totals within a form.

➤ To add formula rows and columns:

1   Open the form, and then click **Layout**.

    See "Selecting and Opening Forms and Folders" on page 86.

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** that is displayed in the row or column, and then enter the formula name.

5   Click the row or column number and specify any of the following displayed in the **Segment Properties** pane to the right:

    ● **Hide** hides the row or column

    ● **Show separator** displays the row or column separator

- **Display formula on form** displays the formula on the form when you click ![icon] in the row or column header.

6 For each dimension in **Formula Data Type** in the right pane, select a data type for the formula result.

The data types are:

- Currency

- Non Currency

- Percentage

- SmartList

  If you select SmartList, select a Smart List from the drop-down list next to the data type.

- Date

- Text

7 Define the formula to use for the row or column by entering the formula name in the **Formula** field, and then clicking ![pencil icon] . See "Editing Formulas " on page 269.

8 Click **Validate** to ensure that the formula does not contain any errors.

9 Click **OK** to save the formula and to close the **Formula** window.

Defining Simple Form Page and Point of View

You can select dimensions and members for the page axis and point of view. Point of view dimensions and members must be valid for the form plan type and not assigned to a page, column, or row axis. The point of view sets the unique dimension members that define intersections of data.

When you set user variables for forms, the variable name displays in point of view. See "Managing User Variables" on page 100.

➤ To define page axis and point of view:

1 Open the form, and then click **Layout**.

See "Selecting and Opening Forms and Folders" on page 86.

2 Click ![icon] and then drag that dimension to **Page** to add it to the form page axis.

3 Click ![icon] for each page axis dimension and select members. See Chapter 7, "Using the Member Selector."

You can specify the number of members in a page dimension that enables a search option. Select **File**, then **Preferences**, and then **Planning**. In **Display Options**, type a value for **Allow Search When Number of Pages Exceeds** and select that option. Click **Save**.

4 Repeat Steps 2 and 3 to assign multiple dimensions to the page axis.

Assigning multiple dimensions to the page axis enables planners to select dimensionality while entering data. Users can select **Display Options** to specify whether Planning sets the page selection to the most recently used selection.

5   In **Dimension Properties**, select or clear options for page dimensions. See "Setting Dimension Properties" on page 64.

6   **Optional**: Click [icon] and then drag that dimension to **Point of View** to add it to the form point of view. Repeat this action for each dimension you want to move to **Point of View**.

7   In **Point of View**, click [icon] for each dimension and then select members.

See Chapter 7, "Using the Member Selector."

8   In **Dimension Properties**, select or clear options for point of view dimensions. See "Setting Dimension Properties" on page 64.

9   Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Creating Composite Forms

Subtopics

● Setting Composite Form Layout
● Setting Composite Form Section Properties
● Setting Composite Form Point of View and Page Dimensions
● Creating Master Composite Forms
● Embedding Charts in Composite Forms

Composite forms display several forms simultaneously, even those associated with different plan types. Users can enter data and see results aggregated to an upper-level intersection, such as Total Revenue.

Watch this video to learn about creating composite forms in Planning.

[icon] Video

➤   To create composite forms:

1   Select **Administration**, then **Manage**, and then **Forms and Ad Hoc Grids**.

2   Select the folder in which to store the form. See "Selecting and Opening Forms and Folders" on page 86.

3   Click **Create Composite Form** above the list of forms.

4   In the **Properties** tab, enter a form name of up to 80 characters, and an optional description of up to 255 characters.

5   **Optional**: Select **Hide Form** to hide the form.

6   **Optional**: Enter instructions for the form.

7   Set the composite form layout. See "Setting Composite Form Layout" on page 73.

8   Set the composite form section properties. See "Setting Composite Form Section Properties" on page 75.

9   Set the composite form Point of View and Page display options. See "Setting Composite Form Point of View and Page Dimensions" on page 77.

10   Click **Save** to save your work and continue, or click **Finish** to save your work and close the form.

Setting Composite Form Layout

Planning provides tools that allow you to 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 two sections that are stacked one above the other. There is also a custom layout option.

➤ To set composite form layout:

1   Open the composite form, and then click **Layout**.

     See "Selecting and Opening Forms and Folders" on page 86.

2   In **Select Layout**, select an option:

     • **Custom Layout** to create your own composite form layout.

       Note: The custom layout option is selected by default.

     • **2–Row Layout** to split the composite form into two sections, one on top of the other, divided by a horizontal line.

     • **2–Column Layout** 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" on page 74, "Rearranging Forms in the Layout" on page 75, and "Deleting Simple Forms from a Composite Form" on page 75.

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.

Note: When you split a composite form section that contains simple forms, the simple forms remain in the original section. For example, if you split a section vertically, the original section is divided into two side-by-side sections. The simple forms from the split section are included in the left section, and the right section is empty.

- **Delete** to remove a section from the composite form.

  When you delete a section from a composite form, the simple forms included in that section are also deleted from the composite form, unless they are included in other sections of the composite form.

- **Add Form** to display the Form Selector dialog box where you can select additional forms to add to the layout.

- **Group as Tabs** to display the forms in that section as tabs.

- **Ungroup Tabs** to clear **Group as Tabs**.

5 Click **Save** to save the Composite Form layout.

Tip: You can edit a simple form from within a composite form. Right-click the simple form, and then select **Form Designer**. Edit the form as described in "Editing Forms" on page 88. Access permissions apply as described in "Forms and Access Permissions" on page 59.

Adding Simple Forms to a Composite Form Layout

➤ To add a simple form to a section in a 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 are 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, can 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

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 is not available for ad hoc forms. Access permissions apply as described in "Forms and Access Permissions" on page 59.

➤ 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" on page 88.

Deleting Simple Forms from a Composite Form

➤ To delete a simple form from a 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 creation. You can edit these properties after creating a composite form.

➤ To set composite form properties:

1 Open the composite form, and then click **Layout**.

    See "Selecting and Opening Forms and Folders" on page 86.

2 Expand **Section Properties**.

3 Click in a composite form section and set the properties as desired.

Table 17   Composite Form Section Properties

Option	Description
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	Section height. Select: ● **Automatic** to have Planning set the height. ● **%** (percentage sign) to set section height to a percentage of the composite form height.
Width	Width of the section. Select: ● **Automatic** to have Planning set the width. ● **%** (percentage sign) to set section width to a percentage of the composite form width.
Forms per Row	Select: ● **Automatic** to have Planning set the number. ● Select a number from 1 to 20. The default is one form per row. If **Forms per Column** is set to a value other than **Automatic**, **Forms per Row** is set to **Automatic**. **Note:** If you have grouped the forms as tabs, this option is not available.
Forms per Column	Select: ● **Automatic** to have Planning set the number. ● Select a number from 1 to 20. The default is one form per column. If **Forms per Row** is set to a value other than **Automatic**, **Forms per Column** is set to **Automatic**. **Note:** If you have grouped the forms as tabs, this option is not 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.

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

  - ❍ Version: Working

  - ❍ Currency: Local

  - ❍ Year: No Year

  - ❍ Period: Beginning Balance

  - ❍ Asset Class: Computers

  - ❍ Line Item: Base SP1

- A simple form called "Cash Flow Impact".

In the master composite form, the user highlights the row `Computers/Base SP1`.

Figure 1   Master Composite Form: "New Computers"

	A	B	C	D	E	F	G	H	I	J
1			Asset Description	Asset Units	Asset Rate	Basic Cost	Salvage	Purchase Date	In Service Date	Premature E
2	Furniture and Fixtures	Base SP1	Office Furniture	0	1,000	1,000	0	1/2/2009	1/2/2009	12/
3		Base SP2	Office Desktops	0	100	100	5	1/2/2009	1/2/2009	
4	Computers	Base SP1	Laptops	0	50	250	0	1/2/2009	1/2/2009	
5	Tangible Assets	Total Specified				1,250				

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 2   Simple Form: "Cash Flow Impact"

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1		-YearTotal	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec		
2	Cash Outflow from Capital Additions	250	250													
3	-Net Cash Flows	-250	-250													

➤   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" on page 86.

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 design 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.

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.
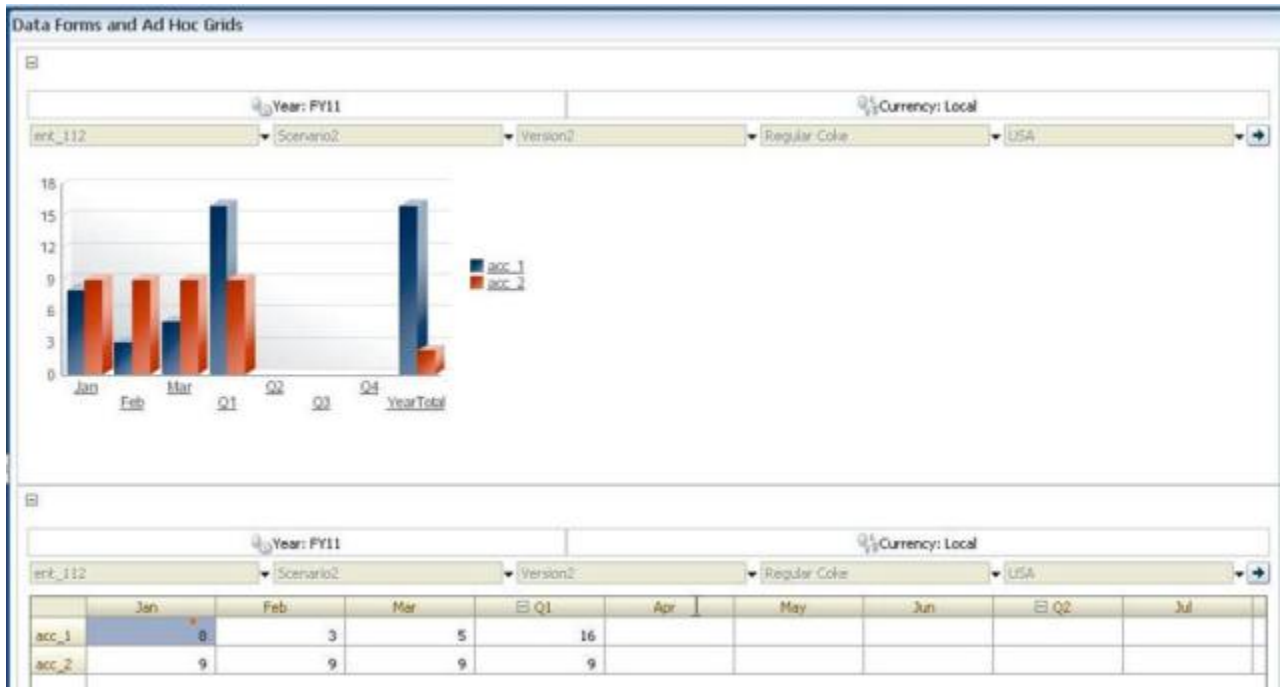
- Include the same ad hoc grid twice, one to display as a grid and the other to display as a chart. Users can then perform ad hoc operations (such as **Zoom In**, **Pivot To**, and **Keep Only**) on the grid and view the changes in the chart.

- Create dashboards. For example:
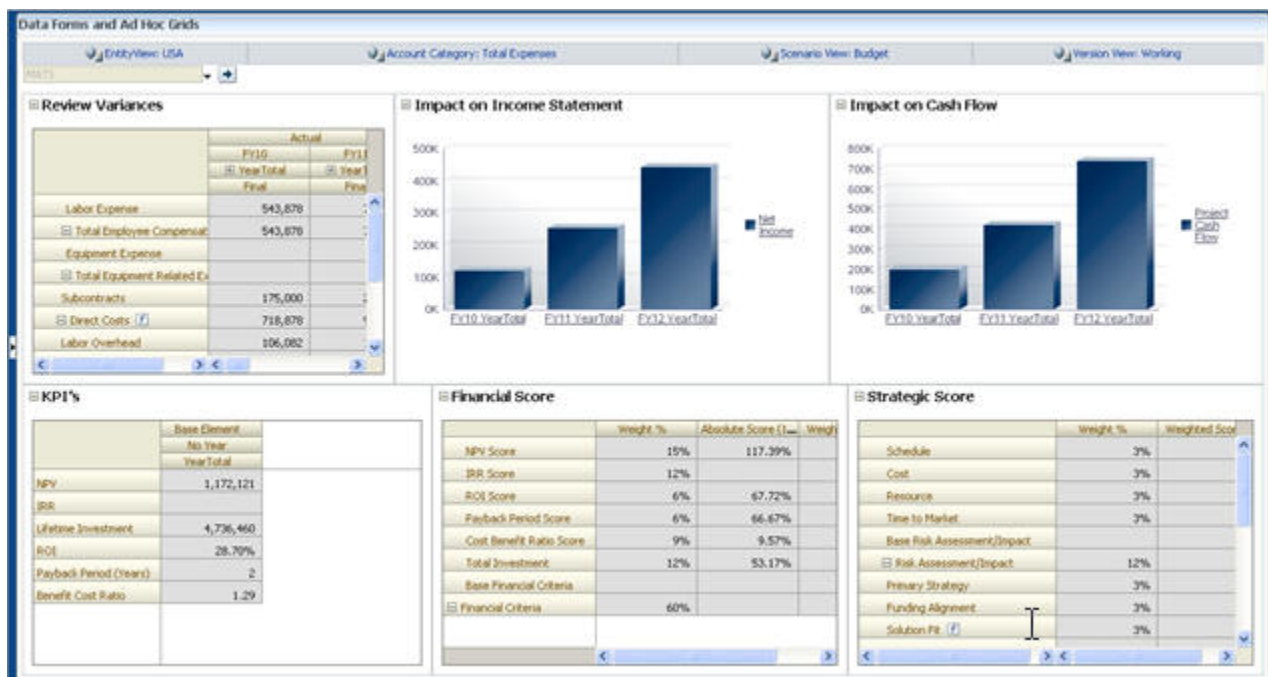
Watch this video to learn about displaying forms as charts in Planning.

▶ Video

➤ To embed charts in composite forms:

1   Create or edit the composite form, and then click **Layout.**

    See "Selecting and Opening Forms and Folders" on page 86.

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:

Table 18   Chart Types

Chart Type	Description
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.

Designing Specific Types of Forms

Subtopics

- Designing Forms for Multiple Currencies
- Designing Forms with Formula Rows and Columns
- Designing Forms with Data Validation
- Designing Forms with Global Assumptions
- Designing Forms for Rolling Forecasts

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" on page 70. For information about creating formulas and using the mathematical functions, see Appendix B, "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" on page 67 and Chapter 8, "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 Application Management, 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 Application Management, this enables migrating the driver data contained in the tagged form. Note that global assumptions cannot be used with composite forms or forms with multiple segments.

Designing Forms for Rolling Forecasts

Subtopics

- 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 3   12-Monthly Rolling Forecast

Year and Period in Columns With No Additional Segment

	FY12 Jul	FY12 Aug	FY12 Sep	FY12 Oct	FY12 Nov	FY12 Dec	FY12 Jan	FY12 Feb	FY12 Mar	FY12 Apr	FY12 May	FY12 Jun	FY13 Jul	FY13 Aug	FY13 Sep	Total 12 month rolling
Actual	50	50	50													
Plan/Budget	200	200	200	200	200	200	200	200	200	200	200	200	300	300	300	
12 month Rolling Aug	50	75	75	75	75	75	75	75	75	75	75	75	125			
12 month Rolling Sep	50	50	75	75	75	75	75	75	75	75	75	75	125	125		
12 month Rolling Oct	50	50	50	75	75	75	75	75	75	75	75	75	125	125	125	

Figure 4   Quarterly Rolling Forecast

	FY12 Q2	FY12 Q3	FY12 Q4	FY13 Q1	FY13 Q2	FY13 Q3	FY13 Q4
FY12 Q2 Review	F	F	F	F			
FY12 Q3 Review	A	F	F	F	F		
FY 12 Q4 Review	A	A	F	F	F	F	
FY13 Q1 Review	A	A	A	F	F	F	F

Figure 5   Quarterly Trailing Rolling Forecast (Rolling Quarters with a Cumulative Total)

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	12 qtrs rolling
Project 1	25	35	45	55	65	75	85	95	105	115	125	135	960
Project 2	50	60	70	80	90	100	110	120	130	140	150	160	1260
Project 3	75	85	95	105	115	125	135	145	155	165	175	185	1560
Project 4	100	110	120	130	140	150	160	170	180	190	200	210	1860
Project 5	125	135	145	155	165	175	185	195	205	215	225	235	2160

Figure 6   Rolling Forecast Where There are Additional Segments for Actual and Plan Year

Year and Period in Columns									12 month Rolling Aug							Actual	Plan
	FY12	FY12	FY12	FY12	FY12	FY12	FY12	FY12	FY12	FY12	FY12	FY12	FY13	FY13	FY13	FY12	FY13
	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	YearTotal	YearTotal
Account 1	50	75	75	75	75	75	75	75	75	75	75	75	125			600	3600

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 "Creating Simple Forms" on page 60.

2   **In the Layout tab, drop the Year 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 Year and Period are in the same grid axis (row or column).

4   **In the Rolling Forecast Setup dialog box, enter the following information:**

Table 19   Rolling Forecast Setup Options

Option	Description
Prefix	Allows rolling forecast substitution variables to be distinct from other substitution variables; for example, **4**QRF designates that the forecast is a 4-quarter rolling forecast.
Reuse existing substitution variables	Select if you wish to specify a prefix that you have previously used.

Option	Description
Start Year	The year in which the rolling forecast starts; for example, FY11. Either enter the start year or click [icon] to open the Member Selection dialog box. 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.
Start Period	The period in the year in which the rolling forecast starts; for example, Q1. Either enter the start period or click [icon] to open the Member Selection dialog box. 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.
Number of Periods	Number of year/period combinations that will be generated as separate segments.

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.

Notes:

- Substitution variables are based on the periods selected for the Start Year and Start Period. See "About Selecting Substitution Variables as Members" on page 110.

- 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

Administrators can revise the values for rolling forecast substitution variables directly in the form.

➤ To modify rolling forecast variables in a form:

1   Select **Administration**, then **Manage**, and then **Forms and Ad Hoc Grids**.

2   Open the rolling forecast form.

3   Right-click any column in the form and select **Set Rolling Forecast Variables**.

4   In the **Set Rolling Forecast Variables** dialog box, enter or edit values for the Year 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 Year 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

Subtopics

- Selecting and Opening Forms and Folders
- Previewing Forms
- Printing Form Definitions
- Searching for Forms
- Editing Forms
- Moving, Deleting, and Renaming Forms
- Using Grid Diagnostics
- Using Business Rules
- How Cell Formatting in Smart View Persists in Planning
- Working with Substitution Variables
- Working with User Variables

Selecting and Opening Forms and Folders

Use these procedures to select and open form folders and the forms they contain. For ease of editing, administrators can open non-ad hoc forms for editing directly from the end-user interface.

➤ To select and open non-ad hoc forms from within the end-user interface:

1   Open the data form.

2   Click the **Form Designer** icon at the top of the page.

The form opens in edit mode in a new tab.

➤ To select and open forms or form folders from within the administrator's interface:

1   Select **Administration**, then **Manage**, and then **Forms and Ad Hoc Grids**.

2   Perform one of the following steps:

- To open a form folder, select a form folder beneath **Form Folders**.

- To open a form, select a form from the list displayed beneath **Form** when the appropriate form folder is open.

After you select a form folder, use the buttons next to **Form 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 for Oracle Planning and Budgeting Cloud Service.*

Previewing Forms

While you are 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 cannot 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 do not 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 Select **Tools**, and then **Reports**.

2 Select **Forms**.

3 Select the form definitions to print by moving them from **Available Forms** to **Selected Forms**:

- To add one or more selected forms, click .

- To add all forms, click ⧉.

- To remove one or more forms, click ◁.

- To remove all forms, click ◁◁.

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:

- Plan type

- 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   Select **Administration**, then **Manage**, then **Forms and Ad Hoc Grids**.

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

Subtopics

- Editing Simple Forms
- Editing Composite 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.

Editing Simple Forms

➤ To edit simple forms:

1 Select the form, then click **Edit** (see "Selecting and Opening Forms and Folders" on page 86).

2 Select:

    a. **Properties** to edit the form name, description and instructions. See "Creating Simple Forms" on page 60.

    b. **Layout** to edit form layout. See "Setting Form Layout" on page 61.

    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" on page 68.

    d. **Business Rules** to change which business rules are associated with the form, or modify business rule properties. See "Using Business Rules" on page 91.

3 Click **Finish** to save your work and close the form.

Editing Composite Forms

➤ To edit composite forms:

1 Take one of these actions:

- Select the form, click the **Show Usage** icon 🔲, and then click **Edit**.

- Select the form, and then click **Edit** (see "Selecting and Opening Forms and Folders" on page 86).

    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.

2 Select:

- **Properties** to edit the composite form name, description or instructions. See "Creating Composite Forms" on page 72.

- **Layout** to edit form layout and properties. See "Setting Composite Form Layout" on page 73.

- **Business Rules** to change which business rules are associated with the form, or modify business rule properties. See "Using Business Rules" on page 91.

3 Click **Finish** to save your work and close the form.

Moving, Deleting, and Renaming Forms

➤ To move, delete, and rename forms:

1 **Select the form.**

   See "Selecting and Opening Forms and Folders" on page 86.

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.**

Using Grid Diagnostics

Grid diagnostics allow you to view the time it takes to open forms and ad hoc grids. You can select whether to view the load times in chart form or tabular form, and you can select on which forms and grids to run the diagnostics. This helps you to identify forms with poor performance and address design issues in forms.

➤ To use grid diagnostics:

1 **Log in to Planning.**

2 **Select Tools, then Diagnostics, and then Grids.**

3 **Select the forms on which you want to run the diagnostics and click Run Diagnostics.**

   A pie chart is displayed in **Diagnostics Summary** that shows the percentage of load times for forms and ad hoc grids that are greater than five seconds, between one and five seconds, and less than one second.

4 **Click on a section of the pie chart to display more detail about a specific load time in the Display area under Diagnostics Summary**

   For example, clicking Load Time (Greater than 5s) on the pie chart displays details about the forms with load times greater than five seconds in the Display area.

5 **Once you have selected the load time detail to display, select whether to display the information in Tabular format or as a Chart.**

   If you select to display the load times as a chart, select the chart type: Area, Horizontal Bar, Bar, or Line.

Using Business Rules

Subtopics

- About Business Rules
- Selecting Business Rules
- Setting Business Rule Properties
- About Runtime Prompts
- Understanding Runtime Prompts
- About Runtime Prompts and Approvals Security
- Designing Secure Runtime Prompts

About Business Rules

With appropriate access, users can launch business rules from Planning. Business rules can also prompt users for input when rules are launched.

For optimum performance, business rules that run in forms should be designed to execute within three minutes. For business rules with longer execution time, you can schedule batch processing or run the business rules during non-peak hours.

For information on:

- Creating and updating business rules, see *Designing with Calculation Manager for Oracle Planning and Budgeting Cloud Service.*

- Using runtime prompts, see "About Runtime Prompts" on page 93.

Selecting Business Rules

On the **Business Rules** tab, you can associate multiple business rules with a form, by plan type. Users 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.

Watch this video to learn about selecting business rules in Planning.

▶ Video

➤ To select business rules for forms:

1   Take an action:

- To update the current form, click **Business Rules**.

- To open a form for editing, click **Edit**, and then click **Business Rules**.

2   From the **Plan Type** drop-down menu, select the plan type.

3   From the **Business Rules** list, select the business rules to associate with the form, and move them to **Selected Business Rules**. See Chapter 7, "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 clear Calculate Currencies if you use customized calculation scripts to calculate currency conversions. You can clear Calculate Form to prevent planners from calculating data in forms.

4   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.

The order in which business rules launch is important and may affect data. For example, it is important to convert currencies first, before subtotaling.

5   To set business rule properties, click **Properties**. See "Setting Business Rule Properties" on page 92.

6   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 do not 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 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 **Business Rules**.

  ● To open a form for editing, click **Edit**, and then click **Business Rules**.

2   Click the **Business Rules** tab.

3   Select **Run on Load** next to a business rule to launch it automatically when the form is opened.

Business rules having runtime prompts cannot launch on load.

4   Select **Run on Save** next to a business rule to launch it automatically when saving a form.

If business set to **Run on Save** contain runtime prompts, users are prompted to enter the runtime prompt value before the form is saved.

5   **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" on page 93.

6 **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

7 Click **OK**.

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 as described in "About Dynamic Members" on page 220, then users can create new members by entering their name in the runtime prompt.

When launching business rules with runtime prompts, Planning validates the value entered, but not the business rule. To set the default member selection in a runtime prompt, see "Setting Business Rule Properties" on page 92. To understand how other settings and conditions affect runtime prompts, see "Understanding Runtime Prompts" on page 93.

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" on page 92)

● Whether the **Runtime Prompt** option is set during form design or set when designing the runtime prompt (see *Designing with Calculation Manager for Oracle Planning and Budgeting Cloud Service*)

● 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 as described in "About Dynamic Members" on page 220. 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 the **Tools**, then **Business Rules** menu), and regardless of whether the runtime prompt is hidden during design. The **Override Value** can be set as a Planning 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 is not 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 **Tools**, then **Business Rules** 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** is not selected

   - A runtime prompt is launched from the **Tools**, then **Business Rules** menu

   - Values cannot 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 is 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 is not 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 20 How Member Availability and Other Settings Affect Runtime Prompts

Availability of Override Value and member on the Page/Point of View	Use Members on Form option is selected	Hide Runtime Prompt property is set during runtime prompt design	Hide Prompt option is selected for the form	Result on Runtime Prompt
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.	Yes	Yes	Yes or No Setting is ignored	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.
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.	Yes	No	Yes	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 cannot 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.
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.	Yes	No	No	The runtime prompt is displayed to users, with values pre-filled from the **Override Value** or Page/Point of View.
Use as Override Value is not set and the member is not available on the Page/Point of View to use as the runtime prompt value.	Yes	Yes	Yes or No Setting is ignored	The business rule displays the runtime prompt to users, with values pre-filled according to Principle 3. For example, the form context cannot 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.
Use as Override Value is not set and the member is not available on the Page/Point of View to use as the runtime prompt value.	Yes	No	Yes	The runtime prompt is displayed to users, with values pre-filled according to Principle 3.

Availability of Override Value and member on the Page/Point of View	Use Members on Form option is selected	Hide Runtime Prompt property is set during runtime prompt design	Hide Prompt option is selected for the form	Result on Runtime Prompt
Use as Override Value is set and **Override Value** is available, and the member is not available on the Page/Point of View to use as the runtime prompt value.	Yes	No	No	If all runtime prompts can be pre-filled from the **Overide Value** and are valid and within limits, the runtime prompts are not displayed. However, if even one runtime prompt value cannot be pre-filled from the **Overide Value**, then all runtime prompts display, with values pre-filled wherever possible. All others follow Principles 1 and 3.
Use as Override Value is not set and the member is not available on the Page/Point of View to use as the runtime prompt value.	Yes	No	No	The runtime prompt is displayed to users, with values pre-filled according to Principle 3.
Use as Override Value is set and **Override Value** is available, and the member is not available on the Page/Point of View to use as the runtime prompt value.	Yes	No	No	The runtime prompt is displayed to users, with values pre-filled according to Principles 1 and 3.
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.	No	Yes	Not available	The business rule runs without displaying the runtime prompt to users. Instead, the design-time values are used.
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.	No	No	Not available	The runtime prompt is displayed to users, with values pre-filled according to Principle 3.
Use as Override Value is not set and the member is not available on the Page/Point of View to use as the runtime prompt value.	No	Yes	Not available	The business rule runs without displaying the runtime prompt to users. Instead, the design-time values are used.
Use as Override Value is not set and the member is not available on the Page/Point of View to use as the runtime prompt value.	No	No	Not available	The runtime prompt is displayed to users, with values pre-filled according to Principle 3.

When hidden runtime prompt values are ambiguous, note:

- If the form context cannot 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 cannot 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 is 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 does not apply to Calculation Manager rulesets.

- When launched from the **Tools**, then **Business Rules** 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 planning units to which they do not have access, according to Approvals rules. For example, the administrator may not want planners to change data after they have promoted the related planning unit. In Calculation Manager, administrators can set runtime **Security** for a member or members:

- **Approvals**: Planning allows a user to change member data if both these conditions are true:
  - The user has write access to the members (as assigned in Planning).
  - If the members belong to a planning unit, the user owns the planning unit.

    If both conditions are not met, the user cannot change the members' data.

- **Write**: Users launching the business rule who have write access to the members (as assigned in Planning) 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 Planning). Approvals status is ignored.

- **Use Default**: Security is applied to the runtime prompt only if member access is not set to None (that is, either read or write).

See .

Designing Secure Runtime Prompts

Planning 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 does not 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 a planning unit dimension from the list of runtime prompts (for example, the runtime prompt does not include a Scenario or Entity), then Approvals security is not applied and the hierarchy is filtered by write access.

See "About Runtime Prompts and Approvals Security" on page 97.

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 Planning's formatting

- **Custom Styles**: To use the formatting saved in Smart View

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 for Oracle Planning and Budgeting Cloud Service*.

- On saving Excel formatting, see the *Working with Smart View for Oracle Enterprise Performance Management Cloud Services*.

Working with Substitution Variables

Subtopics

- About Substitution Variables
- Creating and Assigning Values to Substitution Variables Using Planning
- Deleting Substitution Variables Using Planning

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 variables within Planning. These substitution variables are then available in Planning when you select members for a form.

For more information about selecting substitution variables, see "About Selecting Substitution Variables as Members" on page 110.

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

Creating and Assigning Values to Substitution Variables Using Planning

➤ To create and assign values to substitution variables using Planning:

1 Select **Administration**, then **Manage**, then **Variables**.

2 Select the **Substitution Variables** tab.

3 Click **Actions**, then select **Add**.

4 On the **Add Substitution Variable** page, select the **Plan Type**.

5 For **Name**, enter the name of the substitution variable.

6 For **Value**, enter a value for the substitution variable.

7 Click **OK**.

Deleting Substitution Variables Using Planning

➤ To delete substitution variables using Planning:

1   Select **Administration**, then **Manage**, then **Variables**.

2   Select the **Substitution Variables** tab.

3   Select the substitution variable to delete.

4   Click **Actions**, then select **Delete**.

5   Click **Yes**.

Working with User Variables

Subtopics

- About User Variables
- Managing User Variables
- Creating User Variables
- 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 is a dynamic user variables. 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.

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 "About Selecting User Variables as Members" on page 111.

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, and select user variables for any axis in the form. See "Defining Simple Form Page and Point of View" on page 71.

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.

    See "Creating User Variables" on page 101.

3. When designing the form, associate the user variable with the form.

   See .

4. Instruct users to select a member for the user variable associated with the form.

   Before users can open forms that have user variables, they must select a member for User Variable Options in preferences. After selecting an initial value, they can change it in the form or in preferences. See *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

Creating User Variables

Watch this video to learn about creating user variables in Planning.

▶ Video

➤ To create user variables:

1 Select **Administration**, then **Manage**, then **Variables**.

2 Select the **User Variables** tab.

3 Click **Actions**, then select **Add**.

4 In the **User Variables** window, for **Dimension Name**, select the dimension for which to create a user variable.

5 For **User Variable Name**, enter the name of the user variable.

6 **Optional**: Select **Use Context** to allow user variables to be used in the Point of View. With this setting, the value of the user variable changes dynamically based on the context of the form.

7 Click **OK**.

   You can now associate the user variable with a form. See . Afterward, planners can select members for the user variable. See *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

Deleting User Variables

➤ To delete user variables:

1 Select **Administration**, then **Manage**, then **Variables**.

2 Select the **User Variables** tab.

3 Select the user variable to delete.

4 Click **Actions**, then select **Delete**.

5 Click **Yes**.

7

Using the Member Selector

About Working with Members

Use the Member Selection dialog box to select members and other information to use with features such as forms and business rule runtime prompts. If variables and attributes are defined, you can also select variables and attributes. You can display and select members by member name, alias, or both. 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.

Selecting Members

Use the Member Selection dialog box to select members and other information for Planning features. For example, you can select members for form management, business rule runtime prompts, and Clear Cell Details. You can select members by member name, alias, or both, depending on the setting for the member in the dimension editor for the current application and member selection options set by users.

➤ To select members:

1 From the members list, select members.

2 **Optional:** Select or enter search criteria.

    a.   In the Search box, select options: **Member Name**, **Member Alias**, **Description**, or **UDA**.

        You can also enter all or part of a member name, alias, description, or UDA. Searches include shared members. To search for an exact match, select **Exact** and enter the search

string. The search is not case-sensitive. You can search for a word, multiple words, or wildcard characters. See "Using Wildcards in Searches" on page 106.

    b.   Click .

3   **Optional**: To set how information displays or filter members in the left pane of the Member Selection dialog box for the current session, click one of the following icons above the members list. You can select multiple filters, such as functions and generations.

- Display Properties. Select **Member Name**, **Alias**, **Member Name: Alias**, or **Alias: Member Name**. Or, select **Additional Display Properties**, then **Description** or **Count**. Selected options are displayed in columns in the left and right panes. To remove columns, clear the selections.

- Keep only by Functions. Select **Member**, **Descendants**, **Descendents (inc)**, **Ancestors**, **Ancestors (inc)**, **Siblings**, **Siblings (inc)**, **Parents**, **Parents (inc)**, **Children**, **Children (inc)**, or **Level 0 Descendants**.

- Keep only by Attributes. If attributes are defined, select options for **Attributes**, **Operator**, and **Value**, and then click **OK**. Multiple attributes can be selected.

- Keep only by levels or generations. Select **Level** or **Generation**, select the levels or generations to display, and then click **OK**.

- Refresh the display to remove filters.

4   **From the members list, make a selection.**

Only members to which you have access are displayed.

Notes:

- **For runtime prompts only**: The displayed hierarchy is for the application and plan type against which the business rule is launched. For runtime prompts, members are displayed that satisfy runtime prompt limits.

- **For business rules or validation rules having Cross Dimension or Member Range runtime prompts only**: From the **Select Dimension** list, select a dimension that the business rule designer set for this runtime prompt.

- **For shared members only**: Shared members are displayed in this format: *shared_member.parent_member*(shared). When they are selected on the right side of the dialog box, only the shared member name is displayed.

5   **Optional**: Click or to collapse or expand the hierarchy.

6   **Move members to or from the Selected Members list.**

Click in the middle of the dialog box to select a range of members based on hierarchy relationships.

Table 21 Member Relationships

Relationship	Members Included
Member	The selected member
Descendants	All descendants of the selected member, excluding the selected member
Descendants (inc)	The selected member and its descendants
Ancestors	All members above the selected member, excluding the selected member
Ancestors (inc)	The selected member and its ancestors
Siblings	All members from the same level in the hierarchy as the selected member, excluding the selected member
Siblings (inc)	The selected member and its siblings
Parents	The member in the level above the selected member
Parents (inc)	The selected member and its parent
Children	All members in the level immediately below the selected member
Children (inc)	The selected member and its children
Level 0 Descendants	All descendants of the selected member that have no children

Optional: Click the Function Selector icon on the right side of the dialog box to insert functions for the members selected in the right pane.

7 If variables or attributes are defined, click the **Variables** tab to select members for user variables, substitution variables, and attributes. Members of each category are displayed as children. Only members to which the user has read access are displayed in forms.

- **User Variables**: Select members as described in "About Selecting User Variables as Members" on page 111.

- **Substitution Variables**: Select members as described in "About Selecting Substitution Variables as Members" on page 110. The selection depends on the preference setting for **Enable the Display of Substitution Variables**, described in "Specifying System Settings" on page 174.

- **Attributes**: Select attributes based on the values described in the following table. Selecting a non-level 0 attribute selects all level 0 descendants and applies the operator to each, as described in "About Selecting Attribute Values as Members" on page 106.

  Optional: To select functions that apply to custom attributes, click . To filter by dimension, select an option from the Dimension drop-down list.

Table 22 Attribute Value Selection

Operator	Attribute Values Included
Equal	Equal to the selected attribute

Operator	Attribute Values Included
NotEqual	Not equal to the selected attribute
Greater	Greater than the selected attribute
GreaterOrEqual	Greater than or equal to the selected attribute
Less	Less than the selected attribute
LessOrEqual	Less than or equal to the selected attribute

8   Be sure the member selection options are appropriate for the context from which you invoked the member selector dialog box.

9   Click **OK**.

Using Wildcards in Searches

You can use these wildcard characters to search for members.

Table 23   Wildcard Characters

Wildcard	Description
?	Match any single character
*	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].

About Selecting Attribute Values as Members

If attribute members are defined, you can select attribute values on the Member Selection page. 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 page 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

Medium

3

4

Large

5

6
```

Table 24   Example: Numeric Attribute Evaluation

Selected Operator	Selected Attribute Value	Result	Explanation
Equal	Large	5, 6	The Equal operator is applied to all level 0 descendants of Large, which includes 5 and 6.
Less	Medium	1, 2, 3	The Less operator is applied to all level 0 descendants of Medium. This includes values < 3 OR < 4, which results in 1, 2, and 3.
Greater	Medium	4, 5, 6	The Greater operator is applied to all level 0 descendants of Medium. This includes values > 3 OR > 4, which results in 4, 5, and 6.
GreaterOrEqual	Medium	3, 4, 5, 6	The GreaterOrEqual operator is applied to all level 0 descendants of Medium. This includes values >=3 OR >= 4, which results in 3, 4, 5, and 6.
LessOrEqual	Medium	1, 2, 3, 4	The LessOrEqual operator is applied to all level 0 descendants of Medium. This includes values <=3 OR <=4, which results in 1, 2, 3, and 4.
NotEqual	Medium	1, 2, 5, 6	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.

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 25   Example: Text Attribute Evaluation

Selected Operator	Selected Attribute Value	Result	Explanation
Equal	Medium	Box, Carton	The Equal operator is applied to all level 0 descendants of Medium, which includes Box and Carton.
NotEqual	Medium	Envelope, Packet, Barrel, Crate	The NotEqual operator is applied to all level 0 descendants of Medium. This includes values not equal to Box AND not equal to Carton, which results in Envelope, Packet, Barrel, and Crate.
Less	Medium	Box, Packet, Envelope	The Less operator is applied to all level 0 descendants of Medium. This includes everything at a lower position than Carton OR a lower position than Box, which results in Box, Packet, and Envelope.
LessOrEqual	Medium	Envelope, Packet, Box, Carton	The LessOrEqual operator is applied to all level 0 descendants 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.

About 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, see "Assigning Access to Members and Business Rules" on page 23.

- 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" on page 69.

- 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 is not 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 do not 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" on page 69.

- To set display, functionality, and printing options, see "Setting Form Precision and Other Options" on page 68.

About 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.

Notes:

- When you open or calculate values on forms, the Planning application replaces substitution variables with values assigned to them.

- You create and assign values to substitution variables within Planning. These substitution variables are then available in Planning when you select members for a form. For instructions on creating and assigning values to substitution variables using Planning, see "Working with Substitution Variables" on page 99.

- 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 is not valid to select a substitution variable named `CurrYr` for the Year 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; Planning 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.

- Planning checks the validity of substitution variables when they are used (for example, when the form is opened). It does not 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" on page 60).

2 In **Member Selection**, select substitution variables the same way you select members, using the right, left, and double arrows 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**.

About 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" on page 100.

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" on page 60).

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**.

8

Managing Data Validation

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 planning 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" on page 131.

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.

Watch this video to learn about building validation rules and processing cells in Planning.

▶ Video

➤ To create and update validation rules:

1   Select **Administration**, then **Manage**, and then **Forms and Ad Hoc Grids**. Create or edit a form, and then click the **Layout** tab in the Form Management page.

2   In the **Layout** tab, right-click the grid, row heading, column heading, or cell for which you want to add or update the validation rule.

> Note: When you hover the cursor over cells in the **Layout** tab, a context menu displays whether the cell contains a validation rule. To view the validation message, select **Show Data Validation Messages**. The context menu also displays when a single cell is selected.

3   Select **Add/Edit Validation Rules** to create or update rules.

4   To add a rule, click **Add Rule** 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 **Duplicate** and then update the rule. To view rules, click **View Rule**. See "Viewing Data Validation Rules" on page 117.

5   Update the rule.

a.   Under **Condition**, click ⌄ and 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" on page 118.

b.   From **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" on page 118.

c.   If applicable for the selected **Source Type**, enter a value in **Source Value** by clicking ⌄ to select an option, or 🖊 to enter a free form value.

d.   Select the appropriate operator for the evaluation: **=**, **!=**, **<**, **<=**, **>**, **>=**, **Equals**, **Not Equals**, **Contains**, **Starts With**, or **Ends With, In, or Not In**.

For examples, see "Data Validation Conditional Operators" on page 129.

e.   Select an option for the appropriate **Target Type** for the rule.

f.   Update conditions by clicking an icon in the **Actions** area on the right side 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" on page 118.

- 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.

6 Click the Process Cell icon in the rightmost column to add processing instructions.

See "Formatting Cells and Setting the Promotional Path" on page 116.

7 When you are ready to enable the rule to make it available in the form, select **Enable Validation Rule**.

Tip: While you are 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**.

8 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**.

9 **Optional:** In the Form Management page, view and update rules.

- In the **Validation Rules** pane on the right side of the Form Management page, add, edit, or delete rules by clicking ![icon], ![icon], or ![icon].

- To view rules at the current level or higher, click in the Manage Form page, and then select an option from the Validation Rules drop-down list.

- 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" on page 117.

- To prevent validations associated with the form from executing when validating the planning unit if the currently logged in user does not have access to the form, select **Validate only for users with access to this form**.

10 In the Form 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" on page 60.

When users open the form, they can view and resolve validation messages using the Data Validation Messages pane. See *Working with Planning for Oracle Planning and Budgeting Cloud Service.*

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 the Process Cell icon ![icon] in the rightmost column.

If the icon is not 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 is not displayed until all required columns are selected for a rule. See "Creating and Updating Data Validation Rules" on page 113.

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 does not 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 ![icon]. To remove the cell background color, click ![icon].

- 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 for Oracle Planning and Budgeting Cloud Service*.

- To update the planning 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 are 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 planning unit from being promoted if the condition is fulfilled). See "Modifying the Planning Unit Promotional Path" on page 156.

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.

For additional information about data validation rules, see:

- "Creating and Updating Data Validation Rules" on page 113
- "Formatting Cells and Setting the Promotional Path" on page 116
- "Order of Evaluation and Execution for Data Validation Rules" on page 117
- "Conditions Supported by the Rule Builder" on page 118
- "Data Validation Conditional Operators" on page 129
- "Data Validation Rule Scenarios" on page 131

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 26   Priority for Rules in Forms

Default Condition Priority	Validation Message	Color	Do Not Promote
1 (lowest)	X		
1		X	
1	X	X	
2			X
3	X		X
4		X	X
5 (highest)	X	X	X

For additional information about data validation rules, see:

- "Creating and Updating Data Validation Rules" on page 113
- "Formatting Cells and Setting the Promotional Path" on page 116
- "Viewing Data Validation Rules" on page 117
- "Conditions Supported by the Rule Builder" on page 118
- "Data Validation Conditional Operators" on page 129
- "Data Validation Rule Scenarios" on page 131

Conditions Supported by the Rule Builder

Subtopics

- If Condition Values
- Then Condition Values
- Range Condition Values

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" on page 119.

- **Then:** "Then Condition Values" on page 127.

- **Check Range, Range:** "Range Condition Values" on page 128.

For additional information about data validation, see:

- "Creating and Updating Data Validation Rules" on page 113

- "Order of Evaluation and Execution for Data Validation Rules" on page 117

- "Formatting Cells and Setting the Promotional Path" on page 116

- "Viewing Data Validation Rules" on page 117

- "Conditions Supported by the Rule Builder" on page 118

- "Data Validation Conditional Operators" on page 129

- "Data Validation Rule Scenarios" on page 131

If Condition Values

Subtopics

- Current Cell Value
- Cell Value
- Column Value
- Row Value
- Cross Dim Member
- Member Name
- Member
- Account Type
- Version Type
- Variance Reporting Type
- UDA
- Attribute

These values are supported by the data validation rule builder for If conditions:

- "Current Cell Value" on page 120

- "Cell Value" on page 121

- "Column Value" on page 122

- "Row Value" on page 122

- "Cross Dim Member" on page 123

- "Member Name" on page 124

- "Member" on page 124

- "Account Type" on page 125

For information on other conditions, see "Conditions Supported by the Rule Builder" on page 118.

For additional information about data validation, see:

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:
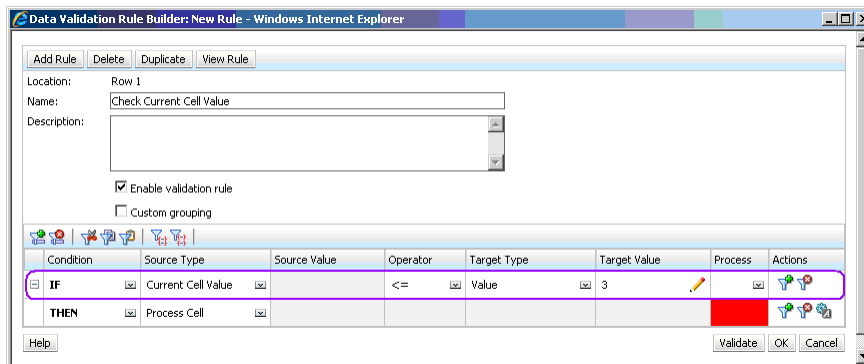
Condition Evaluation:

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.

		A	
		Column_Member 1	Column_Member 2
1	Row_Member 1	1.0	2.0
	Row_Member 2	5.0	6.0
	Row_Member 3	9.0	10.0
2	Row_Member 4	13.0	14.0
	Row_Member 5	17.0	18.0

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:

| ⊟ IF | ⌄ | Cell Value | ⌄ | A,1 | ⌄ | = | ⌄ | Value | ⌄ | 33 | ✏ | | ⌄ | 🔽🔽 |

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).

		A	
		Column_Member 1	Column_Member 2
1	Row_Member 1	1.0	2.0
	Row_Member 2	5.0	6.0
	Row_Member 3	9.0	10.0
2	Row_Member 4	13.0	14.0
	Row_Member 5	17.0	18.0

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:

| ⊟ IF | ⌄ | Column Value | ⌄ | A | ⌄ | < | ⌄ | Value | ⌄ | 3 | ✏ | | ⌄ | 🔻🔻 |

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.

		A	
		Column_Member 1	Column_Member 2
	Row_Member 1	1.0	2.0
1	Row_Member 2	5.0	6.0
	Row_Member 3	9.0	10.0
	Row_Member 4	13.0	14.0
2	Row_Member 5	17.0	18.0

Row Value

Action:

The action is performed when the value for the specified row satisfies the condition.

122

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:

| ⊟ | IF | ⌄ | Row Value | ⌄ | 1 | ⌄ | != | ⌄ | Value | ⌄ | 10 | ✏ | | ⌄ | 🔺🔻 |
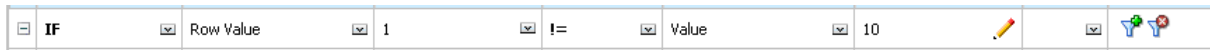
Condition Evaluation:

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.

		A	
		Column_Member 1	Column_Member 2
	Row_Member 1	1.0	2.0
1	Row_Member 2	5.0	6.0
	Row_Member 3	9.0	10.0
2	Row_Member 4	13.0	14.0
	Row_Member 5	17.0	18.0

Cross Dim Member

Action:

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.

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:

	IF		Current Cell Value			>		Cross-Dim Member		Row_Member 5	✏			🔾

Condition Evaluation:

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.

		A	
		Column_Member 1	Column_Member 2
	Row_Member 1	1.0	2.0
1	Row_Member 2	5.0	6.0
	Row_Member 3	9.0	10.0
2	Row_Member 4	13.0	14.0
	Row_Member 5	17.0	18.0

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 plan type 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:

	IF		Member Name		Account		**Equals**		Value		Total Cost	✏			🔾

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 [image], and then clicking the **Variables** tab of 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 dimensions, 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 "About Selecting Attribute Values as Members" on page 106.

Condition Definition for Member Source Type

IF	Member	Account	In	Value	IDescendants("Total Cost")		

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 Botto...		

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:

IF	Var Reporting Type		Is		Non-Expense		

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:

IF	UDA	Scenario	Is	Value	ACTUAL		

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:

| ⊟ | IF | ⌄ | Attribute | ⌄ | Entity | ⌄ | Is | ⌄ | Value | ⌄ | East | ⌄ | ⌄ | |

For additional information about data validation, see:

- "Creating and Updating Data Validation Rules" on page 113
- "Order of Evaluation and Execution for Data Validation Rules" on page 117
- "Formatting Cells and Setting the Promotional Path" on page 116
- "Viewing Data Validation Rules" on page 117
- "Conditions Supported by the Rule Builder" on page 118
- "Data Validation Conditional Operators" on page 129
- "Data Validation Rule Scenarios" on page 131

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" on page 116.

Condition Definition:

| THEN | ⌄ | Process Cell | ⌄ | | | | | |

For information on other conditions, see "Conditions Supported by the Rule Builder" on page 118.

For additional information about data validation, see:

- "Creating and Updating Data Validation Rules" on page 113
- "Order of Evaluation and Execution for Data Validation Rules" on page 117
- "Formatting Cells and Setting the Promotional Path" on page 116

Range Condition Values

Subtopics

- Check Range
- Range

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.

See:

For information on other conditions, see "Conditions Supported by the Rule Builder" on page 118.

For additional information about data validation, see:

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 >= 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 >= 5 and < 10. If this condition is met, the cell is turned red.

Condition Definition:

Condition	Source Type	Source Value	Operator	Target Type	Target Value	Process	Actions
CHECK RANGE	Current Cell Value						
RANGE	Value	5		Value	10		

For information on other conditions, see "Conditions Supported by the Rule Builder" on page 118.

Data Validation Conditional Operators

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 27   Examples of Results for Starts With, Ends With, and Contains

Operator	Compare Value	Compare To Value
Starts With	2.0	2
	1234.0	12.0
	101.0	10
	2.0	2.0
	2.5	"2."
	"YearTotal"	"Year"
Ends With	2.0	2.0
	2.0	2
	2.5	5
	2.5	".5"
	"YearTotal"	"al"
	"YearTotal"	"Total"
Contains	2.0	2.0
	2.0	2
	2.5	5
	2.5	".5"
	2.5	2.5
	23.567	3.5
	23.567	67
	23.567	"23."
	23.567	".56"
	"YearTotal"	"al"

For additional information about data validation, see:

- "Creating and Updating Data Validation Rules" on page 113
- "Order of Evaluation and Execution for Data Validation Rules" on page 117
- "Formatting Cells and Setting the Promotional Path" on page 116
- "Viewing Data Validation Rules" on page 117
- "Conditions Supported by the Rule Builder" on page 118

Data Validation Rule Scenarios

Subtopics

These scenarios provide examples of how data validation can help implement business policies.

For additional information about data validation, see:

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:

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.

For additional scenarios, see "Data Validation Rule Scenarios" on page 131.

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 does not need to update the data validation rules. John updates the form as shown in the following figure.

Form Layout at Design Time:

Properties	Layout	Other Options	Business Rules

Point of View

BU Version_1   entity1

Page

	Columns
	A
	FY09,FY10
	Actual,Budget
Rows	IDescendants(YearTotal)
1 Units,Rate,Total Cost	
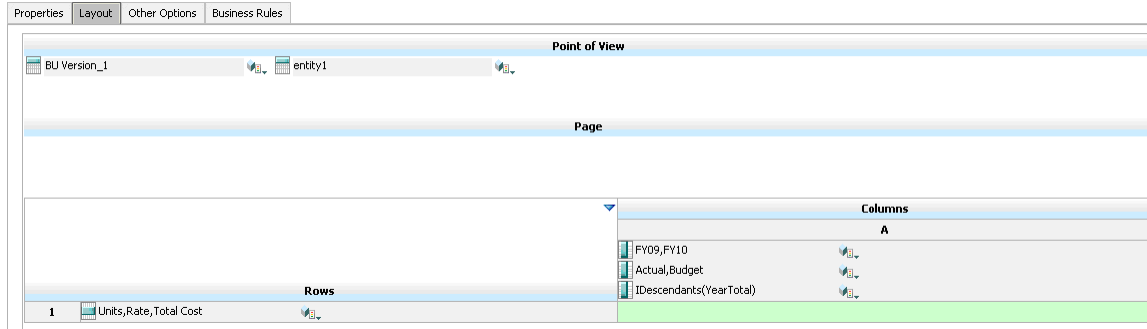
Form at Data Entry Time with Data Validations Applied:

	Version: BU Version_1									Entity: entity1			
	FY09									**FY10**			
	Actual								Budget	Actual			
	Jan	Feb	Mar	⊟Q1	⊞Q2	⊞Q3	⊞Q4	⊟YearTotal	⊞YearTotal	Jan	Feb	Mar	⊟Q1
Units	3	4	6	13	12	24	21	70	70	5	14	7	26
Rate	5	5	5	15	15	15	9	54	52	4	4	4	12
⊟Total Cost	15	20	30	195	180	360	189	3780	3640	20	56	28	312

For additional scenarios, see "Data Validation Rule Scenarios" on page 131.

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:

Properties	Layout	Other Options	Business Rules

Point of View

BU Version_1   entity1

Page

	Columns		
	A	**B**	**C**
	FY09	FY10	FY10
	Actual	Budget	"Variance Percent (Actual, Budget)"
Rows	YearTotal	IDescendants(YearTotal)	YearTotal
1 Units,Rate,Total Cost			

Data Validation Rule at Design Time:

Form at Data Entry Time with Data Validations Applied:

	FY09	FY10					FY10
	Actual	Budget					Variance Perce
	YearTotal	⊞ Q1	⊞ Q2	⊞ Q3	⊞ Q4	⊟ YearTotal	YearTotal
Units	70.0	60.0	20.0	20.0	15.0	115.0	39.13%
Rate	54.0	24.0	4.0	4.0	5.0	37.0	-45.95%
⊟ Total Cost	3780.0	1440.0	80.0	80.0	75.0	4255.0	11.16%

Tips:

- If John is not 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" on page 82.

- 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 do not 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.

For additional scenarios, see "Data Validation Rule Scenarios" on page 131.

Scenario 4

In addition to flagging the cell in red, the rule is also required to prevent anyone from promoting the planning 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:
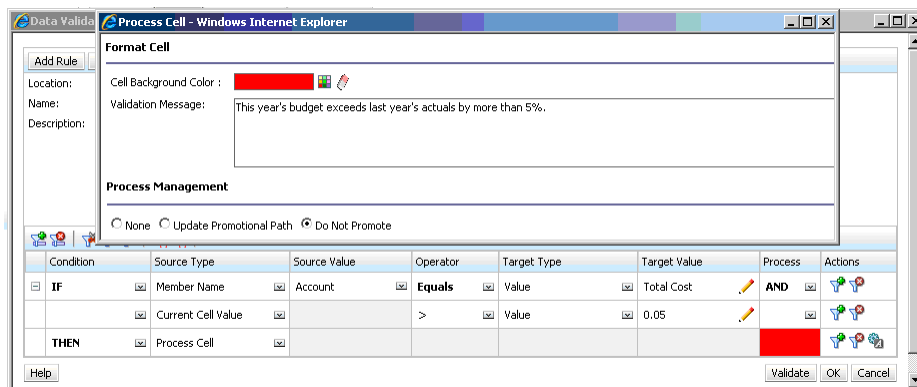
For additional scenarios, see .

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 <= ¾ of the compensation range for the employee's grade, no action is needed.

If Total Compensation is greater than ¾ of the compensation range, a validation message is provided, and the planning 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 cannot promote their planning 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 ¾ 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:

Data Validation Rule to Stop Promotion of Planning Units:

Data Validation Rule to Add the Human Resources Manager as Reviewer:

Form at Data Entry Time with Data Validations Applied and Validation Messages Shown:

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Salary	3000	3000	3000	3000	3000	3000	3000	3010	3100	3300	3300	3300
Merit												
Overtime					500							
⊟ Adjusted Salary	3000	3000	3000	3000	3500	3000	3000	3010	3100	3300	3300	3300
Bonus												10000
Sign On Bonus												
Commissions												
⊟ Total Salary	3000	3000	3000	3000	3500	3000	3000	3010	3100	3300	3300	13300
Health Care Costs	55	55	55	55	55	55	55	55	55	55	55	55
Severance												
Other Compensation												
Turnover Adjustment												
Min Compensation	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
Max Compensation	3500	3500	3500	3500	3500	3500	3500	3500	3500	3500	3500	9500
3/4 of allowed compensation	3200	3200	3200	3200	3200	3200	3200	3200	3200	3200	3200	8000
Total Compensation	3055	3055	3055	3055	3555	3055	3055	3065	3155	3355	3355	13355

Data Validation Messages

Salary not in the allowed range. [3]: 1, 2, 3

Salary exceeds 3/4 of maximum allowed. [2]: 1, 2

Salary exceeds 3/4 of maximum allowed.

For additional scenarios, see "Data Validation Rule Scenarios" on page 131.

9 Managing the Budgeting Process

About the Budgeting Process

You can track budgets and review status, process issues, and planning unit ownership using planning 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

Planning Units

Planning units are combinations of scenario, version, and entity or part of an entity. Scenarios and versions are the basis of the review cycle. Planning units submit planning data for a scenario and version. For example, a planning unit might consist of a version (Best Case), an entity (New York), and a scenario (Actual). Planning units can also include secondary dimensions within any entity, refining the granularity of a planning unit.

Review Process

The planning 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 a planning 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 is not necessarily on the promotional path

The selected approvals template determines the first user to review the budget (see "Setting Planning Unit Hierarchy Name, Scope, and Template" on page 145). 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 planning unit before sending the budget to the next reviewer. The validation runs all data validation rules defined for the planning unit with which the reviewer is working, and reports any data errors or changes in the planning unit promotional path. See "Modifying the Planning Unit Promotional Path" on page 156.

Tip: To display users' full names (instead of their user IDs) in Approvals, select the option **Display Users' Full Names**, as described in "Specifying System Settings" on page 174.

Planning Unit Hierarchy

A planning unit hierarchy contains planning units and entities that are part of the review process.

Parent/child relationships between planning 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 cannot 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 budget administrators can change its status.

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 planning unit data, and designate a specific reviewer or owner to review data that meets some condition.

For example:

- Conditionalizing the planning unit promotional path
- Preventing the promotion of planning units that include invalid data

When you design rules that affect the planning 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 Chapter 8, "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.

Setting Up Email for Approvals Notification

Administrators must specify the email server details before others can enable email notification. The email server details for approvals notifications come from the EPM Registry, which are set using the Oracle Hyperion Enterprise Performance Management System Configurator under Common Settings.

Approvals notification are enabled on the **Application Settings** page. After email notification is enabled, users receive email when they become owners of planning units, or are specified as a user to notify. This feature is available for SMTP email systems only.

To enable approvals notification, see "Setting Application Defaults" on page 173.

Defining the Budgeting Process

Subtopics

- Approvals Process
- Approval Operations and Data Validations

Approvals Process

Planning supports bottom-up, distributed, or free-form budgeting. Typically, high-level users start the planning 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 cannot access it.

To define the budget process, administrators define:

- Planning unit hierarchies
- Owners and reviewers of the planning 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 cannot 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 planning unit for it to display on the report.

Starting and Supporting the Review Process

After administrators start the review process, the planning 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.

➤ To start the review process:

1 Select **Tools**, and then **Manage Approvals**.

2 In **Scenario**, select a scenario.

3 In **Version**, select a version.

4 Click **Go** to display the planning units associated with the selected scenario and version combination.

The planning units listed are enabled for approvals.

If no planning units are started with the selected scenario and version, this message is displayed: **You have not assigned Planning Unit Hierarchy to the selected Scenario and Version combination.**

5 Select **Tree View** and then in **Plan Cycle**, click **Start** to begin the review process.

Note: If the planning unit hierarchy uses the Bottom Up template, selecting **Start** starts the planning unit and runs the Originate action. These actions set the user defined as the planning unit owner in the planning unit hierarchy as the current owner, and the planning unit status changes to **Under Review**.

6 **Optional**: Select **Exclude** to remove a planning unit from the planning process or from being tracked in the system.

Caution!	After you exclude a planning unit, all associated annotations and history are discarded. Planning unit status is returned to **Not Started** and the owner is set to **No Owner**. Data values are retained.

Printing Planning Unit Annotations

Administrators can check planning unit status by reporting on annotations for a set of scenarios, versions, and planning unit members. Reports can be based on process status. The application name, selected scenarios, versions, and entities, planning 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 planning unit annotations:

1 Select **Tools**, and then **Reports**.

2 Select **Planning Unit Annotations**.

3 Under **Planning Units**, 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.

Managing Planning Unit Hierarchies

Subtopics

- Creating Planning Unit Hierarchies
- Assigning Planning Unit Hierarchy Scenario and Version Combinations
- Selecting Planning Unit Hierarchy Scenario and Version
- Editing Planning Unit Hierarchies
- Deleting and Renaming Planning Unit Hierarchies
- Viewing Planning Unit Hierarchy Usage
- Synchronizing Planning Unit Hierarchies
- Exporting Planning Unit Hierarchies
- Importing Planning Unit Hierarchies

Use planning unit hierarchies to adapt the budgeting process to all types of organizational requirements.

- Planning units are a combination of the entity and other dimensions. For example, if an application includes all of a company's products, the planning unit hierarchy for North America can include dimensions and members appropriate to products sold in North America. Similarly, the planning 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 planning 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 planning unit hierarchy. If a change occurs in the dimension, the planning unit hierarchy can be easily updated.

- Import and export planning unit hierarchies.

- Create planning unit hierarchies that differ by scenario and version. For example, the Budget scenario can have a large planning unit hierarchy consisting of departments, accounts, and products, while the Forecast has a simpler process organization with fewer levels of approval.

Creating Planning Unit Hierarchies

Subtopics

- Setting Planning Unit Hierarchy Name, Scope, and Template
- Selecting Planning Unit Hierarchy Members
- Assigning Planning Unit Owners and Reviewers
- About Group-based Approvals

Entity is the primary dimension for each planning 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 planning unit hierarchy structure.

➤ To create a planning unit hierarchy:

1 Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**.

2 Click **Create**.

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 Planning Unit Hierarchy Name, Scope, and Template" on page 145).

4 Select the primary and subhierarchy members to include in the approval process (see "Selecting Planning Unit Hierarchy Members" on page 146).

5 Assign owners and reviewers for each stage of the approval process and create the planning unit promotional path (see "Assigning Planning Unit Owners and Reviewers" on page 147).

6 Click **Save**.

Setting Planning Unit Hierarchy Name, Scope, and Template

➤ To set up a planning unit hierarchy:

1 Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**.

2 Take an action:

- To create a new hierarchy, click **Create**.

- To edit an existing hierarchy, select a planning unit hierarchy, and then click **Edit**.

3 Select **Approvals Dimension**.

4 In **Hierarchy Name**, provide the planning unit hierarchy name.

5 **Optional:** Provide a description.

6 In **Enable Approvals**, select:

- **All** to add all planning units to the approval process.

- **None** to include no planning units in the approval process by default.

  To add planning unit groups or individual planning units to the approval process, see "Selecting Planning Unit Hierarchy Members" on page 146.

- **Custom** to define which planning units to include in the approval process.

  You can add individual planning units and planning units based on Parent member and generation criteria. See "Selecting Planning Unit Hierarchy Members" on page 146.

7 In **Approvals Template**, select:

- **Bottom Up** (see "Bottom-Up Budgeting" on page 145).

- **Distribute** (see "Distributed Budgeting" on page 146).

- **Free Form** (see "Free-Form Budgeting" on page 146).

8 In **Plan Type**, select the plan type from which aggregated planning unit values are derived.

  See "Assigning Planning Unit Owners and Reviewers" on page 147.

9 Take an action:

- Click **Next** or select **Primary and Subhierarchy Selection** to select the planning unit hierarchy members (see "Selecting Planning Unit Hierarchy Members" on page 146).

- Click **Save** and then **Finish** to save changes and close the planning 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 access permissions defined for the planning 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 are not using the approvals features described in "Creating Planning Unit Hierarchies" on page 144.

Selecting Planning Unit Hierarchy Members

➤ To select planning unit hierarchy members:

1 Take an action:

   a. From **Approvals Dimension**, click **Next** or select **Primary and Subhierarchy Selection** to continue defining a planning unit hierarchy.

   b. Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**. Select a planning unit hierarchy, then click **Edit**, and then select **Primary and Subhierarchy Selection** to edit members.

   Notes:

   ● At any time during planning unit hierarchy member selection, you can click **Reset to Default Hierarchy** to reset the planning unit hierarchy to its default membership defined in the Approvals Dimension page.

   ● Shared members are not displayed in the planning unit hierarchy.

2 Define how to display the planning unit hierarchy:

   ● Right-click and select **Expand** or **Expand All Below** to expand the display.

   ● Right-click and select **Collapse** or **Collapse All Below** to collapse the display.

   ● Select **All Entities** to display all potential planning units.

   ● Select **Planning Units** to display only enabled planning units.

   ● For **Search**, select **Name**, **Alias**, or **Both**. Enter any part or all of a name in **Search** to locate an entity, then click to search forward (down) or to search backward (up) in the planning unit hierarchy.

   ● To move from page to page in a multipage planning 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 planning units not included in the default settings for the approval process, check the box to the left of the planning unit name to include the planning unit in the approval process.

4   **Optional:** Right-click a planning unit name to define subhierarchy members for the approval process, and then select one option:

   - **Include Children** to include the children of the planning unit.

   - **Include Member** to include just the planning unit, but none of its descendants.

   - **Include All Descendants** to include all descendants of the planning unit.

   - **Include Generation** to include one or more planning unit generations. Specify the generations to include when prompted.

   - **Exclude Children** to exclude the children of the planning unit.

   - **Exclude Member** to exclude just the planning unit, but none of its descendants.

   - **Exclude All Descendants** to exclude all descendants of the planning unit.

   - **Exclude Generation** to exclude planning unit generations. Specify the generations to exclude when prompted.

5   **Optional:** Add a secondary dimension to a planning unit included in the approval process to provide finer granularity:

   a.   Select a dimension from **Dimension**.

   b.   In **Parent Member**, click 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 planning unit hierarchy that meet the selected criteria.

   e.   **Optional:** Click in **Selected Members** for the planning 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 planning unit ownership (see "Assigning Planning Unit Owners and Reviewers" on page 147).

   - Click **Save** and then **Finish** to save changes and close the planning unit hierarchy.

Assigning Planning Unit Owners and Reviewers

Planning unit ownership is inherited from the planning unit parents. Planning unit reviewers are also inherited. You can also explicitly specify planning unit owners and reviewers to assign owners and reviewers other than those planning units inherit.

➤ To assign planning unit reviewers and owners:

1   Take an action:

a. From **Primary and Subhierarchy Selection**, click **Next** or select **Assign Owners** to continue defining a planning unit hierarchy.

b. Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**. Select a planning unit hierarchy, then click **Edit**, and then select **Assign Owners** to specify planning unit ownership.

2   Select a planning unit, then under **Owner**, click [icon] to select an owner.

A planning 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" on page 149.

3   Under **Reviewer**, click [icon] and select planning 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" on page 149.

> **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 planning unit. The first reviewer in the list is the first user to work on the planning unit. When the first reviewer promotes or submits the planning unit, the second reviewer selected becomes the planning unit owner, and so on through the list of reviewers you create.

4   To enable viewing the aggregated value of planning units during approvals, under **Value Definition Members**, click [icon] to define the intersection of non-planning unit dimensions.

Notes:

- **For non-planning unit dimensions:** To view aggregated planning unit values, on **Member Selection**, you must select one member for each non-planning unit dimension in the plan type selected on the **Approvals Dimension** tab. If you do not select members for the non-planning unit dimensions, the aggregated planning unit value is blank.

- **For planning unit dimensions:** Value definition members specified for a parent member in the planning unit hierarchy are inherited by the descendants of that member. If value definition members are selected for both a parent and child member of a planning 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 planning unit hierarchy are selected.

- **In multicurrency applications:** In **Preferences**, users set the **Reporting Currency** in which to display the aggregated planning unit. See the *Working with Planning for Oracle Planning and Budgeting Cloud Service*. To ensure that totals are correct, remind users to launch the Currency Conversion business rule.

> **Note:**   In multicurrency applications, the reporting currency is assumed, so you do not select a value definition member for currency.

5   Under **Promotional Path**, click to display the planning unit promotional path, verify that it is correct, and then correct any errors.

6   Under **Notify These Users**, click to select the users to notify for each move of the planning unit from one user to another.

7   **Optional**: Repeat Steps 2 through 5 for other planning units to change their inherited owners or reviewers.

8   Click **Save** to save your work and continue, or click **Finish** to save your work and close the planning unit hierarchy.

About Group-based Approvals

When assigning Planning unit owners, you can assign individual users, or you can assign a group. When assigning Planning unit reviewers, you can assign individual users, a group, or multiple groups.

Assigning a Group as the Owner

Only one user or one group can be assigned as an owner of a planning 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 1

```
North America - Bill
    USA - Nick, Sandy, Kim
        CA - John
```

The approval path is John, Nick, Sandy, Kim, Bill.

Example 2

```
North America - Bill
    USA - {Group A}
        CA - John
```

The approval path is John, any user in Group A, Bill.

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 4

```
North America - {Group B}, Bill
     USA - Susan, {Group A}
     CA - John
**A validation rules indicates that if New Hires > 4, then before USA, set {Group HR} as
reviewer.
```

If the condition is true for the CA Planning unit, then the path is: John, any user in Group HR, Susan, any user in Group A, any user in Group B, Bill.

Assigning Planning 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 planning unit hierarchies to the scenario and version combinations included in the budget process.

➤ To assign planning unit hierarchy scenario and version combinations:

1   Select **Administration**, then **Approvals**, and then **Scenario and Version Assignment**.

2   Optional: Click ⊞ next to the planning unit to view what scenarios and versions are assigned to it. Click ⊟ to close the list.

3   Add a scenario and version assignment.

    a. Click 🗐, in the **Actions** column, for the planning unit.

    b. Click **Select** in the **Scenario** column, and then select the scenario to associate with the planning 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.

4   Optional: Click ✖ to remove a scenario and version assignment.

5   Click **Save** to save the scenario and version assignments and continue or click **Cancel Changes** to undo any changes made since the last save.

Selecting Planning Unit Hierarchy Scenario and Version

➤ To select the planning unit hierarchy scenario and version:

1 Select **Tools**, and then **Manage Approvals**.

2 In **Scenario**, select a scenario.

3 In **Version**, select a version.

4 Click **Go** to display the planning unit hierarchy defined for the selected scenario and version.

5 In **Display**, click **Tree View** to display the planning units as a hierarchy, or click **Flat View** to display the planning units as a list.

6 **Optional**: If you own planning units for this hierarchy, check **Select Mine** to select them.

7 For each planning unit, the following is displayed:

- **Plan Cycle** (Tree View only), which displays whether the planning unit is started and if it is 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

- In **Actions**, click **Details**, to display planning unit details and add or edit planning unit annotations

Editing Planning Unit Hierarchies

➤ To edit a planning unit hierarchy:

1 Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**.

2 Check the planning unit hierarchy with which you want to work.

3 Click **Edit**.

4 Select the appropriate tab, depending on the changes you want to make (for example, select **Assign Owners** to edit ownership of the planning units in the hierarchy).

- **Approvals Dimension**

  See "Setting Planning Unit Hierarchy Name, Scope, and Template" on page 145 to modify the planning unit hierarchy description or approvals default scope.

- **Primary and Subhierarchy Selection**

  See "Selecting Planning Unit Hierarchy Members" on page 146 to select or remove members from approvals.

- **Assign Owners**

See "Assigning Planning Unit Owners and Reviewers" on page 147 to modify planning unit owners or reviewers

- **Usage**

  See "Viewing Planning Unit Hierarchy Usage" on page 152 to view the Planning objects (data validation rules or scenario and version assignments) that reference the planning unit hierarchy

5 Click **Save** when done.

To select a planning unit hierarchy scenario and version with which to work, see "Selecting Planning Unit Hierarchy Scenario and Version" on page 151.

Deleting and Renaming Planning Unit Hierarchies

You can delete a planning unit hierarchy if it is not referenced by other Planning objects such as data validation rules or scenario and version assignments. The **Usage** tab displays objects that reference the selected planning unit hierarchy and links you to the objects so you can disassociate them from the hierarchy.

You can also change the name of a planning unit hierarchy. Renaming a planning unit hierarchy does not affect Planning objects that reference the hierarchy.

➤ To delete or rename planning unit hierarchies:

1 Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**.

2 Take an action:

- To delete, select the planning unit hierarchy to delete, and then click **Delete**.
- To rename, select the planning unit hierarchy to rename, and then click **Rename**. Enter the new name for the planning unit hierarchy in the dialog box.

3 Click **OK**.

If you select a planning unit hierarchy that is referenced by other Planning objects, an error message is displayed. See "Viewing Planning Unit Hierarchy Usage" on page 152 to view and edit the Planning objects that reference the planning unit hierarchy.

Viewing Planning Unit Hierarchy Usage

Planning unit hierarchies might have dependencies such as scenario and version assignments or data validation rules which are defined in forms. If dependencies exist for a planning unit hierarchy, the hierarchy cannot be deleted until the dependencies are removed. The Usage tab enables you to view planning unit hierarchy dependencies and links you to the forms or scenario and version assignments so you can remove the dependencies, if needed.

➤ To view planning unit hierarchy usage:

1 Select **Administration**, then **Approvals**, and then **Planning Unit Hierarchy**.

2   Select a planning unit hierarchy, then click **Edit**, and then select **Usage** to view planning 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 planning unit hierarchy. See "Creating and Updating Data Validation Rules" on page 113.

4   Select **Scenario and Version Assignment** to view associated scenario and version assignments:

- If no scenario and version assignments are associated, a message is displayed.

- If scenario and version assignments are associated, they are listed by scenario. Click the version hyperlink to display the assignment in a new tab. You can then remove the scenario and version assignment to disassociate it from the planning unit hierarchy. See "Assigning Planning Unit Hierarchy Scenario and Version Combinations" on page 150.

5   If dependencies are removed during steps 3 or 4, click **Refresh** on the **Usage** tab to view the remaining dependencies.

6   If you are deleting a planning unit hierarchy, repeat steps 3, 4, and 5 until all dependencies are removed.

Once all dependencies are removed from a planning unit hierarchy, you can delete the hierarchy. See "Deleting and Renaming Planning Unit Hierarchies" on page 152.

Synchronizing Planning Unit Hierarchies

When you add, delete, or modify dimension members that are used in planning unit hierarchies, the affected planning unit hierarchy must be synchronized with the changes. When you display the list of planning unit hierarchies, the entry for each planning unit hierarchy specifies whether recent changes are reflected in the planning unit hierarchy. Use this procedure to synchronize dimension member changes with the planning unit hierarchy.

Note: When you add dimension members, they are added as planning units only if they meet the criteria in the inclusion rules for the planning unit hierarchy. For example, if the added entity is a fourth-generation entity, and the inclusion rules specify generations one through three as planning units, the entity is not added as a planning unit. If the entity is a third-generation member, however, it is added as a planning unit the next time the planning unit hierarchy is edited and saved, or synchronized.

➤   To synchronize changes to planning unit hierarchies:

1   Select **Administration**, then select **Approvals**.

2   Select **Planning Unit Hierarchy.** Under the **Synchronized** column, planning unit hierarchies are labeled as follows:

- **Synchronized**—Changes are synchronized with the planning unit hierarchy

- **Not Synchronized**—Changes are not synchronized with the planning unit hierarchy

- **Locked By** *user*—A user is editing or synchronizing the planning unit hierarchy

  Note: If a user begins editing or synchronizing a planning unit hierarchy after you display the planning unit hierarchy list, the planning unit hierarchy list does not display "Locked" for the planning unit hierarchy. If you try to synchronize this planning unit hierarchy, the synchronization does not occur, and an error message states that it is being edited.

3 Select a planning unit hierarchy listed as **Not Synchronized** in **Synchronized**, and then click **Synchronize**.

   Changes are applied to the planning unit hierarchy, and the list of planning units is updated according to the inclusion rules defined for the planning unit hierarchy.

   Note: You cannot synchronize changes to a planning unit hierarchy that another user is editing or synchronizing.

Exporting Planning Unit Hierarchies

When you export a planning unit hierarchy, you create a file that contains the planning unit hierarchy information. After you create this file, you can copy its contents to an existing planning unit hierarchy (see "Importing Planning Unit Hierarchies" on page 154).

➤ To export planning unit hierarchies:

1 Select **Administration**, then **Approvals**, and then **File Based Import/Export**.

2 Select **Export Planning Unit Hierarchy**.

3 In **Existing Planning Unit Hierarchy Name**, select a planning 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 Planning Unit Hierarchies

A planning unit hierarchy can be populated with the contents of a planning unit hierarchy import file. The import file is the result of exporting an existing planning unit hierarchy. See "Exporting Planning Unit Hierarchies" on page 154.

Importing the planning unit hierarchy information does not create a planning unit hierarchy. The planning unit hierarchy populated from the export file must exist and have at least a name before the import. Loading planning unit hierarchies first deletes all members of the planning unit hierarchy, and then adds each member specified in the input file as a new member.

Note: A planning unit hierarchy load deletes an existing member and its children from the hierarchy if the member is not specified in the input file.

➤ To import a planning unit hierarchy:

1 Select **Administration**, then **Approvals**, and then **File Based Import/Export**.

2 Select **Import Planning Unit Hierarchy**.

3 In **Existing Planning Unit Hierarchy Name**, select the planning unit hierarchy receiving the exported information.

> **Note:** The imported planning unit hierarchy includes the defined owner, reviewers, and rules for determining the promotional path.

4 For **Planning Unit Hierarchy with Ownership**, click **Browse** to select the exported planning 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 planning unit hierarchy information in the exported file was successfully copied to the planning unit hierarchy that you selected in **Existing Planning 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 planning unit hierarchy.

Planning Unit Promotional Path

Subtopics

● Modifying the Planning Unit Promotional Path
● Design Considerations for Planning Unit Promotional Path Data Validation Rules

A planning unit moves from person to person and department to department based on the owners and reviewers you assign to each planning unit and its parents in the planning unit hierarchy.

There are two ways to affect the promotional path of a planning unit hierarchy:

● Modify owners and reviewers for planning units and their descendants using approvals actions (for example, Promote or Reject)

● Use data validation rules to determine whether a change in the planning unit promotional path is necessary

Modifying the Planning Unit Promotional Path

When you select an owner and reviewers for a planning unit and its parents, you set the planning unit promotional path (see "Planning Unit Promotional Path" on page 155). 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 validation rules that test for these exceptions, and then modify the promotional path when necessary. You can also use data validation rules to stop a planning unit from passing to the next reviewer when data errors exist in the planning unit.

➤ To modify the planning unit promotional path:

1 Create or select a data validation rule (see "Creating and Updating Data Validation Rules" on page 113).

2 In the **Data Validation Rule Builder**, click [icon] to create promotional path conditions and specify the action taken when a budget calculation does not 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 a planning 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 [icon] and [icon] 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 **Planning Unit Hierarchy** and **Planning Units**, click [icon] to select the planning unit hierarchy and planning units the promotional path condition affects.

8 In **Promotional Path Condition**, specify the promotional path changes for the planning units selected in Promotional Path Condition. Select:

a. In **Position**, specify where the promotional path changes for the planning units selected in **Planning Units**. Select:

- **Before** to insert the alternate owner or reviewer before ownership reaches the planning units specified in **Promotional Path Condition**.

- **After** to insert the alternate owner or reviewer after ownership reaches the planning units specified in **Promotional Path Condition**.

- **Before and After** to insert the alternate owner or reviewer before and after ownership reaches the planning units specified in **Promotional Path Condition**.

> **Note:** After you add promotional path conditions to a data validation rule, the planning unit promotional path displays the potential changes in planning unit ownership as an optional path above the defined promotional path. The display indicates where the path can change and how the alternate path rejoins the defined promotional path.

b. In the **Planning Units** column of **Promotional Path Condition**, click to select the planning units the action selected in **Position** affects.

> **Note:** The selected planning units must be ancestors of the planning units selected in the previous step. If you select planning units that are not ancestors, the promotional path condition is not evaluated when the planning unit hierarchy is validated.

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:** Ensure the prefix specified is the same as the prefix that was assigned while creating the UDA, including any spaces. For instance, if there is a space between the prefix and the username (ProdMgr Kim), ensure there is a space after ProdMgr.

10 **Optional:** Provide messages for the users involved in the planning unit promotional path changes:

a. Click in:

- **Sender Message** to enter the email message text sent to the user promoting the planning unit when the promotional path changes.

- In **Reviewer Message** to enter the email message text sent to the user receiving the planning 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 planning unit changes status.

11 Click **OK** to save the promotional path conditions, and return to data validation rule creation.

Design Considerations for Planning Unit Promotional Path Data Validation Rules

When you design rules that affect the planning 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 Chapter 8, "Managing Data Validation."

Managing Task Lists

Subtopics

- Working with Task List Folders
- Working with Task Lists
- Adding Instructions to Task Lists
- Adding and Defining Tasks
- Adding Tasks to Task Lists
- Editing Task Lists
- Linking Tasks to Planning Pages
- Assigning Access to 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.

Watch this video to learn about using task lists in Planning.

▶ Video

➤ To manage task lists:

1 Select **Administration**, then **Manage**, then **Task Lists**.

2 Use the **Manage Task Lists** page to configure task list folders and create and manage task lists.

If a form contains promotional path rules, you can view validation reports in a Validation Reports folder.

Working with Task List Folders

➤ To create, rename, move, or delete task list folders:

1 Open the **Manage Task Lists** page.

See "Managing Task Lists" on page 158.

2 In the **Task List Folders** area, perform a task:

- To create a task list folder, select the folder in which to create the task list folder, and then click **Create**. Enter the name of the task list.

- To rename a task list folder, select the folder to rename, then select **Actions**, and then select **Rename**. Enter the new name.

- To move a task list folder, select the folder to move, then above the **Task List Folders** area, click **Move**. Select the destination folder.

- To delete a task list folder, select an empty folder to delete, and then click **Delete**.

Note: You cannot move or delete the Task Lists folder. If you try to delete a folder containing additional folders, an error message is displayed.

3 Click **OK**.

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 Open the **Manage Task Lists** page.

See "Managing Task Lists" on page 158.

2 In the **Task List Folders** area, perform a task:

To create a task list:

a. Select the folder in which to create the task list.

b. Above the **Task List** area, select **Actions**, and then **Create**.

c. Enter the task list name.

To rename a task list:

a. Select the folder with the task list to rename.

b. Select the task list, then select **Actions**, and then select **Rename**.

c. Enter the new task list name.

3 Click **OK**.

4 To define the task list, see:

- "Adding Instructions to Task Lists" on page 160.

- "Adding and Defining Tasks" on page 160.

Adding Instructions to Task Lists

➤ To add instructions to task lists:

1 Open the **Manage Task Lists** page.

   See "Managing Task Lists" on page 158.

2 In the **Task List Folders** area, select the folder containing the task list to modify, and select the task list.

3 For **Task List**, select the task list to modify, select **Actions**, and then **Edit**.

4 For **Edit Task List**, click **Instructions**.

5 Enter instructions for the task list.

6 Click **Save** and **Close**.

Adding and Defining Tasks

After creating task lists, you can add and define tasks such as entering data in forms and running required business rules. See "Adding Tasks to Task Lists " on page 160

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. You must configure an email server.

➤ To add tasks to task lists:

1 To the left, select **Manage Task Lists** .

2 Perform a task:

   - Define a task list by selecting **Actions**, then **Create Task List**, and then entering a name.

   - Modify a task list to add a new task by selecting it, selecting **Actions**, and then **Edit Task List**.

3 Select the task list.

4 Select **Actions**, and then select **Add Child**.

5 Under **Task Details**, for **Task**, enter a task name.

6 From **Type**, select the kind of task you are creating and what it should enable users to do. For example, if you are creating a task that enables Service Administrators to modify all dimensions at runtime, select Dimension Editor:

- **URL**—Opens a specified URL

- **Form**—Opens a form

- **Business Rule**—Launches a business rule

- **Manage Approvals**—Starts the review process with a specified scenario and version

- **Job Console**—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 (processing, or error, for example)

- **Copy Version**—Copies the current form's data, including supporting detail, annotation, cell text, and cell documents, for the specified source and destination versions

- **Dimension Editor**—Enables Service Administrators to view and modify a dimension at runtime

- **Form Management**—Enables Service Administrators to view a summary of specific forms

- **Process Management**—Enables Service Administrators to view and modify a specific planning unit

- **Security Management**—Enables Service Administrators to assign access to a dimension or form

- **Financial Reporting**—Enables Service Administrators to access your Financial Reporting home page

- **Refresh Application**—Enables Service Administrators to refresh your application to capture recent changes

- **Export and Import**—Enables Service Administrators to import and export data and metadata via flat files.

7 Specify the information required for the task type using this table:

Table 28   Task Information

Task Type	Action
URL	Enter a fully qualified URL to associate with this task, such as `http://www.company_name.com`, and then go to step 8.
Form	Select the form for users to complete, and then go to step 8.
	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.
Business Rule	Perform these tasks:
	● From **Plan Type**, select the plan type associated with the business rule to execute.
	● From **Business Rules**, select the business rule to execute.
	● Go to step 8.
Manage Approvals	Specify the scenario and version in which the user can start the approvals process, and then go to step 8.

Task Type	Action
Job Console	Perform these tasks: ● In **Job Type** select the kind of job, such as to Copy Decision Package, to be displayed in the console. ● In **Status**, select the situation of the task, such as processing or in error, to be displayed in the console. ● Go to step 8.
Copy Version	Enables users to copy a version of the current form's data. Select values for the following options, and then go to step 8: ● **Scenario** ● **Copy From** ● **Copy To**
Dimension Editor	Enables Service Administrators to view and modify a dimension at runtime. Perform these tasks, and then go to step 8: ● To enable Service Administrators to view and modify all dimensions, select **All** from **Dimension**. ● To enable Service Administrators to view and modify a particular dimension, select it.
Form Management	Enables Service Administrators to view a summary of specific forms. Perform these tasks, and then go to step 8: ● In **Form Folder**, click ➕ to select the folder that contains the forms that Service Administrators can display and review at runtime. ● In **Forms**, click ➕ to select the individual form that Service Administrators can review.
Process Management	Enables Service Administrators to view and modify a specific planning unit at runtime. From **Planning Unit Hierarchy**, select the planning unit that the user can view and modify, and then go to step 8.
Security Management	Enables Service Administrators to assign access to dimensions or forms. From **Security Artifact**, indicate if you want Service Administrators to assign access rights to forms or dimensions, and then go to step 8.
Financial Reporting	Enables Service Administrators to access your Financial Reporting home page. After enabling, go to step 8.
Refresh Application	Enables Service Administrators to refresh the application at runtime to capture recent changes. After enabling, go to step 8.
Export and Import	Enables Service Administrators to import and export data and metadata. From **Load Method,** 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 8.

8 In **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.

9 To ensure that the task is started on a certain date and then completed by a certain date, perform these tasks:

a. Select **Start Date**, then click 📅, and then use the calendar to specify the year, month, day, and time by which the task must begin.

b. Select **End Date**, then 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.

10 **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 [icon] 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.

11 **To make task completion dependent on completing another primary task, click [icon] in Dependency and then select the primary task.**

12 **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).**

13 **Click Save and then click OK.**

Editing Task Lists

Subtopics

- Editing Tasks
- Copying and Moving Tasks
- Moving and Reordering Task Lists
- Clearing Task Lists
- Deleting Tasks and Task Lists

You can use the Edit Task List dialog box to update 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 **Open the Manage Task Lists page.**

   See "Managing Task Lists" on page 158.

2 **Select a task list, select Actions, and then select Edit.**

3 **Select a task, select Actions, and then select Edit.**

4 **In Edit Task:**

- Modify the task name.

- Select another task type. For a description of all available types, see "Adding Tasks to Task Lists " on page 160:

5 For **Instructions**, modify instructions for completing the task.

6 **Optional**: To enter an end date for the task, select **End Date**, and enter the date and time by which users must complete this task.

7 **Optional**: To send email messages if tasks are not completed by the end date, select **End Date**.

   a. Select **Repeat Every** and enter a value.

   b. Select the frequency for email reminders.

8 **Optional**: To send email messages after the alert date and until the end date, select **Alert**:

   a. Select the month, day, year, time, and AM or PM.

   b. Select **Repeat Every** and enter a value.

   c. Select the frequency for email reminders.

9 **Optional**: To make completion of this task depend on completing a primary task, select **Dependency**.

10 **Optional:** To edit task properties, click **Property**.

11 Click **Save**.

12 Click **Close**.

Copying and Moving Tasks

➤ To copy or move tasks:

1 Open the **Manage Task Lists** page.

   See "Managing Task Lists" on page 158.

2 Perform an action:

   To copy tasks:

   a. Select a task list, then select **Actions**, and then select **Edit**.

   b. Select the task to copy, and then click **Save As**.

   c. Enter the name for the new task list.

   d. Click **OK**.

   To move tasks:

   a. Select a task list, then select **Actions**, and then select **Edit Task List**.

   b. Select a task, then select **Actions**, and then select **Edit Tasks**.

   c. Select the task that will be a child of another task, then select **Actions**, and then select **Cut**.

   d. Select the task that will be the parent task, then select **Actions**, and then select **Paste**.

> e.   Click **Save**.

To cut and paste tasks:

> a.   Select a task list, and then click **Edit**.
>
> b.   Select a task, and then click **Cut**.
>
> c.   To move the task to a new position, select the task to appear above it.
>
> d.   Click **Paste**.
>
> e.   Click **OK**.

3   Click **Close**.

Moving and Reordering Task Lists

➤ To move or reorder task lists:

1   Open the **Manage Task Lists** page.

See "Managing Task Lists" on page 158.

2   Perform a task:

To move a task list:

> a.   Select the folder with the task list to move.
>
> b.   Select a task list, then click **Move**.
>
> c.   Select the destination folder.
>
> d.   Click **OK**.

To reorder task lists:

> a.   Select a task list.
>
> b.   Click ⌃ or ⌄.

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 does not delete the dates.

➤ To clear task lists:

1   Open the **Manage Task Lists** page.

See "Managing Task Lists" on page 158.

2   Select the folder with the task list to clear, select the task list, and then click **Edit**.

3   From the **Clear** drop-down list, select an option:

- **Completion Status**: Clears completion status only
- **End 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 the arrow to the right of the **Clear** drop-down list.

5   Click **OK**.

Deleting Tasks and Task Lists

➤ To delete tasks and task lists:

1   Open the **Manage Task Lists** page.

See "Managing Task Lists" on page 158.

2   Perform a task:

- To delete a task, select the folder and task list with the task to delete, and then click **Edit**. Select the tasks to delete, then select **Actions**, and then select **Delete**.
- To delete a task list, select the folder with the task list to delete, and select the task list. Select **Actions**, and then select **Delete**.

3   Click **OK**.

Linking Tasks to Planning Pages

Use **Copy Link** to link tasks to Planning pages. You can copy and paste the URL address from a page in the Planning application to the instructions for a task. The task list user can then access the Planning page from that location.

➤ To copy a URL to a task:

1   Access the page in the Planning application to which to link a task.

2   Select **Tools**, then **Copy Link**.

3   Select a task list, then select **Edit Task List**.

4   Select a task, then select **Edit Task**.

5   Under **Task - Instructions**, use the browser to paste the URL link. For example, in the Microsoft Internet Explorer menu, select **Edit**, then **Paste**.

6   Click **Save**.

7   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 does not mean being able to assign tasks to someone else.

Adding Access to Task Lists

➤ To assign access to task lists:

1 Open the **Manage Task Lists** page.

   See "Managing Task Lists" on page 158.

2 Select the task list to modify.

3 In the **Task List** area, select **Actions**, and then select **Assign Access**.

4 Click **Add Access**.

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 Open the **Manage Task Lists** page.

   See "Managing Task Lists" on page 158.

2 Select the folder and task list to modify, then click **Assign Access**.

3 Perform a task:

To change access to task lists:

a. Select users or groups, and click **Edit Access**.

b. For **Type of Access**, select:

- **Assign**: 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 **Remove Access**.

b. Click **OK**.

4 Click **Close**.

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*.

Notes:

- Selected Copy Data settings are preserved for the current session only.

- Copied dimension members must be present in the selected plan types.

- Data must be copied into cells that can accept data. For example, you cannot copy data into read-only or dynamic cells.

- You can copy account annotations, supporting detail, and cell text. You cannot copy planning unit annotations.

- You cannot use this feature with attributes, so do not select attributes to be copied.

- Because this is an administrative function, Planning assumes you have full access to data you copy. You are not prevented from copying to planning units that are approved.

- This feature does not 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 Select **Administration**, then **Manage**, and then **Copy Data**.

2 In **Plan Type**, select a plan type, and then click **Go**.

You can copy from one plan type at a time. When you click **Go**, dimensions are displayed for this plan type.

3   For **Static Dimensions**, enter the members for the data intersections:

   a.   In **Dimension**, select a dimension from which to copy.

   b.   In **Members**, click **Member Selection** 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.

4   **Optional**: To add another static dimension to the list, click **Add Dimension** and enter dimension members. (To remove a dimension, select **None - Select a Dimension**. The dimension moves to the **Dimensions with Source and Destination** area.)

5   In **Dimensions with Source and Destination**, enter dimensions into which to copy data:

   a.   In **Source**, click **Member Selection**

   b.   In **Destination**, click **Member Selection**.

6   In **Copy Data Options**, select the type of information to copy.

7   Click **Copy Data**.

   Data is copied from one intersection to the other. If data is not copied successfully, a message is displayed. You can also check the log file.

8   To copy data for another plan type, select another plan type in step 2, and repeat the procedure.

Tip: To view the execution status of Copy Data, see "Checking Job Status" in the *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

Clearing Cell Details

You can clear these cell details for a plan type: 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 for Oracle Planning and Budgeting Cloud Service*.

Notes:

● You cannot delete planning unit annotations.

● Because this function is for administrators and interactive users, Planning assumes that you have full access to details you delete.

➤   To clear cell details:

1   Select **Administration**, then **Manage**, the **Clear Cell Details**.

2   For **Plan Type**, select a plan type, and click **Go**.

   When you click **Go**, dimensions are selectable for this plan type.

3   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 is not selected, Planning 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 do not select at least one Account member, no data is cleared, even if you select the **Yes, set value to #Missing** option. If you do not select members from the other dimensions, all data for the selected Account member is cleared across all other dimensions.

---

4 **Optional:** Further refine the data intersection by specifying more members:

● To select another dimension so you can select its members, click **Add Dimension**.

● To select all dimensions in the plan type, click **Add All Dimensions**.

Select members for the displayed dimensions.

5 Specify the type of information to delete by selecting at least one option from **Clear Options**.

6 Click **Clear**.

A Confirmation page displays your selections.

7 Click **Finish** to proceed, or **Back** to change your selections.

If Clear Cell Details is successful, data is deleted from the plan type. If data is not deleted successfully, a message is displayed. You can also check the log file.

8 **Optional:** To view the execution status of Clear Cell Details and review the information that was deleted, select **Tools**, then **Job Console**.

See "Checking Job Status" in the *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

9 To clear cell details for another plan type, select another plan type in step 2, and repeat the procedure.

10

Working with an Application

Managing Forms and Folders

Use the Form Management and the Business Rule Folders pages to manage folders and forms.

Watch this video to learn about managing forms and folders in Planning.

▶ Video

Task	Topic
Create folders	See "Creating Folders" on page 171
Move folders	See "Working with Folders" on page 172
Delete folders	See "Working with Folders" on page 172
Rename folders	See "Working with Folders" on page 172
Create forms	See "Creating Simple Forms" on page 60
Assign access to forms and folders	See "Assigning Access to Forms and Folders" on page 26
Move forms	See "Moving, Deleting, and Renaming Forms" on page 90
Delete forms	See "Moving, Deleting, and Renaming Forms" on page 90

To view all forms or business rules in a Calculation Manager folder, click the folder's name in the lefthand 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 cannot:

- Delete folders unless they are empty

- Select multiple folders

- Rename, move, or delete the top-level folder called, respectively, **Forms** and **CalcMgrRules**

➤ To create folders:

1   Perform an action:

   - For form folders: Select **Administration**, then **Manage**, and then **Forms and Ad Hoc Grids**.

   - For Calculation Manager business rule folders: Select **Administration**, then **Manage**, and then **Business Rule 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: Select **Administration**, then **Manage**, then **Forms and Ad Hoc Grids**.

   For business rule folders: Select **Administration**, then **Business Rule 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**.

About Setting Preferences

On the Preferences page, all users can set individual preferences. Administrators can specify global settings. Preference selections affect only the current application.

Preference options depend on user type. Planners and interactive user types can access Application Settings and Display Options. See *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

See:

Administrators can control which tabs are displayed for setting personal preferences, application defaults, and system settings.

Show Option	Description
Current Application Defaults	Accesses the Application Settings tab and Display Options tab. The values set on these tabs become application defaults. Users can override defaults, and can revert to defaults by selecting Use Application Defaults on preference tabs. See *Working with Planning for Oracle Planning and Budgeting Cloud Service*.
Advanced Settings	Accesses the System Settings tab. See "Specifying System Settings" on page 174.

Setting Personal Preferences

All users can set personal preferences in the current application. For Application settings, Display Options, and Printing Options, see *Working with Planning for Oracle Planning and Budgeting Cloud Service*.

➤ To set personal preferences:

1 Select **File**, and then **Preferences**.

2 Click **Planning**, and then perform an action:

- Select **Application Settings** to set email options, select an alias table, and set options for member selection and approvals.

- Select **Display Options** to set options for number formatting, page selection, warnings for large forms, and the number of dimensions to show on a page.

- Select **Printing Options** to specify how pages are printed.

- Select **User Variables Options** to limit the number of members that display on a form by setting a user variable. See "Managing User Variables" on page 100.

Selecting **Use Application Default** resets the value to the current application default.

Setting Application Defaults

Administrators can specify defaults for the current application. Although users can override these application defaults by setting preferences, they can restore application defaults by selecting **Use Application Default** where available. For information about the individual application

settings and display option defaults, see *Working with Planning for Oracle Planning and Budgeting Cloud Service.*

➤ To set application defaults:

1 Select **Administration**, then **Application**, then **Settings**.

2 Select **Current Application Defaults**.

3 Perform one action:

- Select **Application Settings** to specify defaults for:
    - ○ Email notifications for task lists, approvals, and job console
    - ○ The alias table used
    - ○ The kind of member and alias data to display, such as both name and alias, on the Member Selector dialog box
    - ○ Show planning units as aliases
    - ○ Show planning units that are not started
    - ○ The attribute dimension date format.

- Select **Display Options** to specify defaults for:
    - ○ Number formatting
    - ○ Remember selected page members
    - ○ Allow search when pages exceed a certain number
    - ○ Member indentation
    - ○ The number of grid rows and columns fetched to populate forms
    - ○ UI themes
    - ○ Date formats

4 Click **Save**.

Specifying System Settings

Only administrators can specify system-wide settings.

➤ To specify system settings:

1 Select **Administration**, then **Application**, then **Settings**.

2 Select **Advanced Settings**.

3 Select **System Settings**.

4 Set options:

Table 29 System Settings

Option	Description
Email Character Set	The character set for email messages: ● **UTF-8** ● **Regional Setting**
Business Rules Notification	Set to **Yes** to notify users or groups when rules (which are enabled for notification in Calculation Manager) are completed or encounter errors.
Notify These Users	If **Business Rules Notification** is enabled, select one user or one group to notify.
Enable Use of the Application for	Determine whether users can access the application in maintenance mode, such as during backups. See "Limiting Use of an Application During Maintenance" on page 175.
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**: Display all substitution variables ● **Display None**: Do not display substitution variables ● **Enable Filtering**: Display only substitution variables that are valid for the runtime prompt bug
Select User	Assign ownership of the application to another administrator.
Display Users' Full Names	**Yes**: Show full names (such as Victoria Hennings). **No**: Do not display full names. Only user IDs will be shown (such as VHennings).
Calculation Module	The module used to create, validate, deploy, and administer business rules. (Calculation Manager is the only option.)

Limiting Use of an Application During Maintenance

Administrators can withdraw and grant access to an application during maintenance. If users are logged on to the application and administrators withdraw their access, users are forced off the system.

➤ To limit use of an application:

1   Select **Administration**, then **Application**, and then **Settings**.

2   Select **Advanced Settings**, and then select **System Settings**.

3   For **Application Maintenance Mode**, select an option for **Enable Use of the Application for**:

● **All users**: All users who have access to the application can log on or continue working with the application.

● **Administrators**: Only administrators can log on. Other users are forced off and prevented from logging on until the option is reset to **All Users**.

4   If your selection is more restrictive than the current setting, click **OK**.

5   Click **Save**.

Setting Display Options

Administrators can set the default display options such as number formatting and page display options for an application.

➤ To set the display options for an application:

1 Select **Preferences** in the left hand pane, or select **File**, then **Preferences**, and then **Planning**.

2 Select **Display Options** to specify defaults for:

- Number formatting
- Remembering page member selection
- Allowing a search if the number of pages exceeds a set number
- Member indentation
- Consolidation operators
- Partial grid fetch
- UI themes and text size
- Date formats
- User image

3 Click **Save**.

Setting Printing Options

To set printing options, see .

Mapping an Application for Reporting

Subtopics

- About the Reporting Application
- Creating a Reporting Application
- Mapping a Planning Application to a Reporting Application
- Defining Application Mappings
- Defining Dimension Mappings
- Setting the Point of View
- Setting Data Options
- Validation Rules for Default Members
- Pushing Data
- Unsupported Features with Mapping Applications
- Mapping Applications and Substitution Variables

About the Reporting Application

A reporting application is a target application on which you can report on and aggregate Planning data. The primary use cases:

- The reporting application contains data from any source, such as a data warehouse. You want to push new Planning data to it and report on the data.
- You want to report on Smart Lists in Planning, which you cannot do in Planning.

You can map dimensions between a source Planning application and a reporting application to enable:

- Reporting on Planning data in a reporting application
- Aggregating and querying on Smart Lists, which are converted to regular dimensions in the reporting application
- Linking Planning data to multiple reporting applications for various consolidations

Creating a Reporting Application

➤ To create a reporting application:

1 **If it does not exist, create the target reporting application.**

   A reporting application can be either block storage or aggregate storage.

2 **In the reporting application, create the dimensions on which you want to report or aggregate.**

   When you later map the application, Planning automatically maps same-named dimensions. To report or query on Smart Lists, create dimensions in the reporting application that map to the Smart Lists in the source Planning application.

3 **Into the reporting application, load members on which you want to report or aggregate.**

See Chapter 5, "Importing and Exporting into a Planning Application."

Next:

- Map the source Planning application to the target reporting application.

  See "Mapping a Planning Application to a Reporting Application" on page 178.

- Push data from the source to the target application.

  See "Pushing Data to a Reporting Application" on page 181.

- Report on the data in the reporting application.

Mapping a Planning Application to a Reporting Application

➤ To map a Planning application to a reporting application:

1   **Create the reporting application.**

    See "Creating a Reporting Application" on page 177.

2   **In Planning, select Administration, then Map Reporting Application.**

3   **On the Reporting Application Mapping page, select an option:**

    - To create a mapping, click **New**.

    - To update a mapping, select the mapping and click **Edit**, **Delete**, or **Rename**. See "Defining Application Mappings" on page 178.

    - To refresh dimensions and members from the reporting application, click **Refresh**. When you are defining an application mapping, Refresh ensures that the dimensionality in the member selector for the target application reflects its actual dimensionality.

    - To push the data to the reporting application, click **Push Data**. See "Pushing Data to a Reporting Application" on page 181.

    - To update the mapping for Smart Lists, see "Synchronizing Smart Lists in Reporting Applications" on page 252.

Defining Application Mappings

➤ To define an application mapping:

1   **For new mappings:** Enter a name and description.

2   **Under Source Application, select a Plan Type from the available plan types for the current application.**

    The plan type drives the information that is pushed to the reporting application.

3   **Under Reporting Application, select the Essbase server on which the reporting application resides, and then select the target reporting application.**

4   **Click Next.**

See "Defining Dimension Mappings" on page 179.

Defining Dimension Mappings

On the **Map Dimensions** tab, you map the Planning dimensions on the left to the reporting application members on the right. If mappings are correct, data can be pushed when dimensions in both applications are mapped or have valid default members for storing data in the reporting application (see "Setting the Point of View" on page 180).

➤ To define dimension mappings:

1 For each Planning dimension, select a **Mapping Type**:

   ● **Dimension to dimension**: Displays the available unmapped dimensions in the Planning application. In order to push data, the dimension and all of its members must reside in both the source and target applications.

   Dimensions that are identical between the Planning and reporting applications (for example, Scenario to Scenario) are automatically mapped.

   ● **Smart List to Dimension**: Displays the available Smart Lists. When a Smart List is selected, the account members associated with it are displayed. If there is only one member, it is automatically selected.

   Note: Smart List labels are used for mapping to member names or member aliases in the reporting application.

2 In **Dimension/Smart List name**, select the name of the dimension or Smart List.

   With Smart List to dimension mapping, the source plan type must contain a dense Account dimension with at least one member associated with a Smart List.

   Note: When mapping Smart Lists to dimensions, you can synchronize Smart Lists to identify dimensions in reporting applications to which Smart Lists are mapped. This also adds level 0 members of the selected dimensions as new Smart List entries to the selected Smart Lists. See "Synchronizing Smart Lists in Reporting Applications" on page 252.

3 In **Member Selection**, select the member name by clicking 🔍.

   By default, Lev0Descendants is selected. You can select only level 0 members. If there is only one member, this member is automatically selected.

4 Click either:

   ● **Next** to display the Point of View. See "Setting the Point of View" on page 180.

   ● **Save** if all dimensions are mapped and there is no need for a POV. (If so, the **Next** button is grayed out.)

Setting the Point of View

The Point of View tab displays, for each application, dimensions that are not already mapped or that are missing from either application so that you can specify a POV. All POV dimensions must have only one member selected.

➤ To set the point of view:

1   On the top section of the **Point of View** page, specify a member for each unmapped Planning dimension. Either:

   ● Enter the member name.

   ● To select members that are not displayed, click , and then select the POV member.

   For rules on valid default members, see "Validation Rules for Default Members" on page 181.

2   In the bottom section, select a member for each unmapped reporting application POV member. Either:

   ● Enter the POV member name.

   ● To select members that are not displayed, click , and then select the POV member.

3   Click **Save**.

   Note: You can click **Save As** to save the application mapping with a new name, and then click **Save** in the **Save As** dialog box.

   The members are checked for validity. See "Validation Rules for Default Members" on page 181.

Setting Data Options

Use the Data Options tab to define how to merge relational data such as comments, attachments, and supporting detail when pushing data.

➤ To set data options:

1   Select an option:

   ● **Do not copy**—Do not copy relational data.

   ● **Ignore when multiple cells merge into one cell**—Do copy relational data if multiple cells form one cell.

   ● **Overwrite when multiple cells merge into one cell**—Replace existing cells and attachments with the latest supporting detail data when cells merge

- **Append multiple cell data into one cell**—If multiple cells merge into one cell, add, append or merge the supporting detail from all the source cells into the target cell.

2 Click **Save**, and then **Finish.**

Validation Rules for Default Members

Default members in the reporting application store the data that is pushed from the source Planning application. If any of the following constraints are not met, or if a dimension in either application is not mapped and has no valid default member, then a mapping is not valid and an error message is displayed.

Rules:

- If the reporting application is an aggregation storage database, then the default members must be level 0 members.

- If the reporting application 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 plan type 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 application.

Note: Mappings that were once valid can become invalid if dimensions, members, or Smart Lists are renamed, removed, or added. If a target plan type has a change in dimensionality, you must select the corresponding application mapping on the **Map Reporting Application** screen and click 🔁 to refresh the data.

Pushing Data

Subtopics

- Pushing Data to a Reporting Application
- Pushing Data from One Plan Type to Another Plan Type (Smart Push)

Pushing Data to a Reporting Application

After setting up application mappings, you can push data to a reporting application. Planning validates the selected application mappings, and then pushes the mapped Planning dimension data to the reporting application dimensions. You can also check the Job Console for the job status. See "Mapping a Planning Application to a Reporting Application" on page 178.

Note: If you enabled parent members for dynamic children in both a block storage and aggregate store application and added a dynamic member, you can use **Push Data** to successfully push data from the block storage to the aggregate storage application without refreshing the database.

Note: If the aggregate storage application is created outside of Planning with members that match the names of the dynamic members used in Planning, pushing data will fail because the system looks for the Essbase bucket name. This problem does not occur if the aggregate storage database is created within Planning.

➤ To push data to a reporting application:

1 Create the reporting application.

   See .

2 In Planning, select **Administration**, and then **Map Reporting Application**.

3 On the **Reporting Application Mapping** page, click **Push Data**.

4 Select an option:

   ● **Clear data on destination and push data**: Clears the data in the target reporting application before pushing data to it.

      When the target is an aggregate storage reporting application, note the following when using the **Clear data on destination and push data** option:

      ❍ Members with nonmatching names in the target reporting application 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 application mapping, 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 application. If at least one member name in the source application matches a member in the reporting application, this option proceeds without error. If at least one member does not match, the option cannot proceed.

      When the target is a block storage reporting application, to proceed successfully, the **Clear data on destination and push data** option requires these conditions:

      ❍ If you use member relationships, all member names in the source application must match all member names in the reporting application.

      ❍ If you map Smart Lists to dimensions, all Smart List entries in the source application must match all member names in the reporting application.

      ❍ If you map Smart Lists to dimensions, the Smart List entry label in the source Planning application must match the member name in the reporting application. If a Smart List entry label does not match a member name in the reporting

application, then the Smart List entry name must match the reporting application member name.

If the previous conditions are not met, the **Clear data on destination and push data** option cannot proceed.

- **Push data**: Pushes the data to the reporting application without first clearing the data in the target reporting application.

  If you map Smart Lists to dimensions, Smart List labels must match either member names or aliases in the reporting application. **Push data** does not work with Smart List entry names.

5   Click **OK**.

Pushing Data from One Plan Type to Another Plan Type (Smart Push)

Subtopics

- About Smart Push
- Considerations When Using Smart Push
- Configuring Smart Push for a Form
- Configuring Merge Options
- Pushing Data After Configuring Smart Push
- Viewing the Smart Push Status

About Smart Push

For more meaningful and complete reporting, planners can push comments, attachments, and supporting detail to multiple plan types. Users can then do more analysis on the planning data coming from the different plan types.

For example, assume that you have expense planning in one plan type and revenue planning in another plan type. Assume further that you use a reporting plan type to consolidate your reporting needs. Without Smart Push, the data from your plan types would be moved to the reporting plan type through scheduled jobs. Smart Push allows the data to be moved to the reporting plan type almost immediately.

You can push planning data from:

- Block Storage plan types to Aggregate Storage plan types
- Block Storage plan types to Block Storage plan types

Considerations When Using Smart Push

When using Smart Push, keep in mind the following considerations:

- Smart Push honors metadata and approvals security.
- Smart Push is not supported for forms from source Aggregate Storage plan types.

- Smart Push requires that at least one of the dense dimensions, account or period, is set as a dimension-to-dimension mapping in the application mapping definition.

- Numeric data across all members selected for dimensions listed in the unmapped section of the source plan type in the application mapping is not listed in the Smart Push definition.

- Whenever a target plan type has a change in dimensionality, you must select the corresponding application mapping on the **Map Reporting Application** screen, and then click to refresh the data.

Configuring Smart Push for a Form

Subtopics

- Configuring Smart Push for a Simple Form
- Configuring Smart Push for a Composite Form

Tip: Oracle recommends that you configure Smart Push to allow push only on some forms that act as summary forms in the BSO plan type so that users consolidate their actions before invoking reports. Automatic push should be limited because users tend to save data in a web application before finishing their data entry.

Configuring Smart Push for a Simple Form

➤ To configure Smart Push for a simple form:

1 In Planning and Budgeting Cloud Service Workspace, select **Navigate**, then **Applications**, then **Planning**, and then select an application.

2 In **Form Folder**, expand **Forms**, and then select a simple form.

3 With the simple form selected, select **Actions**, and then **Edit**.

4 Select the **Smart Push** tab.

The defined application mappings for the form are displayed. Click to add new application mappings if desired.

5 Expand each application mapping.

6 For each dimension shown under an application mapping, do one of the following to specify the Smart Push region for the dimension:

- Select **Use Form Context** to push data for all the members selected for the dimension on the **Layout** tab.

- Clear **Use Form Context**, and then click next to **Overwrite Selection** to select a member in the **Member Selection** dialog box.

  See Chapter 7, "Using the Member Selector" for information on selecting members.

Note: An application mapping definition can be overridden only on source plan type dimensions. Dimension members can be overridden if you take the context from a form by selecting **Use Form Context**. Dimension members can also be overridden if you specify members in the Member Section dialog box, or if you leave **Overwrite Selection** blank.

7   For each application mapping, define whether Smart Push will be automatic or manual.

   ● Select **Run on Save** to automatically push the data when the simple form is saved.

   ● Leave **Run on Save** unchecked to manually push the data.

   To manually push the data, planners must click the link under Smart Push Details in the left pane. For example, data map names associated with a form would be listed here. See "Pushing Data After Configuring Smart Push" on page 186.

8   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   In Planning and Budgeting Cloud Service Workspace, select **Navigate**, then **Applications**, then **Planning**, and then select an application.

2   In **Form Folder**, expand **Forms**, and then select a composite form.

3   With the composite form selected, select **Actions**, and then **Edit**.

4   Select the **Smart Push** tab.

   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 on Save** to automatically push the data when the composite form is saved.

   ● Leave **Run on Save** unchecked to manually push the data.

   If you select **Run on Save** to push the Smart Push details for a simple form when the composite form is saved, then regardless of whether you selected **Run on Save** for each application mapping in the simple form, all of the application 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 application mappings are executed regardless of whether **Run on Save** is selected for an application mapping in a simple form.

7   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 an application for reporting. See "Mapping an Application for Reporting" on page 177 and "Setting Data Options" on page 180.

Note: When using Smart Push, relational data such as Supporting Details, Date, Smart List, and Text cannot be merged. Smart List, Date, and Text are pushed only in cases of a a one-to one-mapping between source cells and target cells. Empty cells are not considered for relational data push.

Pushing Data After Configuring Smart Push

See "Pushing Data to a Reporting Application" on page 181 for information on how to push data.

How the data is pushed depends on whether you selected **Run on Save** when you configured Smart Push for a form.

● If you selected **Run on Save** when you configured Smart Push for a form, the data is pushed when you save the form.

● If you left **Run on Save** unchecked, with the form displayed in Planning, click the link under **Smart Push Details** *<form name>* in the left pane. For example, for a form named "Compensation Expenses,"you would click the link under **Smart Push Details Compensation Expenses**.

Note: Smart Push always clears data in the target region before pushing new data.

Note: If planners observe errors in Smart Push execution, the administrator must verify and fix how Smart Push is set up.

Viewing the Smart Push Status

To view the status of the data push, select **Tools**, and then **Job Console**. The Smart Push job will be displayed in the Job Console. If an error occurs, click the error link to troubleshoot the problem. If the data was pushed successfully, the Run Status is displayed as "Complete."

Unsupported Features with Mapping Applications

Mapping an application to a reporting application does not support:

● Aggregation storage database outlines with the "Duplicate Members Allowed" option selected

● User variables

● Attribute dimensions

● Attribute member selections

Mapping Applications and Substitution Variables

When mapping an application that contains substitution variables, note:

- Substitution variables are checked when you click Push Data, not while application mappings are defined.

- For dimension-to-dimension mappings on the **Map Dimensions** tab:

  ❍ The member selector displays the substitution variables that were defined for the Planning application.

  ❍ For the Planning application, the variable name that is selected or typed is passed and evaluated when you click **Push Data**.

  ❍ For the reporting application, the variable name is evaluated against the Planning application, and then the value is used in the Clear Data operation.

- For Point of View fields on the Point of View tab:

  ❍ For the Planning application, the member selector displays the substitution variables that were defined for the Planning application. The variable names that are selected or typed are passed and evaluated when you click Push Data.

  ❍ For the reporting application, the member selector displays the substitution variables defined only for the reporting application. It is 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 **Refresh** before editing or pushing data to synchronize reporting dimensions and members with Planning. For example, if a dimension or member was added to a reporting application, clicking **Refresh** makes the member visible in Planning. Refreshing changes from reporting application metadata may cause mappings to become invalid.

11

Working with Menus

Creating and Updating Menus

Administrators can create right-click 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 Console 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.

Watch this video to learn about associating menus in Planning.

▶ Video

➤ To create, edit, or delete menus:

1 Select **Administration**, then **Manage**, then **Menus**.

2 Perform one action:

- To create a menu, click **Create**, enter the menu's name, and then click **OK**.

- To change a menu, select it, and then click **Edit**.

- To delete menus, select them, click **Delete**, and then click **OK**.

Working with Menu Items

Edit Menu displays menu items on the current menu, including names, labels, required dimensions, icon, and type, such as URL, Form, Business Rule, Manage Approvals, Menu Header, Form, Job Console, and Copy Version.

➤ To work with menu items:

1 Select **Administration**, then **Manage**, and then **Menus**.

2 Select a menu and click **Edit**.

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, click **Edit**.

   ● To delete menu items, click **Delete**.

   ● To change the order of menu items within the same level, click the Up or Down Arrow. You can move multiple items.

      Use **Edit Menu Item** to define the menu item properties.

5 Click **Save**.

   Click **Save As** to save the current selections under a new menu name.

Adding or Changing Menu Items

➤ To define menu items:

1 Select **Administration**, then **Manage**, and then **Menus**.

2 Select a menu and then click **Edit**.

3 Select the menu item and then click **Edit** or **Add Sibling**.

4 Define the menu item:

Table 30   Menu Items

Item	Description
Menu Item	Enter a unique name containing only alphanumeric and underscore characters, with no special characters or spaces
Label	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 `File` directly or to the name of a resource, such as LABEL_FILE, which can be localized.

Item	Description
Icon	**Optional**: In context of the Planning 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`
Type	Select the menu item type to determine available Properties. No properties are available for Menu Header. ● **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. ● **URL**: Navigate to the specified URL ● **Business Rule**: Launch the selected business rule ● **Manage Approvals**: Move to Manage Approvals to work with planning 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 is not available. ● **Previous Form**: Create a menu that returns the user to the previous form. ● **Job Console**: Create a menu that opens the Job Console to view jobs for the current user for the specified Job Type and Status. ● **Copy Version**: Create a menu that opens Copy Version to enable the user to copy data for the current form.
Required Parameter	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:

Type	Options
Form	a. In **Form Folder**, select the folder containing the destination form. b. In **Form**, select the form.
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.

Business Rule	a. In **Plan Type**, select the plan type 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 Planning 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 is launched. This option enables administrators to provide meaningful messages to planners about the consequences of launching business rules.

Type	Options
Manage Approvals	Specify the planning 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.
Copy Version	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: a. In **Scenario**, select the scenario from which to copy. b. In **Copy From**, select the version that contains the data to copy. c. In **Copy To**, select the version to which to copy the data.
Job Console	Enable end users to view the Job Console for the specified job type and job. Select the following default values: ● **Job Type** ● **Status**

6　Click **Save**.

Working with Planning Application Administration

In This Chapter

About Creating an Application with Oracle Planning and Budgeting Cloud Service

You can create and delete a single application. You will perform these steps to create an application:

- On the landing page, indicate if you will build your application using the demo application provided, or using your own parameters.

- If you do not use the demo application, you will set up your application by defining the following and then importing metadata and data:

  ❍ Currency

  ❍ Calendar

  ❍ Dimensions and metadata

Creating an Application

Subtopics

- Creating a Custom Application
- Setting up the Currency
- Customizing the Plan Type Names
- Setting Up the Calendar
- Setting Up Custom Dimensions and Metadata
- Creating a Demo Application

Complete the procedures in this section to create an application manually. After you create the application, you can import metadata and then data.

Tip: You can quickly create an application using the demo application, called Vision, that is provided. See "Creating a Demo Application" on page 198.

Creating a Custom Application

➤ To create an application:

1 Select **Navigate**, then **Administer**, then **Planning Administration**.

2 Click **Create Application**, and then select one:

- **New Application**—Build your application from scratch. Continue to step 3.

- **Demo Application**—Build your application using the artifacts that the demo application, called Vision, provides. See "Creating a Demo Application" on page 198.

3 Enter the name of the application.

The name can contain up to eight characters and should adhere to the naming restrictions listed in Appendix A, "Naming Restrictions."

4 Enter a description of the application.

5 Set up the currency. See "Setting up the Currency" on page 194.

Setting up the Currency

Specify the default currency for entities in the application, and establish if the application supports multiple currencies. For detailed information about currency, see "Working with Dimensions" on page 206.

➤ To set up the currency:

1 In **Create Application**, from the **Currency** drop-down list, select the default currency for entities in the application.

2   Specify if the application supports multiple currencies by selecting **Yes** for a multicurrency application or **No** for a single-currency application.

> Note: By default, a new application is created with the Support Multiple Currencies option set to No. Select Yes to set up a multicurrency application.

After the application is created, you cannot change this option. Multiple currency support is available for level 0 members, regardless of base currency. If you select **Yes**, two additional dimensions are created, Currency and HSP_Rates.

3   Set up the calendar. See "Setting Up the Calendar" on page 195.

Customizing the Plan Type Names

You can customize the names for the plan types during application creation only. If no plan type names are specified during application creation, the block storage plan type name defaults to "Plan1" and the aggregate storage plan type name defaults to the application name.

> Caution!   After a plan type is added, it cannot be renamed or deleted.

➤ To customize the plan type names:

1   In **Plan Type : BSO**, enter a name for the block storage plan type.

2   In **Plan Type : ASO**, enter a name for the aggregate storage plan type.

Setting Up the Calendar

The calendar establishes the application's base time periods, first fiscal year and month, and total number of years. Select the base time period and monthly distribution pattern based on the number of fiscal weeks in a month. The base time period options are the bottom-level time periods in the application. You can create a custom base time period, such as weeks or days. Use monthly distribution patterns to determine how data entered into a summary time period is distributed or spread among the base time period that you select. Users can enter data into summary time periods, such as years or quarters. Planning distributes these values over the base time periods that constitute the summary time period.

For the fiscal year, you can 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 a Planning application, keep in mind that formula expressions such as [TPDate] and [FirstDate] produce different results if the application starts in the same calendar year or previous calendar year.

The following table provides examples of how the Fiscal Year First Month and the Fiscal Start Year options affect the calendar for the application, assuming the Fiscal Start Year is 2012.

Table 31 Examples of Fiscal Year First Month and Fiscal Start Year Calendar Options

Fiscal Year First Month	Fiscal Start Year	Period - Year	Years Dimension
January	Same Calendar Year	Jan-12 to Dec-12	FY12
July	Same Calendar Year	Jul-12 to Jun-13	FY12
July	Previous Calendar Year	Jul-11 to Jun-12	FY12
February	Same Calendar Year	Feb-12 to Jan-13	FY12
February	Previous Calendar Year	Feb-11 to Jan-12	FY12
December	Same Calendar Year	Dec-12 to Nov-13	FY12
December	Previous Calendar Year	Dec-11 to Nov-12	FY12

➤ To set up the calendar:

1   In **Create Application**, to set how calendars roll up, select a **Base Time Period** option:

- **12 Months**: Four quarters per year; months roll up into parent quarters and quarters into years.

- **Quarters**: Quarters roll up into years.

- **Custom**: A custom time period, such as weeks or days.

2   Select the **First Fiscal Year**.

This selection defines the starting fiscal year for the application. You cannot change it after creating the application. Before specifying the first fiscal year, consider how much historical data your organization needs and wants in the application.

3   Select the **Number of Fiscal Years** for the application.

This selection defines the number of years in the calendar. You can add more years to the calendar after the application is created.

4   Enter or select the **Fiscal Year Start Date**.

This selection is the start date in the fiscal year for the application, which depends on the First Fiscal Year option.

5   If you set the base time period as **12 Months**, 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**, Planning 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.

6 To create an All Years parent member that includes all years, select **Yes** for **All Years parent**.

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 parent member does not include the No Year member, if one is defined for the application.

7 Set up custom dimensions for the application. See "Setting Up Custom Dimensions and Metadata" on page 197.

Setting Up Custom Dimensions and Metadata

Dimensions categorize data values. You can create up to 13 user-defined, custom dimensions. Planning 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.

A multicurrency application include the Hsp_Rates dimension for storing exchange rates. This dimension includes these members and others that store currency rates: Hsp_InputValue: (stores data values) and Hsp_InputCurrency (stores currency types for data values).

For detailed information about custom dimensions and Hsp_Rates, see "Working with Dimensions" on page 206.

Note: In the **Metadata and Custom Dimensions** section of the **Create Application** window, 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 Metadata Flat File " on page 40.

➤ 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 more 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 Metadata and Data" on page 39) for incremental updates or for loading larger dimensions.

For instructions on creating flat files, see "Creating the Metadata Flat File " on page 40. 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**.

> **Note:** To subsequently launch the new application from Planning and Budgeting Cloud Service Workspace, select **Navigate**, then **Administer**, then **Planning Administration**, and then click **Launch Application**. The **Create Application** icon is no longer displayed because Planning supports only one application.

5   If you have more metadata to load, continue to "Importing Metadata" on page 39.

Creating a Demo Application

Planning provides a demo application, called Vision, that enables you to quickly create a Planning application with artifacts and data.

Note: To create an application manually, see "Creating a Custom Application" on page 194.

Watch this overview video for an introduction of the Planning demo application.

▶ Overview Video

➤   To create a demo application:

1   From Planning and Budgeting Cloud Service Workspace, select **Navigate**, then **Administer**, and then **Planning Administration**.

2   Click **Create Application**, and then select **Create Demo Application**.

Planning automatically creates the Vision application with artifacts and data. When application creation is completed, Planning displays an **Application created successfully** message.

3   Click **OK**.

4   You must refresh the landing page before you can launch the new Vision application. To refresh the landing page, choose an option:

●   Select **Navigate**, then **Application**, and then **Refresh**.

●   Close the **Planning and Budgeting Service** tab, then select **Navigate**, then **Administer**, and then **Planning Administration**.

> **Note:** To subsequently launch the new application from the landing page, select **Navigate**, then **Administer**, then **Planning Administration**, and then click **Launch Application**. The **Create Application** icon is no longer displayed because Planning supports only one application at a time.

5 To launch the Vision application in Planning and Budgeting Cloud Service Workspace, select **Navigate**, then **Applications**, then **Planning**, and then **Vision**.

Accessing an Existing Application

To access an existing application, select **Navigate**, then **Administer**, then **Planning Administration**, and then click **Launch Application**.

Deleting the Application

➤ To delete the application:

1 Select **Navigate**, then **Administer**, and then **Planning Administration**.

2 Click **Admin Tools**, and then **Delete Application**.

3 Respond to the confirmation message.

About Monitoring and Optimizing Performance Using the Application Monitor

Subtopics

- Assumptions
- How the Application Monitor Works
- Using the Application Monitor Graphs
- Launching the Application Monitor
- Modifying Artifacts for Optimal Performance

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 planning 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 cannot 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 plan types
- Dimensions
- Simple forms
- Standalone business rules
- Planning 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
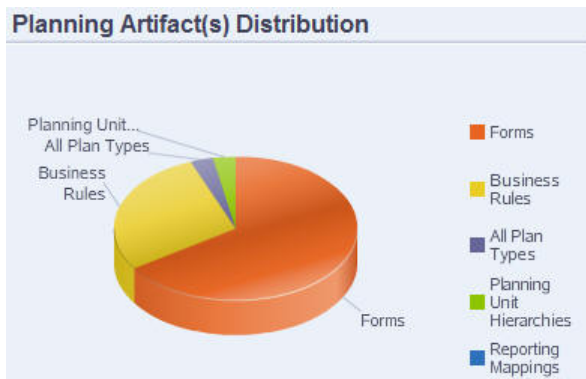
- **Filtered For <Artifact>**—Click the graph portions that represent the artifact status, and to view and drill down on artifacts with a particular status. For example, to view information for only planning unit hierarchies in error status, click the red portion of the graph.

Filtered Graph:

Filtered for Forms
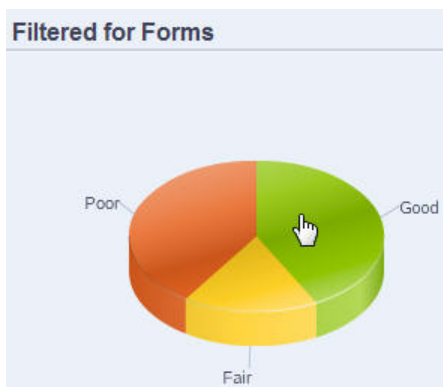
Launching the Application Monitor

➤ To launch the application monitor:

1 After logging in, select **Diagnostics** in the left hand pane.

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:

- To analyze and troubleshoot all application artifacts, select **Application Monitor - All Artifacts**. To the right, a pie graph depicts the distribution of artifacts in the application. The lower grid represents the integrity of all application artifacts. Yellow and red indicate artifacts that you should or must modify for optimal performance.

- To analyze specific types of artifacts, such as forms or planning unit hierarchies, select the type in **Diagnostics**. For example, to ensure that business rules are designed well and will not cause performance issues, click 📊 in **Diagnostics**. If a large portion of the graph is yellow or red, you should modify your rules.More information about all artifacts of that type display in the grid below.

- To "cherry pick" and analyze one particular artifact, find the artifact by specifying its name, start date, end date, and type in **Search**, select the artifact, and then click ➡. Information about the artifact displays in the lower grid.

> **Tip:** For information about what the pie graphs display and how to drill down to access more information, see "Using the Application Monitor Graphs" on page 200.

4   Click ▷ or select **Actions**, and then **Run Application Monitor**.

5   If the **View Details** column does not 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 Chapter 6, "Managing Forms."Chapter 6, "Managing 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 Planning and Budgeting Cloud Service* and "Understanding Runtime Prompts" on page 93.

- Planning unit hierarchies—Clicking a planning unit hierarchy artifact name launches the **Planning United Hierarchy Designer** in a new tab. Edit planning units as described in "Creating Planning Unit Hierarchies" on page 144.

- Plan types—Clicking the plan type 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 "Working with Dimensions" on page 206.

- Reporting Mappings—Clicking the reporting mapping artifact name launches the **Report Mapping Designer** in a new tab. Edit reporting mappings as described in "Mapping an Application for Reporting" on page 177.

Working with Alias Tables

You can create and update alias tables, and set a default alias table for the application. Follow naming conventions in Appendix A, "Naming Restrictions."

About Alias Tables

You can assign alternate names, or aliases, to Planning Account, Currency, Entity, Scenario, Period, Version, Years, and user-defined dimension members. Planning allows up to 10 aliases per dimension member, including the default alias.

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 is not restricted to these language combinations.

You can set alias tables to display members in an application. Planners can set alias tables in preferences.

Creating Alias Tables

➤ To create alias tables:

1 Select **Administration**, then **Manage**, then **Alias Tables**.

2 Click **Add**.

3 In **Add - Alias Table**, enter a name.

4 Click **OK**.

Editing or Renaming Alias Tables

➤ To edit or rename alias tables:

1 **Administration**, then **Manage**, then **Alias Tables**.

2   Select the alias table.

3   Click **Edit.**

4   For **Edit - Alias Table**, enter a name.

5   Click **OK.**

Deleting Alias Tables

➤   To delete alias tables:

1   **Administration**, then **Manage**, then **Alias Tables**.

2   Select the alias table.

> You cannot delete the default alias table.

3   Click **Delete**.

4   Click **OK**.

Clearing Alias Tables

You can clear the contents of alias tables.

➤   To clear alias tables:

1   **Administration**, then **Manage**, then **Alias Tables**.

2   Select the alias table to clear.

> Clearing the alias table removes the contents of the table but does not remove the table.

3   Click **Clear Values**.

4   Click **OK**.

Copying Alias Tables

➤   To copy alias tables:

1   **Administration**, then **Manage**, then **Alias Tables**.

2   Select the alias table.

3   Click **Copy**.

4   Select the destination alias table.

> The destination alias table must exist. Copying does not create tables.

5   Click **Copy**.

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 Select **Administration**, then **Application**, then **Settings**.

2 Select **Current Application Defaults**, and then select **Application Settings**.

3 In **Alias Table**, select a default alias table.

4 In **Member Name\Alias Display**, select the option that enables the kind of member data to be displayed on the Member Selector throughout your 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

5 Click **Save** or **Reset**.

Working with Dimensions

Subtopics

Dimension Overview

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

About Dimensions and Members

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 cannot be assigned to dense dimensions. Planning designates the Account and Period dimensions as dense, and the remaining dimensions as sparse. To optimize performance for sparse dimensions, Planning searches for and calculates only occupied data values in each dimension combination, reducing calculation time and disk usage. You can

modify these settings. See "About Reordering Dimensions" on page 37 and "Setting Dimension Density and Order" on page 219.

About Dimension Hierarchies

Dimension hierarchies define structural and mathematical relationships, and consolidations between members in the database. Relationships are represented graphically in a collapsible hierarchy diagram. The levels below the database 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.

Working with Dimension Hierarchies

Task	Topic
Specify or change dimension properties.	Click **Edit**.
Add a dimension.	Click **Add Dimension**.
Search for a dimension member.	See "Finding Dimensions or Members" on page 209.
Expand or collapse the dimension hierarchy.	Click **Expand** or **Collapse**.
Add or edit a dimension member.	Click **Add Child** or **Add Sibling**.
Move a dimension member.	See "Moving Members Within the Dimension Hierarchy" on page 209.
Delete a dimension member.	See "Deleting Members" on page 224.
Assign access to a dimension member.	See "Assigning Access to Members and Business Rules" on page 23.
View a member's ancestors.	Click **Show Ancestors**.

Expanding and Collapsing Dimension Hierarchies

➤ To expand dimensions or members:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 For **Dimension**, select the dimension and member to expand.

3 Perform one action:

- Click **Expand**.

- Click ±.

- Click the closed folder.

➤ To collapse dimensions or members:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 For **Dimension**, select the dimension to collapse.

3 Perform one action:

- Click **Collapse**.

- Press the **Left Arrow**.

- Click −.

- Click the open folders.

Navigating Dimension Hierarchies

- Press Up Arrow to move to the previous member.

- Press Down Arrow to move to the next member.

- In Page, enter the page to view and click Go or press Enter.

- Click Start, Prev, Next, or End to view other pages.

By default, 14 members are displayed per page. You can change this by setting preferences for Show the Specified Members on Each Dimensions Page.

Filtering the Dimension View by Plan Type

You can filter the dimension view by plan type. When you select a plan type, only members used in that plan type are displayed on the **Dimensions** page.

➤ To filter the dimension view by plan type:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 For **Plan Types**, select the plan type.

Planning displays only the members used in the selected plan type.

Finding Dimensions or Members

➤ To find dimension members in dimension hierarchies:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   For **Dimension**, select the dimension for the member.

3   For **Search**, select name, alias, or both.

4   Enter the member name, alias, or partial string for which to search.

5   Click **Search Down** or **Search Up** .

Sorting Members

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

➤ To sort members:

1   Select **Administration**, then **Manage**, then **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.

Moving Members Within the Dimension Hierarchy

You can move one member or a group of members in the same branch. If you move Account members whose Valid For Plan Type settings differ from their new parent, the moved member settings change to match the setting of the new parents. If you move members whose Source Plan Type settings differ from their new parent, the moved member Source Plan Type is reset to match the first valid plan type.

➤ To move members or branches among siblings:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   For **Dimension**, select the dimension for the members to move.

3   Select the member or branch to move.

4   Perform one action:

- Click ⋀ to move the member up one position.

- Click ⋁ to move the member down one position.

➤ To move members, including parents and children:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   For **Dimension**, select the dimension with the members to move.

3   Select the member or branch to move.

4   Click **Cut**.

    You cannot Cut members after adding or editing dimensions, navigating to different pages, deleting members, or logging off Planning. Not available for root dimension members.

5   Click the destination level under which to move the members.

6   Click **Paste**.

7   Click **OK**.

8   Update and validate business rules and reports.

Viewing a Member's Ancestors

➤ To view a member's ancestors:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   For **Dimension**, select a dimension.

3   Select the member in the dimension hierarchy.

4   Click **Show Ancestors**.

5   Click **Close**.

Determining Where Members Are Used in an Application

➤ To view where members are used in an application:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   Select the dimension whose member's usage you want to view.

3   Click **Show Usage**.

4   At the bottom of the **Member Usage** window, select where in the application to view the member's usage.

    Options are appropriate for the selected dimension member.

5   Click **Go**.

6   Click **Close**.

About Custom Dimensions

Planning includes two custom dimensions: Account and Entity. You can edit the names of these dimensions, and create up to 13 user-defined dimensions. 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

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 (do not aggregate, regardless of hierarchy)

Storage Options

Table 32   Storage Options

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

About Dynamic Calc

With dynamically calculated members, Planning 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, depending 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 does not 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:

- Do not use Dynamic Calc for base-level members for which users enter data.
- Do not 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

Do not 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 cannot 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 cannot assign attributes to label-only members.
- In a multicurrency application, you cannot 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!** | Do not design forms in which label-only parents follow their first child member, as you cannot save data in the first child member. Instead, create forms with label-only parents selected before their children, or do not 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 the Budgeting Process" on page 139).

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, Planning 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.

Base Currency

For a multicurrency 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).

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

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 33   Using Account Types

Account Type	Purpose
Expense	Cost of doing business
Revenue	Source of income
Asset	Company resource
Liability and Equity	Residual interest or obligation to creditors
Saved assumption	Centralized planning assumptions ensuring consistency across the application

Summary of Account Types

Table 34   Summary of Account Types

Account Type	Time Balance	Variance Reporting
Revenue	Flow	Non-Expense
Expense	Flow	Expense
Asset	Balance	Non-Expense
Liability	Balance	Non-Expense
Equity	Balance	Non-Expense
Saved Assumption	User-defined	User-defined

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

Time Balance Property

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

Table 35   Time Balance Properties

Time Balance Property	Description	Example
Flow	Aggregate of all values for a summary time period as a period total.	Jan: 10 Feb: 15 Mar: 20 Q1: 45
First	Beginning value in a summary time period as the period total.	Jan: 10 Feb: 15 Mar: 20 Q1: 10
Balance	Ending value in a summary time period as the period total.	Jan: 10 Feb: 15 Mar: 20 Q1: 20

Time Balance Property	Description	Example
Average	Average for all the child values in a summary time period as the period total.	Jan: 10 Feb: 15 Mar: 20 Q1: 15
Fill	The value set at the parent is filled into all its descendents. 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.	Jan: 10; Feb: 10; Mar: 10; Q1: 30
Weighted Average - Actual_Actual	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 is a leap year, the result is calculated: (10 * 31 + 15 * 29 + 20 * 31) / 91 = 15	Jan: 10 Feb: 15 Mar: 20 Q1: 15
Weighted Average - Actual_365	Weighted daily average, based on 365 days in a year, assuming that February has 28 days; does not 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 is not a leap year, the result is calculated: (10 * 31 + 15 * 28 + 20 * 31) / 90 = 15	Jan: 10 Feb: 15 Mar: 20 Q1: 15

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 Planning calculates and spreads data with the different Time Balance settings, see *Working with Planning for Oracle Planning and Budgeting Cloud Service.*

Account Types and Variance Reporting

An account's variance reporting property determines whether it is 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 36   Effect of Skip Options When Time Balance is Set to First

Skip Option	Description	Example
None	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.	Jan: 0 Feb: 20 Mar: 25 Q1: 0

Skip Option	Description	Example
Missing	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 the calculating parent values, so Q1 = second child (Feb), or 20.	Jan: #MISSING Feb: 20 Mar: 25 Q1: 20
Zeros	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.	Jan: 0 Feb: 20 Mar: 25 Q1: 20
Missing and Zeros	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.	Jan: 0 Feb: #MISSING Mar: 25 Q1: 25

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, Planning 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.

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.

Accounts, Entities, and Plan Types

By assigning plan types for Entity and Account members, you set to which plan types 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 plan type to a member prevents that member's children from accessing that plan type.

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

Entities and Plan Types

Typically entity members prepare different plans. When defining entity members, specify plan types for which they are valid. Because forms are associated with plan types, you can control which entity members can enter data for each plan type.

Accounts and Plan Types

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

About User-Defined Custom Dimensions

You can add up to 13 user-defined custom dimensions. For example, you could add a dimension called Project to budget operating expenses for each project. You define properties, including name, alias, plan type, security, attributes, and attribute values.

Caution!   You cannot delete custom dimensions after you create them.

User-defined custom dimensions differ from the Entity and Account dimensions in that you assign valid plan types at the dimension level, not at the member level. All members of a user-defined custom dimension are valid for plan types assigned at the dimension level.

Adding or Editing User-Defined Custom Dimensions

User-defined custom dimensions must conform to guidelines listed in Appendix A, "Naming Restrictions."

Table 37 Properties for User-Defined Custom Dimensions

Property	Value
Dimension	Enter a name that is unique across all dimensions.
Alias	**Optional**: Select an alias table. Enter a unique alternate name for the dimension.
Description	**Optional**: Enter a description.
Valid for Plan Types	Select plan types for which the dimension is valid. Clearing this option makes all members of the dimension invalid for the deselected plan type.
Apply Security	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.
Data Storage	Select a data storage option. The default is Never Share.

Watch this video to learn about adding custom dimensions in Planning.

▶ Video

➤ To add or change user-defined dimensions:

1   Select **Administration**, then **Manage**, then **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

Dimension properties must conform to guidelines listed in Appendix A, "Naming Restrictions."

Table 38 Dimension Properties

Property	Value
Dimension	Enter a dimension name.
Alias	**Optional**: Select an alias table and enter an alternate name of up to 80 characters. Follow the dimension naming restrictions.
Description	**Optional**: Enter a description.

Property	Value
Valid for Plan Types	Select plan types for which the dimension is valid. Not available for Entity or Account dimensions.
Apply Security	Allow security to be set on dimension members. If you do not select this option, there is no security on the dimension, and users can access its members without restriction. Must be selected before assigning access rights to dimension members.
Data Storage	Select data storage options.
Display Option	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.
Enable custom attribute display	Display available and selected attributes for dimensions with associated attributes. Enable custom attribute display for dimensions with attributes.

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   Select **Administration**, then **Manage**, and then **Dimensions**.

2   Select **Performance Settings**.

3   For each dimension, set its **Density** as **Dense** or **Sparse**.

    See "About Sparse and Dense Dimensions" on page 206.

4   Set the order of precedence by selecting a dimension and clicking ⌃ or ⌄ .

    See "About Reordering Dimensions" on page 37.

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.

Watch this video to learn about setting the evaluation order in Planning.

Video

➤ To set evaluation order:

1   Select **Administration**, then **Manage**, and then **Dimensions**.

2   Select **Evaluation Order**.

3   Select the plan type and click **Go**.

4   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 is not "Unspecified"). The data type "Unspecified" does not conflict with another data type.

5   If you select multiple dimensions, set the order of precedence by clicking or .

6   Click **Save**.

Working with Members

Subtopics

- About Dynamic Members
- Adding or Editing Members
- Deleting Members
- Deleting Parent Members
- Viewing Member Properties from Forms
- Working with Shared Members
- Creating Shared 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

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 Planning and Budgeting Cloud Service*.

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 Edit the parent member and select the option **Enable for Dynamic Children** (see).

- **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 cannot be used until the database is refreshed.

- **Optional**: Set the member property **Access Granted to Member Creator** (the default is Inherit).

2 Refresh the database to create the placeholders for dynamic members in the databases for the plan types in which the members are used.

3 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 Planning.

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 Planning and Budgeting Cloud Service*.

Notes:

- In Oracle Hyperion 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 cannot 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 Appendix A, "Naming Restrictions." Shared members must be consistent with "Working with Shared Members" on page 225.

Table 39 Member Properties

Property	Value
Name	Enter a name that is unique across all dimension members.
Description	**Optional**: Enter a description.
Alias Table	**Optional:** Select the alias table to store the alias name. Enter an alternate name for the member.
For Account members only: **Account Type**	Select the account type.
For Account members only: **Variance Reporting**	If the account type is Saved Assumptions, for Variance Reporting, select Expense or Non-Expense. Designate the saved assumption as a revenue, asset, liability, or equity account.
For Account members only: **Time Balance**	For Time Balance, select Flow, First, Balance, Average, Fill, Weighted Average - Actual_Actual, or Weighted Average - Actual_365. For descriptions, see "Time Balance Property" on page 214.
For Account members only: **Skip**	If the account type is Asset, Equity and Liability, select None, Missing, Zeros, or Missing and Zeros. For descriptions, see "Setting Account Calculations for Zeros and Missing Values" on page 215.
For Account members only: **Exchange Rate Type**	For Exchange Rate Type, select an option.
For Account members only: **Data Type**	For Data Type, select a data type.
Distribution	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.
Hierarchy Type	Hierarchy Type is available for dimensions bound to an aggregate storage plan type. Aggregate storage dimensions are automatically enabled to support multiple hierarchies. The first hierarchy in a multiple hierarchy dimension must be stored. **Note:** For members with a stored hierarchy type, the only valid plan type 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 plan type 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.

Property	Value
For Entity members only: **Base Currency**	Select the Entity member's base currency.
Display Option	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.
Plan Type	Select plan types for which the member is valid. **Note:** A member can belong to both aggregate storage and block storage plan types. Select an aggregation option for each selected plan type. You can select a source plan only if multiple plan types are valid for the member. Only plan types and aggregation options for which the member's parent is valid are available. If the parent is not valid for a plan type or aggregation option, neither is the child member. Deselecting a plan type 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 plan type 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 plan type is the source plan type. Members of a custom dimension can set usage by plan type, similar to the Account and Entity dimensions.
For Entity members only: Base Currency	Select the base currency for the Entity member.
For Account members only: Source Plan Type	Select the source plan type for the member. A shared member is a pointer to the base member and is not stored; this is disabled for shared members. The Source Plan Type of a shared Account member matches the source plan type of the base member, even though Source Plan field is unavailable because it does not apply to shared members.
Smart Lists	**Optional**: Select a Smart List to associate with the member.
Enable for Dynamic Children	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" on page 220).
Number of Possible Dynamic Children	This option is available if **Enable for Dynamic Children** is selected. Enter the maximum number of dynamically-added members that users can create. The default is 10.
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 do not retroactively affect dynamic members.

Watch this video to learn about adding Entity dimension members in Planning.

▶ Video

➤ To add or edit members:

1 Select **Administration**, then **Manage**, and then **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 **Add Child**.

- To add a sibling, select the level of the dimension hierarchy to which to add a sibling and click **Add Sibling**.

- To edit a member, select that member from the dimension hierarchy and press **Enter** or click **Edit**.

Note: To add an All Years parent member that includes all year members, select the Years dimension and then click **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 does not include the No Year member, if one is defined for the application.

4 On **Member Properties**, set or change member properties described in the table above.

If you do not 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 Members and Business Rules" on page 23.

- Specify attributes.

Deleting Members

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

Before deleting members, understand where in the application they are used (in which forms, planning units, exchange rates, and so on) by using Show Usage.

You must delete the entity member throughout Planning 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 planning units for the subtree (by excluding the root member) for all scenarios and versions. See "Review Process" on page 139.

➤ To delete members:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select the dimension whose member you want to delete.

3 From the dimension hierarchy, select the entity member to delete.

4 Click **Delete**.

Deleting a base member also deletes its shared members.

5 Click **OK**.

6 Update and validate business rules and reports.

Deleting Parent Members

Data values are identified by a set of dimension member values and a plan type. Deleting dimension members or deselecting the plan type results in data loss when refreshing the application.

➤ To delete a parent member and all its descendants from the dimension hierarchy:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select the dimension whose member and descendants to delete.

3 Select the member whose branch to delete.

4 Click **Delete**.

5 Click **OK**.

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 a Planning 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 plan types 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 cannot be moved to another parent member. You must delete shared members and recreate them under different parent members. Shared members must be at the lowest level (level zero) in the hierarchy and cannot have children. 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 Smart View.

Creating Shared Members

You create shared members the same way as other members, with these differences:

- The base member cannot be the parent of the shared member.
- You cannot 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 cannot assign attributes to label-only members. Attribute dimensions do not have aggregation properties because parents are dynamically calculated.

The Account dimension is usually defined as dense, so you cannot assign attributes to it unless it is changed to sparse for all plan types. 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" on page 227. Attribute names must conform to guidelines listed in Appendix A, "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 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select a sparse dimension for which to define an attribute, attribute value, or alias.

   Only sparse dimensions can contain attributes.

3 Select the top level in the dimension hierarchy, and click **Edit**.

4 In the Dimension Properties dialog box, click **Custom Attributes**.

If the dimension is not sparse, **Custom Attributes** is not available.

5   Select options.

- To create attributes, click **Create**. Type an attribute name, and select a data type: **Text**, **Date**, **Boolean**, or **Numeric**. See "Understanding Attribute Data Types" on page 227. You cannot modify the data type after the attribute is created.

   Plan type options are available for Entity dimension attributes only. You cannot change this setting after the attribute is created.

- To modify attributes, click **Modify**, and update the attribute name.

- To set aliases for attributes, select an attribute and an attribute value, click **Alias**. Select an alias table, type an alias name, and click **Close**.

6   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.

7   Update and validate business rules and reports.

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 preferences.

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 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select the sparse dimension for which to delete an attribute, and click **Edit**.

3 Click **Custom Attributes**.

4 Select the attribute to delete.

5 Above the **Attributes** column, click **Delete**.

6 Click **OK**.

7 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

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 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select the sparse dimension for which to create an attribute value.

3 Select the top level in the dimension hierarchy.

4 Click **Custom Attributes**.

5 On the **Manage Attributes and Values** page, select the attribute for which to specify a value.

6 Above the **Attribute Values** column, click **Create**. If the options are available, you can click **Add Child** or **Add Sibling**.

7 On **Create Attribute Value**, in **Name**, enter a name.

8 Press **Enter** or **Save**.

9 Click **Cancel**.

Assigning Attribute Values to Members

You can assign attribute values members of a dimension that are defined as sparse for all plan types. Attribute values must be assigned to the same-level sparse dimension members. Otherwise, errors display during refresh.

➤ To assign attribute values to members:

1   Select **Administration**, then **Manage**, then **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 **Edit**.

    For members assigned attribute values: Click **View** to change a member's attribute value.

5   Select **Attribute Values**.

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 《《.

8   Click **Save**.

Modifying Attribute Values

➤ To modify attribute values:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   Select the sparse dimension for which to modify an attribute value.

3   Select the top level in the dimension hierarchy.

4   Click **Custom Attributes**.

5   For **Attributes**, select the attribute containing the value to modify.

6   For **Attribute Values**, select the attribute value.

7   Above **Attribute Values**, click **Modify**.

8   On **Modify Attribute Value**, in **Name**, enter a name.

9   Click **Save**.

Deleting Attribute Values

When you delete an attribute value, it is removed from custom dimension members to which it is assigned.

➤ To delete attribute values:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select the sparse dimension containing the attribute for which to delete a value.

3 Select the top level in the dimension hierarchy.

4 Click **Custom Attributes**.

5 For **Attributes**, select the attribute containing attribute values to delete.

6 For **Attribute Values**, select attribute values to delete.

   To select all attribute values for deletion, select **Attribute Values**.

7 Above the **Attribute Values** column, click **Delete**.

8 Click **OK**.

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 Planning database. Use the Years dimension to add years to the calendar.

Table 40 Calendar Tasks

Task	See Topic
Define how years roll up.	"Defining How Calendars Roll Up" on page 230.
Create and edit Summary Time Periods.	"Creating and Editing Summary Time Periods" on page 231.
Delete Summary Time Periods.	"Deleting Summary Time Periods" on page 231.
Work with the fiscal year and period hierarchy.	"Working with the Years Dimension" on page 232.

Defining How Calendars Roll Up

Table 41 Calendar Roll Up

Base Time Period	Roll Up
12 Months	Four quarters are created per year. Months roll up into parent quarters and quarters roll up into years.
Quarters	Quarters roll up into years.
Custom	No default rollup structures. A flat list of the custom base time periods displays.

After the application calendar is created, you cannot 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 cannot change the order of base time periods or skip base time periods. The range cannot extend beyond the current fiscal year.

You must work from the top of the hierarchy to the bottom when creating summary time periods. (Otherwise, Planning views the rollup structure as asymmetrical and you cannot 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   Select **Administration**, then **Manage**, then **Dimensions**.

2   Select **Period**.

3   Perform one action:

- To add a time period, select the level in the dimension hierarchy above which to add, and click **Add**.

- To edit a time period, select the time period and click **Edit**.

4   For **Name**, enter or change the name for the summary time period.

5   **Optional**: For **Description**, enter a description.

6   **Optional**: For **Alias**, select an alias table to use. Enter an alias.

The default table is used if you do not select one.

7   For **Start Period**, select the starting period.

The range cannot 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 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Period**.

3 For **Period hierarchy**, select the summary time period to delete.

   You cannot delete base time periods.

4 Click **Delete**.

5 Click **OK**.

Working with the Years Dimension

Use the Years dimension to work with calendar years.

Table 42 Years Tasks

Task	See Topic
Add years to the calendar.	"Adding Years to the Calendar" on page 232.
Add or update the description and alias for a year.	"Editing Year Information" on page 233.
Set the fiscal year and work with years.	"Setting the Fiscal Year" on page 233.
Add an All Years parent member that includes all Years members (except No Year, if that member exists).	"Adding or Editing Members" on page 222

Adding Years to the Calendar

You can add years to the calendar, but you cannot reduce the number of calendar years without creating a database.

➤ To add years to the calendar:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Years**.

3 Click **Add Years**.

4 For **Number of Years to Add**, enter the number of years to add to the calendar.

5 Click **Add Years**.

> **Note:** To add an All Years parent member that includes all years members, click **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. (This parent member does not include the No Year member, if one exists.)

Setting the Fiscal Year

You can change the time period or current year.

➤ To change the current year or time period:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Years** or **Period.**

3 Click **Options**.

4 On **Set Current Period and Year**, from **Current Year**, select the current year.

Current Month and Current Year set defaults for the Month and Year when scenarios are created. For example, if Current Year is set to FY08 and Current Month is set to Aug, when users create scenarios, these values display as defaults in the Start Yr, Start Period, End Yr, End Period fields.

5 For **Current Time Period**, select the current period.

6 Click **OK**.

Editing Year Information

You can add or update the description and alias for a year.

➤ To edit years:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Years**.

3 Click **Edit**.

4 Enter a description for the year.

5 For **Alias Table**, select the alias table to use, and then enter an alias name.

6 Click **Save**.

Renaming Time Periods

You can rename root-level, base time periods, and user-defined summary time periods.

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 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Period.**

3 Select the summary time period.

4   Click **Edit**.

5   For **Alias Table**, select the alias table to use.

6   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   Select **Administration**, then **Manage**, then **Dimensions**.

2   For **Dimensions**, select **Period**.

3   Select the first member, **BegBalance**.

4   Click **Edit**.

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

●   Whether a triangulation currency converts currencies

●   When currency conversions occur

Enabling Multiple Currencies

If an application supports multiple currencies, you can enable multiple currencies per entity on forms. See "Setting Form Layout" on page 61. When selecting business rules for forms, you can select the Calculate Currencies business rule to convert values among the available currencies. See "Selecting Business Rules" on page 91.

Working with Multiple Currencies

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 cannot 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 does not 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 calc scripts only if the time periods have default names, TP 1 through 500. Otherwise, the conversion calc script you try to create exceeds the 64K limit.

- User-defined currency conversion calc scripts created when a database is created or refreshed may be available in Smart View, depending on user access. When using user-defined currency conversion calc scripts, Oracle recommends changing the order so the currency conversion calc script is first, before Calculate Form.

- Currency codes associated with input values are stored as numeric values. These codes are calculated in dimension formulas, calc 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 is not 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 is not 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

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:

- Does not 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

These exchange rates 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.

Planning supports currency conversion by triangulation through a triangulation currency.

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

A reporting currency is the currency in which your company prepares financial statements. Planning 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, or has a conversion or exchange relationship with other currencies.

➤ To see how currencies are used:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Currencies**.

3 Select the currency for which you want information.

4 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

- 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 is a reporting currency

➤ To create currencies:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Currency**.

3 Click **Add**.

4 From **Create Currency**:

- To add a predefined currency, select **Select standard Currencies**.

- To create a currency, select **Create new currency** and specify properties:

  ❍ For **Code**, enter an abbreviation or identifier of up to three characters.

  ❍ Optional: For **Description**, enter a name, such as Japanese yen.

  ❍ For **Symbol**, enter a symbol or select a symbol from the list.

  ❍ 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**, 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.

5 Optional: Select **Reporting Currency** (see "Working with Multiple Currencies" on page 235).

6 Optional: For **Thousands Separator**, select how to display the thousands separator (it must differ from the decimal separator).

7 Optional: For **Decimal Separator**, select how to display numbers with decimal values (it must differ from the thousands separator).

8 Optional: For **Negative Sign**, select how to display negative numbers:

- **Prefixed minus**: -1000.

- **Suffixed minus**: 1000-

- **Parentheses**: (1000)

9 Optional: For **Negative Color**, select the display color.

10 Optional: Select the type of **Data Storage**.

11 Optional: Select **Two Pass Calculation**.

12 Optional: Select the **Data Type**.

13 Optional: Select a **Smart List**.

14 Click **Save**.

Editing Currencies

➤ To edit currencies:

1 Select **Administration**, then **Manage**, and then **Dimensions**.

2   Select **Currency**.

3   Select the currency to edit.

4   Click **Edit**.

5   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.

  Note: Administrators can override this setting for forms. See "Setting Form Precision and Other Options" on page 68.

- To specify the currency as a reporting currency, select **Reporting Currency**.

  See "Working with Multiple Currencies" on page 235.

- 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" on page 237).

- For **Negative Color**, select the display color.

6   Click **Save**.

Deleting Currencies

You cannot delete the default currency.

➤   To delete currencies:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   For **Dimension**, select **Currencies**.

3   For **Currency**, select the currency to delete.

4   Click **Show Usage** to determine if the currency is the default currency, a triangulation currency, or associated with an entity. You cannot delete a currency that meets these criteria.

If you delete a currency defined in the exchange rate table, it is deleted from the table.

5   Click **Close**, **OK**, **Delete**, and **OK**.

6   Update and validate business rules and reports.

Specifying Exchange Rates

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

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 plan types. See "Creating and Refreshing Application Databases" on page 32.

Hsp_Rates Dimension

A 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" on page 219).

Triangulation

Planning 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 cannot 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 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.

Task	Topic
Create a scenario.	See "Creating Scenarios" on page 242.
Edit a scenario.	See "Editing Scenarios" on page 243.
Delete a scenario.	See "Deleting Scenarios" on page 243.
Copy a scenario.	See "Copying Scenarios" on page 243.

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 an 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 Scenarios

➤ To create scenarios:

1 Select **Administration**, then **Manage**, and then **Dimensions**.

2 Select **Scenarios**.

3 Click **Add Child**.

4 For **Scenario**, enter a name.

5 **Optional**: For **Description**, enter a description.

6 For **Start Yr.**, **Start Period**, **End Yr.**, and **End Period**, select the time period to associate with the scenario.

7 **Optional**: For **Exchange Rate Table**, select an exchange rate table to associate with the scenario.

   If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversions.

8 **Optional**: For **Alias**, select an alias table to associate with the scenario, and enter a description.

9 **Optional**: Select **Include BegBal as Time Period** to include the BegBalance time period in this scenario for currency conversion.

10 **Optional**: Select **Enabled for Process Management** to include this scenario in approvals.

11 Click **Save**.

Editing Scenarios

➤ To modify scenarios:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 For **Dimension**, select **Scenario**.

3 Select the scenario to edit.

4 Click **Edit**.

5 **Optional**: For **Scenario**, enter a name.

6 **Optional**: For **Description**, enter a description.

7 For **Start Yr.**, **Start Period**, **End Yr.**, and **End Period**, select the time period to associate with the scenario.

8 **Optional**: For **Exchange Rate Table**, select an exchange rate table to associate with the scenario.

  If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversion.

9 **Optional**: For **Alias**, select an alias table to associate with the scenario, and enter the description.

10 **Optional**: Select **Include BegBal as Time Period** to include the BegBalance time period in this scenario for currency conversion.

11 **Optional**: Select **Enabled for Process Management** to use this scenario in approvals.

12 Click **Save**.

Deleting Scenarios

When you delete scenarios, all planning units that use the scenario (including data) are deleted. You cannot delete scenarios used in planning 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 delete scenarios:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Scenarios**.

3 Select the scenarios to delete. At least one scenario must remain in the application.

4 Click **Delete**.

5 Click **OK**.

6 Update and validate business rules and reports.

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 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Scenarios**.

3 Select the scenario to copy.

4 Click **Copy Scenario**.

5 For **Copy to Scenario**, enter a name.

6 **Optional**: For **Description**, enter a description.

7 For **Start Yr.**, **Start Period**, **End Yr.**, and **End Period**, select the time period to associate with the scenario.

8 **Optional**: For **Exchange Rate Table**, select an exchange rate table to associate with the scenario.

If an application uses multiple currencies, associate a scenario with an exchange rate table to enable currency conversion.

9 **Optional**: For **Alias Table**, select an alias table to associate with the scenario, and enter a description.

10 **Optional**: Select **Include BegBal as Time Period** to include the BegBalance time period in this scenario for currency conversion.

11 **Optional**: Select **Enabled for Process Management** to include this scenario in approvals.

12 Click **Save**.

Specifying Versions

Use versions to group data used by an application.

Task	Topic
Create a version.	See "Creating Versions" on page 245.
Edit a version.	See "Editing a Version" on page 245.
Delete a version.	See "Deleting Versions" on page 246.

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 do not 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 Versions

➤ To create versions:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select **Version**.

3 Click **Add Child**.

4 For **Version**, enter the name of the version you are adding to the application.

5 For **Type**, 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 **Optional**: For **Description**, enter a description.

7 **Optional**: For **Alias Table**, select an alias table to associate with the version, and enter a description.

8 **Optional**: Select **Enabled for Process Management** to include this version in approvals.

This option is not available for target versions.

9 **Optional**: Select **Enable for Sandboxes** so this member can be used for sandboxes.

For more information on sandboxes, see *Administering Oracle Planning and Budgeting Cloud Service Using the Simplified Interface*.

10 Click **Save**.

Editing a Version

You can change the version name and access rights.

➤ To modify versions:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2   Select **Version**.

3   Select the version to edit.

4   Click **Edit**.

5   **Optional**: For **Version**, modify the name of the version.

6   **Optional**: For **Typelist**, modify 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.

7   **Optional**: For **Description**, enter a description.

8   **Optional**: For **Alias Table**, select an alias table to associate with the version, and enter a description.

9   **Optional**: Select **Enabled for Process Management** to include this version in approvals.

   This option is not available for target versions.

10   **Optional**: Select **Enable for Sandboxes** so this member can be used for sandboxes.

   For more information on sandboxes, see *Administering Oracle Planning and Budgeting Cloud Service Using the Simplified Interface*.

11   Click **Save**.

12   Update and validate business rules and reports.

Deleting Versions

You cannot delete versions that are used in planning 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 delete versions:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   Select **Version**.

3   Select the versions to delete.

4   Click **Delete**.

5   Click **OK**.

   If planning units are started, you cannot delete versions associated with them.

6   Update and validate business rules and reports.

Displaying Versions

➤ To display versions:

1   Select **Administration**, then **Manage**, then **Dimensions**.

2   Select **Version**.

3   For **Display**, select the version types to display.

- **Standard Target**: Values are entered from the parent level down.

- **Standard Bottom Up**: Values are entered at the lowest member level and aggregated up.

Sorting Version and Scenario Members

You can sort version and scenario members in ascending or descending order. Sorting members affects the outline itself.

➤ To sort version and scenario members:

1   On **Dimensions**, select the Scenario or Version dimension.

2   For **Sort**:

- To sort by ascending order, click ▤↑.

- To sort by descending order, click ▼↓.

3   Click **OK**.

The next time you create or refresh the database, the outline is generated with members placed in the same order as on the Dimensions tab.

Moving Scenario and Version Members in the Dimension Hierarchy

You can change the order of scenario and version members in the dimension hierarchy.

➤ To move a member's position in the dimension hierarchy:

1   On **Dimensions**, select the scenario or version to move.

2   Perform one action:

- To move the member up, click ▣.

- To move the member down, click ▣.

3   Click **OK**.

The next time you create or refresh the database, the outline is generated with members placed in the same order as displayed on the Dimensions tab.

Adding an Aggregate Storage Outline to a Planning Application

Subtopics

- About Aggregate Storage
- Aggregate Storage Outline Plan Type Characteristics
- Process for Adding an Aggregate Storage Database to a Planning Application
- Adding Plan Types

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 Plan Type Characteristics

- Planning does not generate XREFs on aggregate storage databases. XREFs can only be generated on block storage databases.

- Because Planning does not 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 a Planning Application

To add an aggregate storage database to a Planning Application:

1. Create an aggregate storage plan type. Perform one task:

    - Create an aggregate storage plan type during application creation.

        See "Creating an Application" on page 194.

    - Add a new plan type using the plan type editor.

2. Add dimensions to the aggregate storage plan type. See "Working with Dimensions" on page 206.

    Note: If a Currency, Years, Scenario, or Version dimension is valid for an aggregate storage plan type, the dimension members are also valid for an aggregate storage plan type.

3. Add dimension members. See "Working with Members" on page 220.

4. Refresh the outline for the Planning application. See "Creating and Refreshing Application Databases" on page 32.

5. Create a form using the dimensions associated with the aggregate storage database. See Chapter 6, "Managing Forms."

Adding Plan Types

You can add a plan type by selecting **Administration**, then **Manage**, and then **Plan Types**.

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

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

➤ To add a plan type using the plan type editor:

1 From Planning, select **Administration**, then **Manage**, and then **Plan Types**.

2 Click **Add Plan Type**, and then complete the plan type details.

> Note: For an aggregate storage plan type, you must specify an application name to contain the database since an aggregate storage database must reside in its own application.

3 Click **Save**.

Setting up Dynamic Time Series Members

You can use Dynamic Time Series (DTS) members to create reports that show period-to-date data, such as quarter-to-date expenses. DTS members are created automatically during application creation, and can be used with members of the Period dimension. To set up DTS, you enable a predefined DTS 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 member and associate it with generation number 2. You can then use the Q-T-D DTS member to calculate monthly values up to the current month in the quarter.

> Note: DTS is not supported for the Period dimension in an aggregate storage application.

Planning provides eight predefined DTS members:

● H-T-D: History-to-date

● Y-T-D: Year-to-date

● S-T-D: Season-to-date

● P-T-D: Period-to-date

- Q-T-D: Quarter-to-date
- M-T-D: Month-to-date
- W-T-D: Week-to-date
- D-T-D: Day-to-date

The DTS 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 day-to date (D-T-D), week-to-date (W-T-D), month-to-date (M-T-D), quarter-to-date (Q-T-D), and year-to-date (Y-T-D) information. If the database contains monthly data for the past 5 years, you can report year-to-date (Y-T-D) and history-to-date (H-T-D) information, up to a specific year. If the database tracks data for seasonal time periods, you can report period-to-date (P-T-D) or season-to-date (S-T-D) 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   Select **Administration**, then **Manage**, then **Dimensions**.

2   Select the Period dimension, and click **DTS**.

3   Select **Enabled** for the DTS series to use: **H-T-D**, **Y-T-D**, **S-T-D**, **P-T-D**, **Q-T-D**, **M-T-D**, **W-T-D**, or **D-T-D**.

4   Select a generation.

    The number of generations displayed depends on the number of generations in the time dimension. You cannot associate DTS members with the highest generation (the dimension root).

5   **Optional**: Select an alias table and type an alias name. (If necessary, resize the window to view the fields.)

    See "Working with Alias Tables" on page 203.

6   Click **Save**.

Additional Supported Planning Application Features

Additional Planning application features are supported for Planning 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" on page 207). You can also use shared members for these dimensions, and can set two pass calculations at the root level for all dimensions. For example, you can:

Feature	More Information
In Scenario and Version dimensions, create hierarchies and use shared members. If you assign children to bottom-up versions, these versions display as read-only parents on forms.	See "Setting Up Scenarios" on page 241 and "Specifying Versions" on page 244.
In the Period dimension, create alternate hierarchies 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 BegBalance, can be set to any valid consolidation operator. For example, it can be set to + instead of ~ (ignore).	See "Working with the Years Dimension" on page 232, "Editing the BegBalance Member" on page 234, and "Editing Exchange Rate Tables" on page 34.
Turn on two pass calculation at the root level, for example, for Account. **Caution!** Two pass calculation is ignored on any non-Account member not set to Dynamic Calc. When using this setting, consider the impact on currency conversion scripts.	See "Adding or Editing Members" on page 222.
For attributes, create hierarchies and assign aliases.	See "Working with Attributes" on page 226.

Note: For a multicurrency application, the pre-built currency conversion calc scripts does not function properly if you change data storage to dynamic for any Scenario, Version, Period, or Years member, or for certain dimension roots (such as Entity, Version, Currency, and custom dimensions). When changing data storage, consider the impact on currency conversion scripts.

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.

Using Smart Lists, UDAs, and Member Formulas

Subtopics

- Working with Smart Lists
- Working with UDAs
- Working with Member Formulas

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 cannot 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, described here.

- 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

Watch this video to learn about creating and associating Smart Lists in Planning.

▶ Video

➤ To create or work with Smart Lists:

1 Select **Administration**, then **Manage**, then **Smart Lists**.

2 Perform one action:

- To create a Smart List, click **Create**, enter the name, and click **OK**.

- To change a Smart List, select it and click **Edit**.

- To delete Smart Lists, select them, click **Delete** and **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 **Synchronize** to synchronize Smart Lists between an application that uses Planning application administration and a reporting application. See "Synchronizing Smart Lists in Reporting Applications" on page 252.

Synchronizing Smart Lists in Reporting Applications

For applications that map Smart Lists to dimensions in reporting applications, you can synchronize Smart Lists in the Planning application. This 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 "Mapping a Planning Application to a Reporting Application" on page 178.

➤ To synchronize Smart Lists in reporting applications:

1 Refresh the application database. See "Creating and Refreshing Application Databases" on page 32.

2 Refresh the reporting application mapping. See "Mapping an Application for Reporting" on page 177.

3 Select **Administration**, then **Manage**, then **Smart Lists**.

4 Click **Synchronize**, then click **OK**.

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 is 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. See .

5 If Smart List items are mapped to more than one dimension, create a new Smart List with a new name, and then manually transfer related data.

> **Note:** Smart List names cannot have spaces in them. If you are synchronizing Smart Lists in a reporting application, ensure that any new members do not have spaces in the name.

Adding or Changing Smart List Properties

Use the Edit Smart List Properties tab to set Smart List properties.

➤ To set Smart List properties:

1 Select **Administration**, then **Manage**, then **Smart Lists**.

2 Select a Smart List and click **Edit**.

3 Define Smart List properties on **Properties**:

Table 43   Smart List Properties

Property	Description
Smart List	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.
Label	Enter the text to display when the Smart List is selected. Spaces and special characters are allowed.
Display Order	How Smart Lists are sorted in the drop-down list: by ID, Name, or Label
#MISSING Drop-Down Label	Enter a label (for example, "No Justification") to be displayed as an entry in the Smart List whose value is #MISSING. **Notes:** ● It displays as the first selection in the Smart List drop-down, allowing #MISSING as a selection in the form. ● When the cell is not 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. ● #MISSING labels determine only the display of cells with #MISSING data; #MISSING remains the stored value.

Property	Description
#MISSING Form Label	Determines how #MISSING values are represented in cells associated with Smart Lists. Options: ● Drop-Down Setting: Displays the label set in #MISSING Drop-Down Label. ● **Form Setting**: Displays #MISSING or leaves cells blank, depending on the Display Missing Values As Blank selection for the form. This selection determines what is displayed in the cell when it is not the focus. When the cell is in focus, the Smart List item that is selected from the drop-down is displayed.
Automatically Generate ID	Generate a numeric ID for each Smart List entry. If you do not select this option, you can customize Smart List ID values.

4 Click **Save**.

5 Select **Entries**.

Use the Entries tab to define selections on Smart Lists.

Adding or Changing Smart List Entries

Use the Edit /Add Smart Lists Entries tab to define the selections in the Smart List.

➤ To define Smart List entries:

1 Select **Administration**, then **Manage**, then **Smart Lists**.

2 Select a Smart List and click **Edit**.

3 On **Entries**, define drop-down list items:

● **For first items only**: enter information into the first row.

● To add an item, click **Add** and enter the information.

● To delete an item, select it and click **Delete**.

● To edit an item, change the information in its row:

Table 44   Smart List Entries

Entry Property	Description
ID	Unique number that sets the order for the displayed entry. Customizable only if **Automatically Generate ID** is not selected on the Properties tab.
Name	Unique alphanumeric name containing alphanumeric and underscore characters (for example: Customer_Feedback) and no special characters or spaces
Label	Displayed text for the Smart List entry on the drop-down list (for example: Customer Feedback).

Items highlighted in red are duplicates.

4 Perform one action:

● Click **Save**.

● Select **Preview**.

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.

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:

Option	Guideline
Blank	When designing forms, select **Display Missing Values as Blank**. When setting Smart List properties, select **Form Setting**.
#MISSING	When designing forms, do not select **Display Missing Values as Blank**. When setting Smart List properties, select **Form Setting**.
A custom label, such as "No Change"	When setting Smart List properties, enter the custom label in the **#MISSING Drop-Down Label** field (for example, `No Change`). Select **Drop-Down Setting**.

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 plan types that are not the source plan type 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 planning 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 select UDAs for members:

1 Select **Administration**, then **Manage**, then **Dimensions**.

2 Select the dimension for whose members to associate the UDA.

3 From the dimension hierarchy, select a member and click **Edit**.

4 Select **UDA**.

5 **Optional**: To create a UDA, click **Create**.

6 Select UDAs for the member by moving them to **Selected UDA** and clicking **Save**:

- ⟫ move selected UDAs

- ⟨ remove selected UDAs

- ⟪ remove all UDAs

Creating UDAs

➤ To create UDAs:

1 Navigate to the **UDA** tab.

2 On **UDA**, click **Create**.

3 Enter a name and click **Save**.

> **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.

Changing UDAs

➤ To change UDAs:

1 Navigate to the **UDA** tab.

2 On **UDA**, select a UDA and click **Edit**.

3 Change the name and click **Save**.

Deleting UDAs

Deleting a UDA removes it for the whole dimension.

➤ To delete UDAs:

1 Navigate to the **UDA** tab.

2 Select the UDA and click **Delete**.

   If you delete UDAs, you must update all member formulas, calc scripts, and reports that reference them.

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 Select **Administration**, then **Manage**, then **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:

   ● **Plan Type**

   Note: A formula entered for the default plan type will be applied to all plan types unless it is overridden by a different formula entered for a specific plan type.

   ● **Data Storage**—Select a data storage option. The default is **Store**.

   Note: The plan type specific data storage field will not display the **Shared** or **Label Only** options. This is because a member cannot be set to Shared or Label Only in one plan type and not another.

   ● **Solve Order**—For aggregate storage plan types 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.

See the on-premises *Oracle Essbase Database Administrator's Guide* for syntax, rules, and usage on Essbase formulas.

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   **If the member formula is not valid, click Show Details.**

If the member formula is valid, **Show Details** is not selectable.

3   **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 is not 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   **Select Administration, then Manage, then Dimensions.**

2   **Select the dimension for whose member to add or change a formula.**

3   **Select the member and click Edit.**

4   **Select Member Formula.**

5   **Select options for the following fields:**

- **Plan Type**

    Note: A formula entered for the default plan type will be applied to all plan types unless it is overridden by a different formula entered for a specific plan type.

- **Data Storage**—Select a data storage option. The default is **Store**.

    Note: The plan type specific data storage field will not display the **Shared** or **Label Only** options. This is because a member cannot be set to Shared or Label Only in one plan type and not another.

- **Solve Order**—For aggregate storage plan types 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. See the on-premises *Oracle Essbase Database Administrator's Guide* for syntax, rules, and usage on Essbase native formulas.

Planning provides predefined formula expressions that you can use in member formulas. You cannot 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 do not 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 cannot have extra spaces or characters such underscore (_).

- You can include subcalls to other functions within an expression.

- Do not enter text where a number is required.

- The order of the outline is important in a member formula. For example, do not 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 (Planning 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

Planning formula expressions can include these predefined variables and functions.

Table 45   Variables in Formula Expressions

Variable	Description
OpenInputValueBlock	Generates an IF statement if the Planning application is a multicurrency application, or an empty string if it is one currency application. Used with ClosedInputValueBlock.
CloseInputValueBlock	Generates an End IF statement if the Planning application is a multicurrency application, or an empty string if it is one currency application. Used with OpenInputValueBlock.
NumberOfPeriodsInYear	Returns the number of time periods in the year
NumberOfYears	Returns the number of years in the application

Table 46   Functions in Formula Expressions

Function	Description
Dimension(dimTag)	Returns the name of a predefined dimension. The dimtags are: ● DIM_NAME_PERIOD ● DIM_NAME_YEAR] ● DIM_NAME_ACCOUNT ● DIM_NAME_ENTITY ● DIM_NAME_SCENARIO ● DIM_NAME_VERSION ● DIM_NAME_CURRENCY
Period(periodName)	Returns the specified period. The periodName options are: ● FIRST_QTR_PERIOD ● SECOND_QTR_PERIOD ● THIRD_QTR_PERIOD ● FOURTH_QTR_PERIOD ● FIRST_PERIOD ● LAST_PERIOD
CrossRef(accountName)	Generates a cross reference for the account
CrossRef(accountName, prefix)	Generates a cross-reference for the account. The account name contains a prefix that you define. The default prefix is No, followed by a blank space and the account name, for example, No Salary.
getCalendarTPIndex()	Generates a member formula that returns an index for the time period; the index is based on the calendar year.
getFiscalTPIndex()	Generates a member formula that returns an index for the time period; the index is based on the fiscal year.
CYTD(memberName)	Generates a calendar year-to-date formula for the member

Function	Description
CYTD(memberName, calTpIndexName, fiscalTpIndexName)	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."

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, click **Show 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 strings are required

A

Naming Restrictions

Restrictions for Applications and Databases

When naming an application, follow these rules:

- Use no more than eight characters for an application's name.

- Do not use spaces.

- Do not 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, do not use extended characters (except for underscores).

- For aggregate storage databases, do not 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.

- Do not use HTML tags in member names, dimension names, aliases, and descriptions.

- Do not use quotation marks, brackets, backslashes, or tabs. Brackets are permitted but not recommended in block storage outlines. They cause errors when converting to aggregate storage outlines.

- To begin dimension or member names, do not use these characters:
  - at signs
  - backslashes
  - brackets
  - commas
  - dashes, hyphens, or minus signs
  - equal signs
  - less than signs
  - parentheses
  - periods
  - plus signs
  - quotation marks
  - underscores
  - vertical bars

- Do not place spaces at the beginning or end of names. Such spaces are ignored.

- Do not use forward slashes in member names.

- For time periods in custom calendars, do not use spaces in prefixes.

- Do not 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, do not use History, Year, Season, Period, Quarter, Month, Week, or Day.

❍ 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
- SKIPNONE
- SKIPZERO
- STATUS
- TO
- TOLOCALRATE
- TRAILMISSING
- TRAILSUM
- TYPE
- UMINUS
- UPPER
- VARORXMBR

- ❑ XMBRONLY

- ❑ $$UNIVERSE$$

- ❑ #MISSING

- ❑ #MI

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		>	greater than sign
*	asterisk		<	less than sign
@	at sign		()	parentheses
\	backslash		%	percent sign
{ }	braces		.	period
:	colon		+	plus sign
,	comma		;	semicolon
-	dash, hyphen, or minus		/	slash
!	exclamation point		~	tilde
=	equal sign			

In calculation scripts and formulas, enclose member names that are also Oracle Essbase keywords in quotation marks (" ") 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 Names

User names can have up to 80 characters.

B

Form Formula Functions

Using Formulas and 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" on page 270.

Note: Planning requires level 0 members that are Dynamic Calc to have a member formula. For Dynamic Calc members that do not have a formula, Planning inserts a semicolon (;) when refreshed.

Creating Formulas

➤ To create formulas:

1 Select the row or column for which to associate the formula.

2 If it does not automatically expand, select **Segment Properties**.

3 In Formula, enter the name for the formula and then click ✎ .

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" on page 270.

5 Click **Validate** to ensure that the formula does not 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 ✏ 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 the **Delete** button.

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" on page 70.

The syntax for mathematical functions:

```
FunctionName(arguments)
```

Table 47 Mathematical Function Syntax

Variable	Description
FunctionName	The name of a mathematical function
arguments	A numeric value, a row, column, or cell reference, or an embedded function

Table 48 Mathematical Functions

Function	Description
Abs	Returns the absolute value of numeric values or references
Average	Returns the average of a group of numeric values or references
AverageA	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.
Count	Returns the number of values in a group of numeric values or references

Function	Description
CountA	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.
Difference	Returns the absolute value of a numeric value or reference subtracted from another numeric value or reference
Eval	Evaluates an expression. Eval is useful for embedding expressions as function arguments
IfThen, If	Returns one value if a condition equals true, and another value if a specified condition equals false
Max	Returns the maximum value from a group of numeric values or references
Min	Returns the minimum value from a group of numeric values or references
Mod	Returns the remainder, modulus, from a division formula
PercentOfTotal	Returns the result of a numeric value or reference divided by another numeric value or reference, multiplied by 100
Pi	Returns the number 3.14159265358979, to 15 digits
Product	Multiplies all numbers or references and returns the product
Random	Returns a random number between 0.0 and 1.0
Rank	Returns the highest or lowest value of a specified column or row
Round	Rounds a number up or down by specified digits
Sqrt	Returns the square root of a numeric value, row, column, or cell
Sum	Returns the sum of a group of numeric values or references
Truncate / Trunc	Removes the specified number of digits from numeric values
Variance / Var	Evaluates the difference between the specified values based on the account type for the current account
VariancePercent / VarPer	Evaluates the percentage difference between the specified values based on account type for the current account

Arguments

Mathematical functions accept numeric values, row, column, or cell references, or embedded functions as arguments. There are four argument types:

- Numeric
- Property
- Row, column, or cell reference
- Embedded functions

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:

```
FunctionName(GridName.GridElement[segment(range)].Property)
```

Table 49   Argument Components

Argument	Description
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 is not specified, the default is the name of the current form.
GridElement	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 is not 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.
segment	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.
range	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* is not provided, all expanded cells are used. **Note:** If a segment expands to only one row or column, do not use the range argument.

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:
	
	`Average(row[1], 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:
	
	`Average(row[1].average, row[2].average)`
	
	The default property for a row, column, or cell reference is sum . For example, `row[2]` is equivalent to `Sum(row[2])`.

Because segment is the only required part of a reference, the following references are the same:

```
Grid1.row[1].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):

```
row[1].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:

```
row[1].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 Property Argument (see "Aggregate Property Argument" on page 273)
- Reference Property Argument (see "Reference Property Argument" on page 274)

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:

```
FunctionName(GridName.Gridelement[segment(range)].property)
```

You apply the following aggregate properties to a row, column, or cell reference.

Table 50   Aggregate Properties

Properties	Description
Average	Returns the average of a row, column, or cell. The calculation excludes #MISSING and #ERROR values.
AverageA	Returns the average of a row, column, or cell. The calculation includes #MISSING and #ERROR values.
Count	Returns the number of values in a row, column, or cell. The calculation excludes #MISSING and #ERROR values.
CountA	Returns the number of values in a row, column, or cell. The calculation treats #MISSING and #ERROR values as zero (0).
Max	Returns the maximum value of a row, column, or cell
Min	Returns the minimum value of a row, column, or cell
Product	Returns the product of rows or columns

Properties	Description
Sum	Returns the sum of a row, column, or cell

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:

```
Average(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:

```
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
```

Argument	Description
AXIS	One of these keywords: row, column, or cell Optional
Segment(range)	Indicates any valid axis reference, such as a row number, column letter
IfNonNumber	Indicates how to treat missing or error data within the Axis Ref
(arg)	Indicates what number to use if missing or error data is encountered within the AxisRef
AggregateProperty	The aggregate function is used for aggregate segments Optional

For 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:

sum(row[3:5], avg(row[4:6], 40, 50), 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 does not change. The function syntax:

> **Abs (**argument**)**

where *argument* is one of the following:

Argument	Description
numeric	A numeric value. For example, Abs (- 20) returns the value 20. Numeric values can include decimals and negative values.
row, column, or cell reference	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`.
function	An embedded function

Examples:

The following expression includes a numeric argument and returns the value 30:

> `Abs ( -30 )`

The following example returns the absolute value of the value in row 1:

> `Abs(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 does not 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:

Argument	Description
numeric	A numeric value. For example, Average (10, 20, 30) returns the value 20. Numeric values can include decimals and negative values.
row, column, or cell reference	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.
function	An embedded function

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:

```
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:

```
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:

```
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:

```
AverageA(arguments) or AvgA(arguments)
```

where *arguments* is one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, AverageA (10, 20, 30) returns the value 20. Numeric values can include decimals and negative values.
row, column, or cell reference	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.
function	An embedded function

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:

```
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:

```
Count (arguments)
```

where *arguments* is one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, Count (10, 20, 30) returns the value 3. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

Examples:

The following example returns the count of three rows, 1, 6, and 8:

```
Count(row[1], row[6], row[8])
```

The following example returns the count of 3 columns:

```
Count( column[E], column[G], column[I] )
```

The following example calculates the count of the cell located at row 4, column D:

```
Count(cell[D,4])
```

The following example calculates the count of aggregate row 3 in grid 5:

```
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:

CountA(arguments**)**

where *arguments* is one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, `CountA(10,20,30,50)` returns the value 4. Numeric values can include decimals and negative values.
row, column, or cell reference	A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax: `CountA(GridName.GridElement[segment(range)]).property`
function	An embedded function

Example

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:

```
CountA([1:4])
```

The following example returns the count of four rows:

```
CountA(row[1], row[6], row[8] row[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:

Difference(arg1, arg2)

where *arg2* is subtracted from *arg1* and are one or more of the following arguments:

Argument	Description
numeric	A numeric value. For example, Difference (3, 5) returns the absolute value 2. Numeric values can include decimals and negative values.
row, column, or reference	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])
function	An embedded function

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:

```
Difference(3, -5)
```

The following example calculates the difference of two aggregate columns:

```
Difference( column[E], column[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:

Eval(*expression*)

where *expression* is one or more of these arguments:

Argument	Description
numeric	A numeric value. Numeric values can include decimals and negative values.
row, column, or reference	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`
function	An embedded function
operators	Use any of the supported arithmetic operators (+, -, *, /, ^, %, and so on).

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 is supported by a conditional operator.

Table 51   Conditional Operators

Conditional Operator	Syntax	Logic
Equal To	*expression = expression*	Tests whether the left expression is equal to the right expression. **Note:** The routine that evaluates the condition does not consider any rounding. If rounding is required, use the Round function. Example**:** 1= 4 Returns false
Greater Than	*expression > expression*	Tests whether the left expression is greater than the right expression. **Example:** 1 > 4 Returns false
Greater Than or Equal To	*expression >= expression*	Tests whether the left expression is greater than or equal to the right expression. **Note:** The correct syntax is ">=". The syntax "=>" is not supported. **Example:** 1 >= 4 Returns false
Less Than	*expression < expression*	Tests whether the left expression is less than the right expression. **Example:** 1 < 4 Returns true

Conditional Operator	Syntax	Logic
Less Than or Equal To	`expression <= expression`	Tests whether the left expression is less than or equal to the right expression. **Note:** The correct syntax is "<=". The syntax "=<" is not supported. **Example:** `1 <= 4` Returns true

Not Equal To	`expression <> expression`
	
	`expression != expression`
		Tests whether the left expression is not equal to the right expression.
		
		Note: The routine that evaluates the condition does not consider any rounding. If rounding is required, use the Round function.
		
		Example:
		
		`1 <> 4`
		
		Returns true
		
		`1 != 4`
		
		Returns true

IsMissing	`IsMissing (reference)`
	
	`IsMiss (reference)`
		Tests whether the reference contains a #MISSING result.
		
		Note: If the reference is an expanded row or column, then all resulting cells must be #MISSING in order for the condition to be true.
		
		Example:
		
		`IsMissing([1])`
		
		Returns true if row 1 has a #MISSING value.

IsError	`IsError (reference)`
	
	`IsErr (reference)`
		Tests whether the reference contains an #ERROR result.
		
		Note: 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.
		
		Example:
		
		`IsError([2])`
		
		Returns true if row 2 has a #ERROR value.

IsNonNumeric	`IsNN (reference)`
	
	`IsNonNumerid (reference)`
	
	`IfNN (reference)`
	
	`IfNonNumber (reference)`
		Tests whether the reference contains a #MISSING or #ERROR results.
		
		Note: 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.
		
		Example:
		
		`IsNN([3])`
		
		Returns true if row 3 has a #MISSING or #ERROR value.

Parenthesis	`(condition)`	Used to group a condition. Used mostly for visual clarity. **Example:** `(1 > 4)` Returns false

Table 52   Conditional Operators

Complex Conditions	Syntax	Logic

And	`(condition AND condition)`
	
	`(condition & condition)`
		Complex condition used to compare two conditions. Returns true if all conditions result in true.
		
		Example:
		
		`(1 > 4 AND 5 > 2)`
		
		Returns false

Not	`NOT (condition)`
	
	`! (condition)`
		Used to negate the result by reversing the result of the condition.
		
		Example:
		
		`Not (1 > 4)`
		
		Returns true

Or	`(condition OR condition)`
	
	`(condition | condition)`
		Complex condition used to compare two conditions. Returns true if any of the conditions result in true.
		
		Example:
		
		`(1 > 4 OR 5 > 2)`
		
		Returns true

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 does not 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])
```

Invalid example:

```
If ( [A] > [B] and [A] > 1000, [A], [B])
```

Max

Max function is a mathematical function that returns the maximum value in a group of numeric values, rows, columns, or cells. The function syntax:

```
Max (arguments)
```

where *arguments* is one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, Max (10, 20, 30) returns the value 30. Numeric values can include decimals and negative values.
row, column, or cell reference	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.`
function	An embedded function

Examples:

The following example returns the maximum value in rows 1, 6, and 8:

```
Max(row[1], row[6], row[8])
```

The following example calculates the maximum of the sums of aggregate rows:

```
Max(row[1].sum, row[2].sum, row[3].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:

```
Min (arguments)
```

where *arguments* is one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, Min (10, 20, 30) returns the value 10. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

Examples:

The following example returns the minimum value in rows 1, 6, and 8:

```
Min (row[1], row[6], row[8])
```

The following example calculates the minimum of the sums of aggregate rows:

```
Min(row[1].sum, row[2].sum, row[3].sum)
```

Mod

Mod is a mathematical function that returns the remainder, or modulus, from a division. The function syntax:

Mod (*arg1, arg2*)

where *arg2* is the divisor and *arg1* and *arg2* are one of these arguments:

Argument	Description
numeric	A numeric value. For example, Mod (6, 4) returns the value 2. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

Example:

The following example divides 10 by 5 and returns the remainder of 0:

```
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:

PercentOfTotal (arg1, arg2**)**

- where *arg1* is a component of the running total (*arg2*). Normally, this is a row or column reference.

- where *arg2* is the running total relative to *arg1*. Normally this is a cell reference containing the grand total.

- *arg1* is divided by *arg2*, with the result multiplied by 100. *Arg1* and *arg2* are one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, PercentofTotal (100, 20) returns the value 500. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

Examples:

The following example returns the value of 5 percent.

```
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:

```
PercentOfTotal ([A], [A,5]),
```

Using the above example, the following table shows the PercentOfTotal results in column B:

		A	B
1		Sales	% Total
2	Mkt1	60	**20%**
3	Mkt2	120	**40%**
4	Mkt3	120	**40%**
5	Total Mkt	300	**100%**

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:

Argument	Description
numeric	A numeric value. For example, Product(2, 20) returns the value 40. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

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*, *integer*)

where *arg1* is one or more of these arguments:

Argument	Description
numeric	A numeric value. For example, Round(81.3987,3) returns the value 81.399. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

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:

```
Round(3594.5567,3) = 3594.557
```

The following example rounds to the nearest integer:

```
Round(3594.5567,0) = 3595
```

The following example rounds to the thousands. This is also known as scaling:

```
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:

```
Sqrt (argument)
```

where *argument* is one of these arguments:

Argument	Description
numeric	A numeric value. For example, Sqrt(100) returns the value 10. Numeric values can include decimals and negative values.
row, column, or cell reference	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`
function	An embedded function

Example

The following example returns the value of 4:

```
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:

Argument	Description
numeric	A numeric value. For example, Sum(10, 20, 30) returns the value 60. Numeric values can include decimals and negative values.

Argument	Description
row, column, or cell reference	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`
function	An embedded function

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:

Argument	Description
numeric	A numeric value; for example, 234.567.
row, column, or cell reference	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`
function	An embedded function

- 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:

`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:

`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.

`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: `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:

`Var (reference1, reference2)`

where `reference1` and `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 53   Expected Results when Using the Variance Function

Column A	Column B	Var ([A] , [B])=0	Var ([A] , [B])>0	Var ([A] , [B])<0
Asset	Asset	0	Returns a positive value	Returns a negative value

Column A	Column B	Var ([A] , [B])=0	Var ([A] , [B])>0	Var ([A] , [B])<0
Liability	Liability	0	Returns a positive value	Returns a negative value
Equity	Equity	0	Returns a positive value	Returns a negative value
Revenue	Revenue	0	Returns a positive value	Returns a negative value
Expense	Expense	0	Returns a negative value	Returns a positive value

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:

Sales	Expense	Result
-400	100	-500

- When the Variance function is applied to a dimension that is not tagged as an Accounts type, an #ERROR results at runtime.

- #MISSING is treated as zero (0), unless specified differently using the ifnonnumber property.

Examples

The Variance function accepts cell, column, or row references only.

Syntax	Example
Sample syntax referencing a column:	Var ([A], [B])
Sample syntax referencing a row:	Var ([3], [4])
Sample syntax referencing a cell:	Var (Cell [3,A], [3,B])

In this example, the variance between column A (**Actual**) and column B (**Budget**) is calculated as:

Var([A],[B])

This example produces the following report:

```
                      Year    Product Market
            Actual   Budget  Variance
          ======== ======== ========
Sales(Income)   400,855  373,080    27,775
COGS (Expense)  179,336  158,940   -20,396
```

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 54   Expected Results when Using the VariancePercent Function

Col A	Col B	VarPer ([A] , [B])=0	VarPer ([A] , [B])>0	VaPer ([A] , [B])<0
Asset	Asset	0	Returns a positive value	Returns a negative value
Liability	Liability	0	Returns a negative value	Returns a positive value
Equity	Equity	0	Returns a positive value	Returns a negative value
Revenue	Revenue	0	Returns a positive value	Returns a negative value
Expense	Expense	0	Returns a negative value	Returns a positive value

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:

Sales	Expense	Result
-400	100	-5.

- When the VariancePercent function is applied to a dimension that is not 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.

Syntax	Example
Sample syntax referencing a column	`VarPer ([A], [B])`
Sample syntax referencing a row	`VarPer ([3], [4])`
Sample syntax referencing a cell	`VarPer (Cell [3,A], [3,B])`

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:

	Year Actual	Product Budget	Market VariancePercent
Sales(Income)	400,855	373,080	7%
COGS (Expense)	179,336	158,940	–13%

C Using Smart View to Import and Edit Planning Metadata

Note: All procedures described in this appendix are performed within the Smart View application. To work with this feature, install the Planning Admin Extension. To install the Planning Admin Extension, see "Installing Clients" in *Using Oracle Planning and Budgeting Cloud Service*.

You can use the Smart View application to quickly perform several tasks with your Planning metadata:

- "Importing Planning Dimensions in Smart View" on page 293
- "Editing Members in Smart View" on page 295
- "Adding Planning Members in Smart View" on page 296
- "Moving Members in Smart View" on page 297
- "Designating Shared Members in Smart View" on page 298
- "Refreshing and Creating Cubes in Smart View" on page 299

Watch this video to learn about entering data in Planning using Smart View.

▶ Video

Importing Planning Dimensions in Smart View

Importing a Planning dimension into Smart View allows you to rapidly add, edit, and move the members of the dimension.

- "Importing Dimensions in Smart View" on page 294

Importing Dimensions in Smart View

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 From the Smart View Panel, select **Planning**.

3 Right-click **Dimensions** in a folder. The Planning dimensions in the folder are displayed.

    Note: The Time Period and Attributes dimensions are not displayed.

4 Right-click the dimension name and select **Edit Dimension**.

Using Smart View Grids

The Smart View grid allows you to rapidly add, edit, and move the members of a Planning dimension.

Overview of the Smart View Grid

A Smart View 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. Each member in a metadata dimensions corresponds to a specific member property valid for the corresponding Planning dimension. Instead of containing numeric values, the metadata grid data cells hold the specific, corresponding property values.

The Smart View Grid Display

The Smart View grid displays the Planning dimension with member names in rows and member properties in columns.

The ribbon displayed here is similar to the ribbon in a conventional (non-Planning) Smart View grid but contains fewer options.

For detailed information on using the Smart View ribbon, see "Ad Hoc Analysis" in the *Working with Smart View for Oracle Enterprise Performance Management Cloud Services*.

Guidelines for Using the Smart View Grid

The following guidelines will assist you in using the Smart View grid to edit Planning metadata:

- The following functionality is not 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 does not 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.
- After a metadata grid is opened, it cannot 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.

Editing Members in Smart View

The Smart View 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" on page 294).

2 Highlight a member property in the grid.

3 In the drop-down menu, select a value.

   Note: Modified cells are displayed in a different color.

4 Click **Submit Data** to save the grid.

Note: You can save changes to multiple members and properties in the same **Submit Data** operation. If Submit Data fails for one member, the Planning server will stop the operation and not save any changes.

Adding Planning Members in Smart View

The Smart View grid allows you to rapidly add members to a Planning dimension.

- "Adding Members in Smart View" on page 296
- "Guidelines for Adding Members in Smart View" on page 297

Adding Members in Smart View

The Smart View grid allows you to rapidly add members to a Planning dimension.

- Watch this video to learn about adding Entity dimension members in Smart View.

  ▶ Video

- Watch this video to learn about adding Account dimension members in Smart View.

  ▶ Video

- Watch this video to learn about adding Scenario dimension members in Smart View.

  ▶ Video

➤ 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" on page 294).

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**.

  Note: If your administrator has set the member add mode to Dimension Editor, you must perform a **Refresh** after you add a member. If the member add mode is Submit without Refresh, you are not required to perform a Refresh. If you are not certain which mode your Smart View application is using, check with your administrator.

  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" on page 295.)

> **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 the following:

- To modify the properties of the new member, see "Editing Members in Smart View" on page 295.

- 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 Planning to a new member. The specific default properties are based upon those of the dimension.

Moving Members in Smart View

The Smart View 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 *Working with Smart View for Oracle Enterprise Performance Management Cloud Services*.

➤ 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" on page 294.)

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 Planning server stops the operation and does not 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 Smart View.

Designating Shared Members in Smart View

The Smart View grid allows you to rapidly designate members for sharing within a dimension.

> **Note:** The Shared data storage option is not available for members of the Currency dimension in Smart View.

➤ 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.

> **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 Cubes in Smart View

The Smart View grid allows you to quickly refresh a cube or create a new one.

➤ To refresh or create a cube in Smart View:

1 From the Smart View ribbon, click **Panel**.

2 From the Smart View Panel, select **Planning**.

3 Right-click the folder name and select **Dimensions** in a folder. The Oracle Hyperion Planning dimensions in the folder are displayed.

> **Note:** The Time Period and Attributes dimensions are not displayed.

4 Right-click the *root* Dimension and select **Refresh Database** or **Create Database**. The Oracle Smart View for Office 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.